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

Raviv Murciano-Goroff*

Stanford University

January 28, 2018

— Draft —

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 9.84% 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 12.37% 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.

*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. 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 2017, tech giants such as Google, Facebook, and Twitter had 20%, 19%, and 15% 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] 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 gender 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 at tech firms using data from a large online recruiting platform. On this platform, job seekers advertise their credentials and skills by composing digital resume "profiles." Recruiters view job seekers' profiles and choose the individuals they wish to contact regarding job openings.

[1]The data for these statistics comes from the equal-opportunity data websites for these firms as well as news reports. Data for Google is available at https://www.google.com/diversity/, Facebook at https://www.cnbc.com/2017/08/02/facebook-diversity-numbers.html, and Twitter at http://fortune.com/2017/01/19/twitter-diversity/ and https://blog.twitter.com/en_us/topics/company/2017/building-a-more-inclusive-twitter-in-2016.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/.

The first question this paper asks is which candidate attributes do recruiters show interest in. On the platform examined in this study, recruiters can view candidates' work histories, educational credentials, as well as a list of skills that candidates self-report knowing. While recruiters could have focused on concrete credentials and treated the self-reported skills as "cheap talk," I show that self-reported technical skills predict the largest increases in the probability a recruiter expresses interest in a candidate.

The second question asked by this paper is whether or not female programmers who have experience in a programming language list that language among their self-reported skills at the same rate as their male counterparts with similar previous experience. Unlike the educational credentials and work history portions of an individual's profile, the list of self-reported skills allows the job seeker to use his or her discretion regarding which skills to self-promote or advertise to recruiters. Job seekers must decide whether or not they feel that their ability in performing a skill warrants listing based on how they anticipate recruiters will respond to seeing that skill listed. This creates the potential for male and female candidates to have different self-imposed thresholds for self-reporting a skill and different beliefs regarding the response of recruiters.

The third question investigates whether or not recruiters' predicted response to the display of self-reported skills on profiles is different for male and female candidates. If gender differences exist in the propensity of candidates to self-report a skill, I predict that recruiters would use that information to find the most experienced and talented recruits. For example, if female coders are less inclined to self-report a known programming language than their male counterparts with similar previous experience, recruiters could infer that female coders who do self-report that language are more experienced on average than males who also self-report knowing the language.

My analysis utilizes a large and rich dataset from an online recruiting platform.[4] The data includes the information displayed on the candidate profiles of almost 4 million individuals who could be considered by tech firm recruiters when filling software engineering job openings. Of the profiles, just under 21% of the represented individuals are female, though, female candidates make up less than one-tenth of the candidates recruited on the platform.

[4]My agreement with the data provider stipulates that the name of the platform cannot be used in this paper. Furthermore, the names of employers and recruiters were anonymized prior to the analysis.

The data contains two unique elements. First, I observe a means of gauging some of the candidates' actual previous coding experience in a set of programming languages. Approximately 5% of the profiles can be linked to open source software contributions, free computer source code that programmers post online for others to use. When an individual posts source code in a programming language, this code demonstrates the individual's knowledge and experience in that language. Second, I will observe when a recruiter who subscribes to the platform "saves" an individual's profile in order to contact them about job opportunities.

On the labor demand side, I predict which candidates the recruiters decided to contact regarding job openings in order to uncover the relative importance of over 259 different attributes for garnering recruiter attention. On the labor supply side, among those who contribute to open source software projects, I compare whether or not male and female coders with similar levels of previous experience in a programming language, as measured by the quantity and popularity of their open source code, list that language among their self-reported skills on their profile. Finally, I return to the demand side and use a linear prediction model to assess if recruiters leverage information about gender differences by showing more interest in candidates who self-report knowledge of a programming language when the candidate is from a group with a lower propensity to self-report.

I find three empirical results. First, 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 22% 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 9.84% 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 external validation from other coders. 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.

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. Possibly because it indicates individuals' preferences over working with particular technologies, recruiters pay attention to self-reported skills. Female candidates, however, are less incline toward taking the self-promotional action of listing all of the programming languages that they know on their profile. This could be partially due to differences in preferences over occupations and partially due to differences in perceptions regarding the experience threshold required for self-reporting knowledge of a skill. Furthermore, while recruiters could leverage gender differences in the propensity to self-report in order to find the most talented workers, they do not appear to do so. The most likely reason why is because recruiters are unaware of such gender differences as well as an inclination toward not using gender as a factor in hiring decisions. Ultimately, this means that neither the labor supply side nor the labor demand side leverages the self-reporting mechanism in ways that could increase the percentage of women recruited for software engineering positions.

2. Setting and Data

Many tech companies actively solicit job applications from qualified individuals. Recruiters at these companies frequently use online recruiting platforms, which aggregate information about individuals and their qualifications into searchable "candidate profiles." Recruiters can examine the profiles and choose the individuals they wish to contact about job openings.[5]

The platform whose data I use for my analysis in this paper facilitates such recruiting and is popular used by tech company recruiters searching for software engineers. Profiles on this platform are constructed by merging how a candidate describes their own qualifications with the platform's assessment of the candidate's computer programming experience. The top of each profile displays information from what candidates composed about themselves and posted online for recruiters to view. This includes the candidate's educational degrees and work history. In addition, profiles display the "skills" candidates

[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."

write that they feel proficient in under the title "Self-Reported Skills." These skills come in three categories. First, candidates can list non-technical abilities, such as "Writing" or "Social Media." Second, candidates can list technical skills, such as "Machine Learning" or the database program "MySQL." Lastly, candidates can list the names of programming languages they can code in.

Next to the information candidates write about themselves, each candidate's profile displays the platform's estimation of that candidate's programming abilities. Many computer programmers upload open source software, 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. Using proprietary analysis of the candidate's open source code and question answers, the platform displays an estimate of whether the candidate has a "High" or "Low" level of experience in each "Verified" programming language.

A programming language, therefore, can appear on a candidate's profile in one of five different. The candidate can either list the language among their self-reported skills or not; The platform can list the language as "Verified - Low Experience," "Verified - High Experience," or not list it among the verified languages list.

Recruiters search for candidates on the basis of educational credentials, work experience, geographic location, or "verified" programming languages. When viewing a profile, recruiters can click a button indicating that they are interested in contacting the candidate, an action known as "saving" a profile.

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 1. Among the 259 profile attributes, the dataset includes indicator variables for the appearance of 25 different self-report non-technical and technical skills on the profile. Variables are included 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 will only use profiles where the first name of the candidate is strongly associated with either the male or female gender.[6] Finally, I use only the profiles that appeared on the site throughout 2014 to 2016.[7]

The Profiles dataset contains 3,927,150 candidate profiles. Of the candidates, 20.63% are female. Table 2 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 19%, 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 2001. 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: 64.16% of female candidates and 54.40% of the male candidates list at least one skill. The current job titles of 38.54% of male candidates and 27.15% of female candidates are strongly associated with coding. Finally, the rates at which recruiters contacted male and female candidates is different: 1.776% of male candidates and 0.654% of the female candidates were saved by at least one recruiter.

In Table 3, I show the frequency of how candidates who know the programming language JavaScript display this language on their profile. I focus on the programming language JavaScript for three reasons. First, in 2015, JavaScript appeared as a required skill on more job posting online than any other language. In that year, JavaScript was also the language in which the largest share of open source code uploaded in 2015 was written. Lastly, JavaScript is less commonly taught as a first programming language. Therefore,

[6] I drop 14.85% of the observations because I am not able to code the gender of the first name.

[7] 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.

those who know this language likely made some investment in order to learn it.[8]

Each cell in Table 3 shows a percentage of the 263,422 profiles where JavaScript is listed in either the "Self-Reported Skills" or "Verified Languages" lists. For example, the top-right cell shows that 56.19% of these profiles show this language in their "Self-Reported Skills" list but not their "Verified Languages" list.

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," 25.75% also list the language within their "Self-Reported Skills." In contrast, among those who have the language in their "Verified" list with "High Experience," 55.05% also list the language within their "Self-Reported Skills."[9]

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,[10] 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.[11]

This dataset contains 729,426 candidate-language pairs. I use only the set of pro-

[8]Similar patterns are shown for other programming languages in the Online Appendix.

[9]The fraction of those who list the programming languages that they know might seem low, but this all must be taken in the context that not everyone is using the self-reported skills section of their resume. In fact, only 56.41% of the sample list at least one skill—technical or non-technical—to begin with. Among the subsample of profiles that have at least one self-reported skill, the profiles that are "Verified - Low" 60.67% also list JavaScript in the "Self-Reported Skills." Among the "Verified - High" group 79.87% also self-report this language.

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

[11]In this paper, I use the total number of lines of code uploaded to open source in a programming language as a measure of previous experience in that language. Programmers who contribute more lines of code are not necessarily more proficient. Sometimes a short program that accomplishes a task is better than a long one. But the metric of lines of code contributed to open source that I use is measured over an entire lifetime rather than in a specific program. Indeed, the average number of JavaScript lines contributed by an individual is over 30,000. A high count therefore indicates extensive and intense work in a given language. In the Online Appendix, I show the results of this paper using other measures of previous coding experience in a language. For example, I also use the number of days in which the individual uploaded open source code, the number of StackOverflow answers to questions about that programming language submitted, as well as the number of "watchers" of the candidate's GitHub open source repositories that use that programming language.

gramming 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. In short, these languages define the largest labor markets for coders. In addition, I use only profiles that list at least one self-reported skill.[12] 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.[13]

After the sample selection, the dataset, which I refer to as the "OS Contributors" dataset, contains 149,141 candidate-language pairs. These pairs are derived from 59,394 distinct candidates. This sample is 5% of the main sample, but accounts for 40% 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 6. 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 48,341.68, but the median is 5,174. 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 5 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 182,303 profiles 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 received recruiter attention. In addition, this subsample is more gender skewed: a mere 7% of the open source contributors are female. On average, women also make open source contributions in a smaller number of programming languages, and thus have fewer "Verified" languages listed on their profiles. In addition, the average number of programming languages that appear in the "Self-Reported Skills" section of the female profiles are lower than the average for

[12]The exact languages included include, JavaScript, C#, Python, and Ruby.

[13]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.

male profiles.

Figure 1 shows the number of lines of JavaScript source code uploaded by candidates to open source repositories over their lifetime. For candidates with at least one line of open source code in JavaScript, I split the candidates into deciles by the quantity of code uploaded. The blue histogram shows the fraction of male candidates in each decile. The red histogram shows the fraction of female candidates in each decile. The horizontal axis is labeled with the mean number of lines of code contributed within each decile (divided by 1,000 and rounded for aesthetics).

These overlapping histograms show that, among open source contributors, the average female contributor uploads slightly fewer lines of code than males do. The mean and median lines of code uploaded by male candidates are 49,086.23 and 5,418 respectively, while for female candidates they are 38,798.98 and 3,134.

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.

3. Empirical Implementation

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 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 the candidate's attributes:

$$saved_i = \sum_{l \in \mathcal{L}} \beta_{1,l} sr\_only_{i,l} + \beta_{2,l} vl\_only_{i,l} + \beta_{3,l} sr\_and\_vl_{i,l} + \beta_{4,l} vh\_only_{i,l} + \beta_{5,l} sr\_and\_vh_{i,l}$$
$$+ \alpha + \gamma SRS_i + \theta X_i + \epsilon_i \tag{1}$$

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: in the self-reported list, one of the "Verified" lists, or both the "Self-Reported" and one of the "Verified" lists. I include these indicators for each of the follow languages: JavaScript, Java, Python, and C#. This set of languages is denoted by $\mathcal{L}$. I also include a vector of indicators, SRS, represent the 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 1. 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.

Next, I test if recruiters respond to candidate's self-report of a language even if the candidate's profile displays "Verified" experience in that language. I perform this test using the candidates "Verified" in the programming JavaScript. I examine profiles that are extremely similar, including their "Verified" level of experience in JavaScript, but vary in whether or not they list JavaScript as a self-reported skill. I refer to these sets of similar profiles as "Profile Groups." The profile groups are constructed by matching precisely all of the attributes from Table 1 as well as which lists the programming languages Java,

Python, and C# appeared in on the profile. I do not, however, match the gender of the candidate or whether or not JavaScript is listed on the self-reported skills list.[14] 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.

This test uses the structure of the platform's profile search engine: When recruiters on the platform searched for candidates with knowledge of a particular programming language, the results shown were based on the "Verified" programming languages only and not the self-reported skills. In other words, if a recruiter searches for candidates who know the programming language JavaScript, the candidates who self-report knowing the language but do not have the language listed as "Verified" would not be shown to recruiters. Any search a recruiter would have performed would have shown profiles within the same profile group as results. Therefore, 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.[15]

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

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.

The coefficient on the indicator for self-reporting the language 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. By including profile group fixed effects, these comparisons are made within profiles that show otherwise similar information.[16]

[14]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.

[15]Because there are endogenous indicators included as covariates, I chose to estimate this as a linear probability model as a probit could create additional issues (Heckman, 1978). See the Online Appendix for the probit margins.

[16]Because the coefficient on sr_i is in levels, I will estimate this equation separately for "Verified - Low" profiles and "Verified - High" profiles. Otherwise, the highly recruited profile groups would drive the results.

3.1 Testing for Gender Differences in the Propensity to Self-Report 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. In particular, if a male and female candidate both have previous experience coding in the programming language JavaScript, do they both list that language among their "Self-Reported Skills"?

I use open source code contributions to find a subset of candidates where I can observe a portion of their previous coding work. The OS Contributors dataset of candidate-language pairs is used for examining 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." An observation in this linear probability model is a candidate-language pair where the candidate is "Verified" in the language based on open source contributions:

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

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. I use three different measures of previous experience in a language: the number of lines of open source code uploaded in that language, the number of days with open source code uploads in that language, and the number of StackOverflow answers posted and tagged with that language. As the number of lines of code uploaded is extremely right skewed, I convert that metric into a z-score by subtracting the mean number of lines of code and dividing by the standard error. Finally, C_i represents the year that the candidate graduated from college.

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 ex-

ample, 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.

3.2 Recruiters' Response to Self-Reported Programming Languages

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:

$$
\begin{aligned}
saved_i = &\sum_{l \in \mathcal{L}} \Big(\beta_{1,l} sr\_only_{i,l} + \beta_{2,l} vl\_only_{i,l} + \beta_{3,l} vh\_only_{i,l} + \beta_{4,l} sr\_and\_vl_{i,l} + \beta_{5,l} sr\_and\_vh_{i,l} \Big) \\
&+ female_i \times \sum_{l \in \mathcal{L}} \Big(\delta_{1,l} sr\_only_{i,l} + \delta_{4,l} sr\_and\_vl_{i,l} + \delta_{5,l} sr\_and\_vh_{i,l} \Big) \\
&+ \gamma_1 SRS_i + \gamma_2 female_i \times SRS_i \\
&+ \alpha + \theta X_i + \epsilon_i
\end{aligned}
\tag{4}
$$

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 $sr\_and\_vl_{i,l}$ and $sr\_and\_vh_{i,l}$. 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 $sr\_and\_vl_{i,l}$ and $sr\_and\_vh_{i,l}$ 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.[17]

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

$$
\begin{aligned}
saved_i = & \sum_{l \in \mathcal{L}} \Big(\beta_{1,l} sr\_only_{i,l} + \beta_{2,l} vl\_only_{i,l} + \beta_{3,l} vh\_only_{i,l} + \beta_{4,l} sr\_and\_vl_{i,l} + \beta_{5,l} sr\_and\_vh_{i,l} \Big) \\
& + female_i \times \sum_{l \in \mathcal{L}} \Big(\delta_{1,l} sr\_only_{i,l} + \delta_{2,l} vl\_only_{i,l} + \delta_{3,l} vh\_only_{i,l} + \delta_{4,l} sr\_and\_vl_{i,l} + \delta_{5,l} sr\_and\_vh_{i,l} \Big) \\
& + \gamma_1 SRS_i + \gamma_2 female_i \times SRS_i \\
& + \alpha + \theta_1 X_i + \theta_2 female_i \times X_i + \epsilon_i
\end{aligned}
\tag{5}
$$

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

4. Results

4.1 Recruiters Value Specific Technical Skills

Recruiters could search for candidates based on a variety 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 7 shows the esti-

[17] 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.

mated coefficients from that regression.

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 0.025 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 0.008 higher probability of being saved by a recruiter, while currently being employed by one of the top tier tech companies predicts a 0.049 higher probability of recruiter attention.[18]

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 0.049 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 0.115 higher probability of being saved. Candidates who are "Verified" with "High Experience" in JavaScript are predicted to be saved at a 0.148 higher probability, while those who also self-report knowing the language are associated with a 0.245 higher probability.

The way in which programming languages appear on candidate's 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 0.197 or 88.74% 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 1.57% of those without a verified language, while they saved 29.01% of those with a verified language.

These findings imply that recruiters focused on candidates with demonstrated experience in specific technical skills. A number of aspects of this labor market encourage recruiters to search for candidates based on technical skills. First, the labor market for

[18]My agreement with the data provider prevents me from naming any of the specific companies or universities. While almost all the employers have very small associated increases in the probability of being saved, three companies have large predicted increases. These three companies drive the estimated coefficient.

software engineers exhibits very high churn. Candidate's in the Profiles dataset report switching employers on average every 2.4 years. From the employers 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 their workers 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. In addition, 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, many recruiters believe that recruiting based on a candidate's skills is more "objective" than recruiting on the basis of which school an individual attended.[19]

4.2 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 7 showed candidates with "Low Experience" in JavaScript are predicted to have a 6.5 percentage point higher probability of being saved simply by also self-reporting this language. Candidates with "High Experience" who also self-report are associated with a 9.8 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 complex set of interactions of a candidate's attributes when deciding whether or not to save their

[19]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 more "objective" hiring criteria because of their emphasis on the skills of recruiters.

profile. Therefore, the magnitude of the coefficients on self-reporting might misrepresent the importance of this action. 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 self-reporting garners more recruiter attention even when that language is already shown as "Verified." 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 8 shows the results of estimating this regression. The first column shows the results using only the 95,263 candidates who are "Verified - Low Experience" in JavaScript, while the second column shows the results using the 20,122 candidates who are "Verified - High Experience." 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 0.058. Relative to the probability that a candidate with this level of experience is saved, 0.189, this correspondents to a 30.69% higher predicted rate of being save. Similarly, candidates who are "Verified - High Experience" in JavaScript and self-report are associated with a 0.082 higher probability of being saved, equivalent to a 16.23% 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.

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 based in part on technologies which they enjoy working with as well as those that they feel will help them build experience useful for finding their next position. In fact, as a recent working paper by Sonny 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). Furthermore, recruiters are incentivized to care about whether or not candidates they contact are likely to respond, score well on interviews, and accept job offers if they are extended.[20]

4.3 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 2 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 47,875 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 deciles according to the total number of lines of code in JavaScript they have contributed to open source over their lifetime.[21] The probability that male and female candidates within each decile self-reported JavaScript is shown on the vertical axis.

[20]This is often referred to as the "recruiting funnel" 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 a 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.

[21]The deciles are defined across male and female coders, however, the graph shows the means within each decile separately. Note that female open source contributors contribute slightly less total code on average (see Figure 1). This could be a product of female open source contributors being slightly younger than male contributors. The regression of Equation 3 shown in Table 9 controls for these differences by including the year of college graduation. In the Online Appendix, I also run the same analysis but without the restriction that the candidate has at least one self-reported skill. The results are largely the same.

The figure reveals that within every decile the female coders have a lower probability of self-reporting the programming language on their resume than their male counterparts. The pattern is consistent and robust to various different measures of previous experience in this programming language (see the Online Appendix).

A similar pattern appears when coders are ranked by measures of their reputation in the open source community. In Figure 3, 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.[22] This figure again shows again 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.

In order to explore if the observed gender gap in JavaScript represents a more general difference in the propensity to self-report programming languages, I test test for differences in self-reporting when pooling across programming languages. Table 9 shows the results of estimating Equation 3, which predicts if a candidate lists a programming language as a self-reported skill. The regression uses the 82,779 candidate-language pairs in the OS Contributors dataset. The explanatory variables are measures of a candidate's previous experience in a programming language. The three columns in Table 9 represent running this regression on the three distinct measures of previous experience in a language: the number of lines of code uploaded to open source, the number of days with uploads to open source, and the number of answers to questions related to the language on StackOverflow. Finally, in all three regressions, I control for the year in which the candidate completed their bachelor's degree.

The coefficient estimates confirm differences in the propensity of male and female coders to self-report programming languages in which they have previous experience. All three regressions show a negative coefficient of a similar magnitude on the indicator for the candidate being female. This coefficient reveals that female coders self-report their programming language skills at a probability 6 percentage points lower than their male counterparts. Relative to the probability that the average male coder self-reports, female coders are 9.84%-10.32% less likely to self-report programming languages that they have

[22] "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.

previous experience in.

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. For example, while male coders increase the probability that they self-report a programming language by 0.001 with each additional answer they post regarding a language on StackOverflow, female candidates increase by 0.005. While this is a five times higher increase, the regression predicts that female candidates would need to write at least 12 StackOverflow answers in order to close the average difference in self-reporting between males and females. In this sample, however, 90.07% of the candidates "verified" in JavaScript never answer a single question on StackOverflow and only 1.91% write 12 or more answers. Furthermore, the estimated interaction when using lines of code as the measure of experience is not statistically significant. Indeed, as Figure 2 showed, female coders who have written and uploaded extremely large amounts are less likely than their male counterparts to self-report knowing this language.

As least two possible reasons could contribute tot he observed gender gap in self-reporting. First, men and women may on average have different preferences over occupations or follow different career paths.[23] Second, men and women may on average have 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 a "confidence gap" identified in social psychology and behavioral economics studies (Gneezy et al. 2003; Bursztyn et al. 2017; Niederle and Vesterlund 2007; Correll 2001; Baldiga 2014).[24]

One way to test if gender differences in average preferences over occupations can ac-

[23]Note that these preferences may be formed in part in response to the work environments. For example, given the reports of incidents of discrimination and harassment at tech companies, women who know how to code may be less inclined to work in programming roles or at tech companies (See reports such as http://money.cnn.com/technology/sexual-harassment-tech/, https://www.theatlantic.com/magazine/archive/2017/04/why-is-silicon-valley-so-awful-to-women/517788/, and https://www.nbcbayarea.com/news/local/Gender-Discrimination-Lawsuits-in-Silicon-Valley--436347823.html).

[24]For example, previous studies have shown that female test takers have a higher degree of risk aversion than similar male test takers (Baldiga, 2013). If female candidates for coding positions are more concerned about the possibility of being unable to proficiently answer coding questions during a technical interview for a programming job, this could contribute to the gender gap in self-reporting.

count for the observed gap in self-reporting, I investigate subsamples of candidates with relatively homogenous occupational choices. For example, 45,611 candidates list at least one self-reported skill related to software engineering jobs that is not a programming language. These skills include, "Software Engineering," "Scrum," and "Git." The male and female candidates with at least one of these self-reported skills have similar current occupations: 53.9% of the male candidates and 52.5% of the female candidates have a current job title that is associated with software engineering. In Table 10, I show the results of predicting if a candidate self-reported a language by estimating Equation 3 using the 72,287 candidate-language pairs from these individuals. Again, the female candidates are on average less likely to self-report knowing programming languages they have experience in than their male counterparts. On average, the female candidates in this subsample are 4.70% less likely than the male candidates to self-report languages they have experience in.

While the difference between between male and female candidates in their propensity to self-report is smaller among this subsample, the fact that detectable differences remain indicates that preferences can only account for some but not all of the gender differences in supply side behavior.

Given the results in the previous sections, recruiters appear to search and find candidates largely based on self-reported skills. Since female candidates are on average less likely to self-report their skills, they are also less likely to be contacted by recruiters. While two factors that might be motivating female candidates to self-report at lower rates—a "confidence gap" and preferences over jobs—cannot be differentiate in the current dataset, they would have very different implications.

If female candidates are less interested in jobs involving coding skills, the lower propensity to self-report programming languages simply reflects the optimization of their profiles for occupations they are more interested in. In this case, there is actually a social benefit when candidates only display signals of their actual interests and tech recruiters only contact candidates who are interested in working in computer programming occupations.[25]

If, however, female candidates want coding jobs or if preferences are informed by mes-

[25]Of course, preferences might reflect the widespread reports of tech firms with inappropriate workplace cultures, discrimination, and harassment, which might dissuade qualified and interested candidates from pursuing careers in tech.

sages from recruiters, the lower propensity to self-report might create an inefficiency in the labor market. Female candidates who do not self-report have a substantially lower probability of being contacted by a recruiter, and thus might miss job opportunities. In addition, individuals might learn about available jobs as well as the applicability of their skills for occupations from recruiter solicitations. For example, job seekers might harbor biased self-assessments about their abilities in a particular occupation based on stereotypes and cultural beliefs (Correll, 2001). Much like the psychology literature on "framing," receiving a recruiter's job application solicitation might attenuate these biases and encourage individuals to apply for positions that they might not have considered before. As self-reporting skills is seemingly costless for candidates on this platform, any underreporting of known skills could lower the efficiency of matching talent with employers in this labor market.

4.4 Gender Differences in Recruiters' Response to Profile Attributes

The previous analysis revealed that recruiters seek candidates with extensive experience in technical skills. Furthermore, female candidates who self-report programming languages have on average more experience than their male counterparts who also self-report the same languages. An implication of these results is that recruiters should be more inclined to save female candidates who self-report languages than male candidates with similar profiles. 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 12.

The estimated coefficients on the interaction of self-reporting specific technical skills and the candidate being female are mostly negative. For example, candidates who self-report knowing at least one of the most popular version control software programs, Git and SVN, are predicted to have a 0.0439 higher probability of being saved. The 6,618 female candidates with this skill self-reported have a predicted increase in their probability of being saved that is 0.01 less than the male candidates, equivalent to a 26.36% lower probability. Similarly, the popular big data platform Hadoop, which 1,521 female candidates self-report knowing, is predicted to increase the probability that male candidates are saved by 0.046, but only 0.037 for female candidates. This represents a 19.47% lower increase in probability. Finally, self-reporting the database programs PostgresQL, MongoDB, and Redis show coefficients representing 28.75%, 37.47%, and 88.03% lower predicted gains in the probability of being saved for female candidates relative to their male counterparts.

A similar pattern can be seen for programming languages. Displaying knowledge of a programming language, such as JavaScript, is associated with a higher probability of a recruiter saving a candidate. For example, self-reporting JavaScript and displaying the language as "Verified - Low Experience" is associated with a 0.116 increase in probability of being saved for male candidates, but a 2.416% lower increase for the 1,827 female candidates with profiles in this configuration. Finally, self-reporting JavaScript and displaying the language as "Verified - High Experience" is associated with a 0.248 increase in probability of being saved for male candidates, but a 13.464% lower increase for the 491 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, the appearance of "Project Management" on a male candidate's list of self-reported skills is associated with a 0.002 lower probability of being saved. For the 119,962 female candidates who list this skill, the associated decrease in the probability of being saved is half that of 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 not significantly distinguishable from zero, they are almost uniformly negative and most are economically significant. This implies that recruiters are likely saving male candidates with slightly lower levels of previous experience in programming languages, while overlooking similar female candidates with more experience.[26]

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 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 gender balanced may not be aware that female coders self-report their technical skills with a lower propensity than male coders. Anecdotally, in my conversations with numerous recruiters and human resource managers, no one had thought about the potential for gender differences in the self-promotional behavior of job seekers. Additionally, well intentioned 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. Especially for hiring managers who use recruiting platforms with searching, filtering, and algorithmic candidate recommendations based on keywords, not incorporating an adjustment for the gender of the candidate means that you might get more male candidates simply because of their higher propensity to self-report. Finally, recruiters might have an objective other than getting the most experienced coders. For example,

[26]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 they are at only slightly negative or close to equal treatment. That being said, the gender differences in the predicted probability of being saved do not show systematic differences for the self-reported versus non-self-reported ways of languages being reported. Therefore, it seems unlikely that the information about self-reporting propensity is being used at all.

some companies may be more concerned with employee retention, and thus gravitate toward male candidates because of they on average continue in software engineering occupations for longer than their female colleagues.

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. Many tech companies profess to be actively seeking female tech workers. 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 1, 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 were all male versus the average probability if all the candidates were female.

The estimated coefficients from Equation 5 are shown in Table 13. 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 0.009 higher probability of being saved. For the 154,024 female candidates with this degree, however, the predicted increase is only 50.21% 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 36.36% less attention from potential employers.

Two profile attributes, however, are associated with larger gains for female candidates. First, the coefficient on the candidate being female is 0.001 or 6.67% higher than the unconditional probability of being saved. In addition, female candidates who are programmers at highly ranked tech companies—an attribute which is associated with a 0.026 higher probability of being saved for males—are predicted to garner a 53.85% higher chance of

being saved than their male colleagues.

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 0.0157. Under the hypothetical world in which all candidates are female, the average predicted probability of being saved is 0.0137. This amounts to female candidates being predicted to receive 12.37% less attention from recruiters.[27]

Using the estimated coefficients from the above regression, we can also examine the average predicted probability that the most sought after JavaScript coders, those who self-report and are "Verified - High Experience" are saved. Among the 11,011 candidates in this category, if all the candidates were male, we would predict that 60.39% of them would be saved by a recruiter. If these candidates were all female, we would predict that 54.66% of them, or 9.49% fewer, would be saved.[28]

Another way to visualize that female candidates receive less recruiter attention on average than male candidates with similar previous experience in coding is by comparing candidates based on the popularity of their open source coding work. The GitHub platform, the most popular website for uploading open source computer source code, allows coders to subscribe to updates about the code contributions made by another user. Those who subscribe to updates about another user are known as "followers." The number of "followers" that a coder has on GitHub is an indication of their reputation in this community. In addition, users can subscribe to updates about changes to a particular open source project. Known as "watchers," these users are typically utilizing that project's code or software in their work, and thus they would like to know when bugs are fixed or new features are added. The number of "watches" of an open source project is an indication of the level of interest in that source code.

Among candidates "Verified" in JavaScript, Figure 5 shows the probability that a candidate is saved by a recruiter between 2014 and 2016 versus the number of open "followers" that the candidate has on GitHub. While recruiters show more interest in candidates with stronger reputations in the open source community, female candidates have a lower change of being saved than their male counterparts with the same number of "follow-

[27] This difference is statistically significant at the 0.01 level.
[28] This difference is also statistically significant 0.01 level.

ers." Figure 6 plots the probability that a candidate is saved against the total number of "watchers" to open source projects created by a candidate. Again, while recruiters are more interested in candidates with higher numbers of "watchers," the female candidates less attention from the recruiters on average than male candidates with similar interest in their open source work. In the Online Appendix, I show that these patterns are consistent even after controlling for other profile attributes.

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 the female candidates that are available might not receive as much attention as 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 these possible motivations, however, and further empirical studies and experiments would 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.

4.5 Recruiters vs The Crowd

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 "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 5 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 6 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.

5. 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 for 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 analyze the behav-

ior of individuals and recruiters on a recruiting platform. I focus on quantitatively one of the most important predictors for which candidates are recruited: the self-reporting of technical skills. In particular, I ask if the labor demand or labor supply side adjust their behavior with regard to self-reporting in ways that could increase the representation of women among software engineering recruits.

The data used in this investigation come from one particular recruiting platform used by tech companies for finding software engineering candidates. I augment this data by constructing measures of each individual's previous programming experience in programming languages based on actual computer source code they wrote and shared within the open-source software community. This novel dataset provides a means of comparing the propensity of male and female coders to self-report programming language skills after conditioning on actual previous experience in languages. Furthermore, by viewing both labor supply and labor demand behavior on the same platform, I can investigate if recruiters' actions on the platform are consistent with them adjusting for gender differences in job seekers' actions.

My analysis reveals a gender difference in self-reporting 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 9.84% less likely to self-report knowing that 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 12.37% less recruiter attention overall after controlling for all of 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 male candidates based on their visible profile attributes while overlooking some female candidates with similar actual previous experience. If, however, the gender difference in propen-

sity to self-report is primarily due to average differences in preferences for occupations involving coding or for receiving unsolicited messages from recruiters, 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.

6. References

Baldiga, Katherine. 2014. Gender Differences in Willingness to Guess. *Management Science* 60 (2): 434–448.

Bertrand, Sendhil Mullainathan, Hong Chung, Almudena Fern, and Mary Anne. 2004. Are Emily and Greg More Employable Than Lakisha and Jamal? A Field Experiment on Labor Market Discrimination. *The American Economic Review* 94 (4): 991–1013.

Beyer, Sylvia, Kristina Rynes, Julie Perrault, Kelly Hay, and Susan Haller. 2003. Gender Differences in Computer Science Students. *Proceedings of the 34th SIGCSE technical symposium on Computer science education*: 49-53.

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

Bursztyn, Leonardo, Thomas Fujiwara, and Amanda Pallais. 2017. 'Acting Wife' Marriage Market Incentives and Labor Market Investments. *American Economic Review* 107 (11): 3288–3319.

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

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

Foschi, Martha. 2000. Double Standards for Competence: Theory and Research. *Annual Review of Sociology* 26: 21–42.

Goldin, Claudia, and Cecilia Rouse. 1997. Orchestrating Impartiality: The Impact of 'Blind' Auditions on Female Musicians. *The American Economic Review* 90 (4): 715–41.

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

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

Heckman, J. 1978. Dummy endogenous variables in a simultaneous equation system. *Econometrica* 46 (4): 931–959.

Mobius, Markus M., Muriel Niederle, Paul Niehaus, and Tanya S. Rosenblat. 2011. Managing Self-Confidence: Theory and Experimental Evidence. *NBER Working Paper Series* 17014.

Niederle, Muriel and Lise Vesterlund. 2007. Do Women Shy Away from Competition? Do Men Compete Too Much? *The Quarterly Journal of Economics*, 122 (3): 1067–1101.

Oyer, Paul, and Scott Schaefer. 2011. Personnel Economics: Hiring and Incentives. *Handbook of Labor Economics* 4b: 1769-1823.

Rynes, Sara L., Amy E. Colbert, and Kenneth G. Brown. 2002. HR Professionals' Beliefs about Effective Human Resource Practices: Correspondence between Research and Practice. *Human Resource Management* 41 (2): 149–74.

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

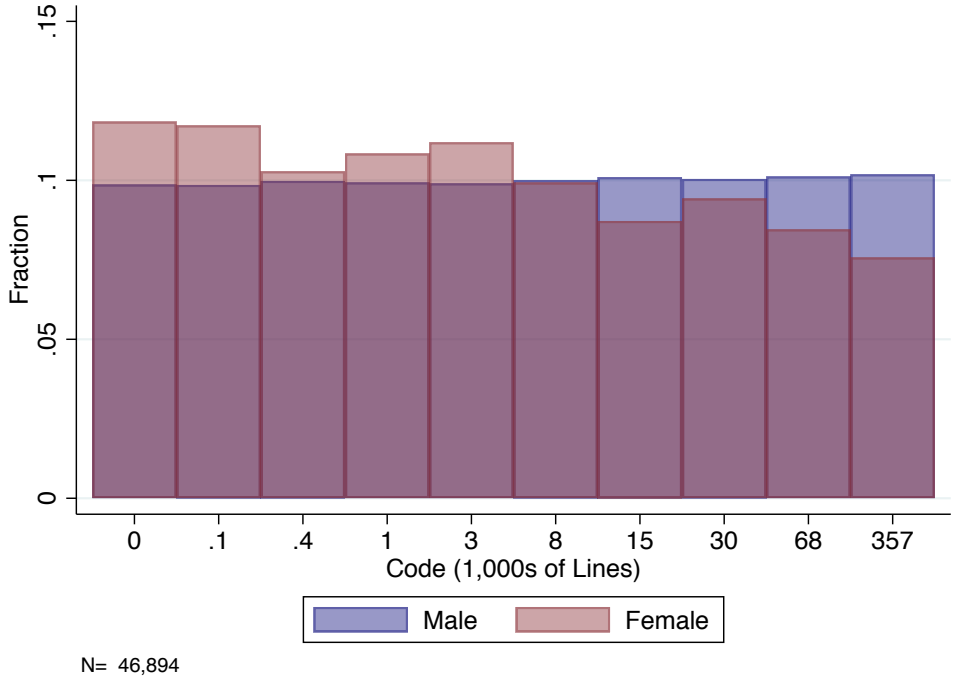
Steele, Claude M. 1997. A Threat in the Air: How Stereotypes Shape Intellectual Identity and Performance. *American Psychologist* 52 (6): 613–629.

Tambe, Prasanna, Xuan Ye, and Peter Cappelli. 2017. Paying To Program? Engineering Brand And High-Tech Wages. Working Paper.

Terrell, Josh, Andrew Kofink, Justin Middleton, Clarissa Rainear, Emerson Murphy-Hill, Chris Parnin, Jon Stallings. 2017. Gender Differences and Bias in Open Source: Pull Request Acceptance of Women Versus Men. *PeerJ Computer Science* 3 (e111).

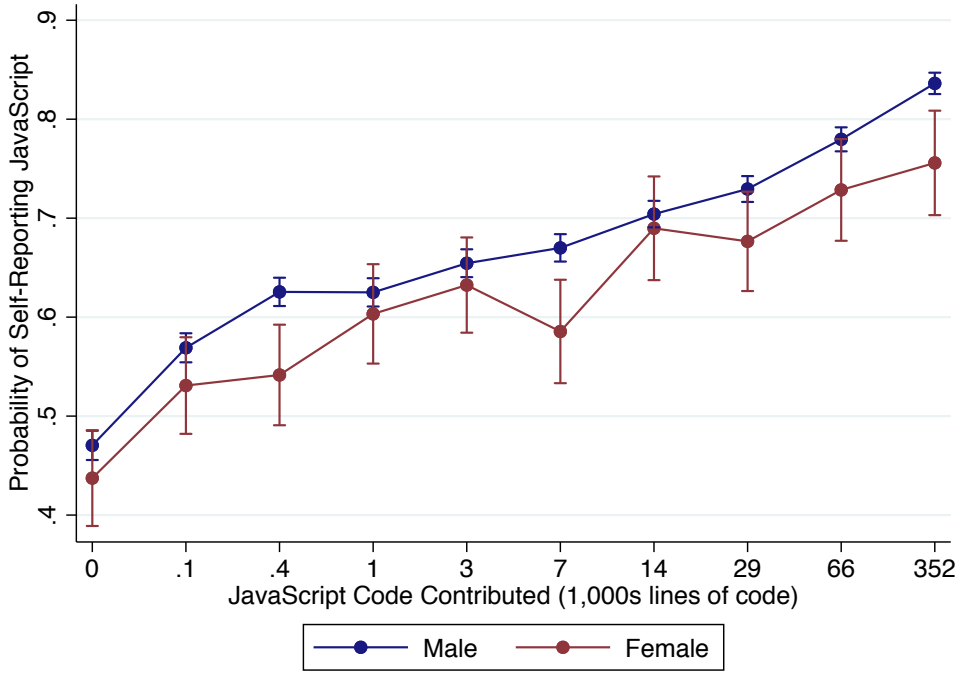
7. Figures

Figure 1: Distribution of Lines of Code Contributed to Open Source in JavaScript



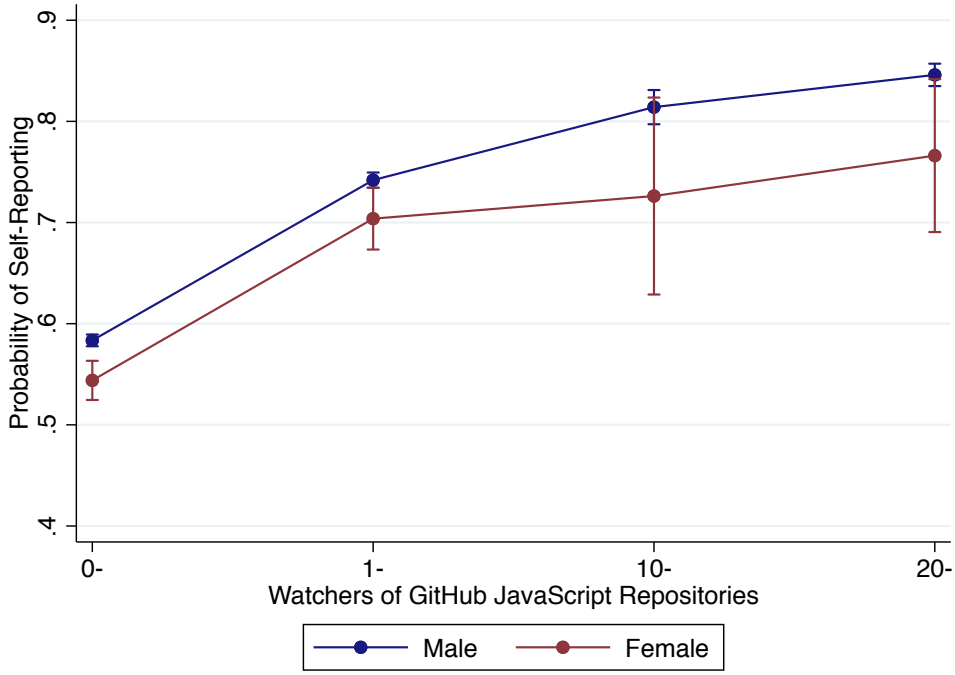
Note: The above figure shows distribution of the number of lines of JavaScript code uploaded to open source amongst candidates with at least one line of code and at least one self-reported skill on their resume. The candidates are grouped into deciles of code contributed. The average lines of code contributed for each decile is shown on the horizontal axis. The blue distribution represents the distribution of male candidates, while the red distribution is that of the female candidates. Because there are four times as many male candidates than female candidates, and because the deciles are constructed using all candidates, the male distribution should be close to uniform. The number of lines of code is adjusted from the raw numbers in order to exclude copied or "forked" code from other open source contributors.

Figure 2: Probability JavaScript is Listed in the Self-Reported Skills Section of the Profiles of JavaScript Open Source Contributing Candidates



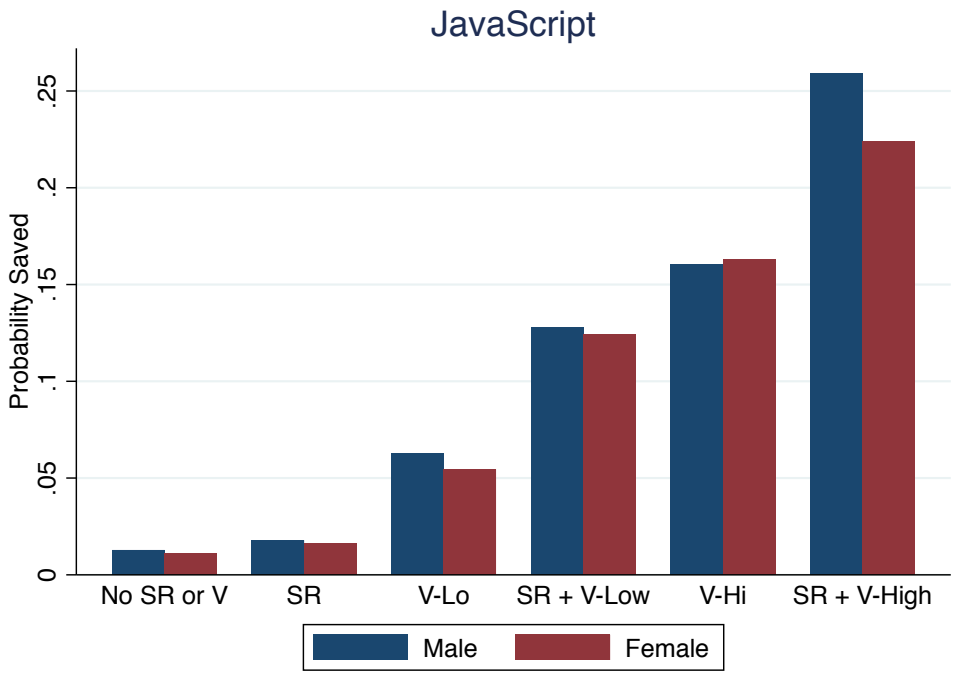
Note: Candidates with at least one line of open source code in JavaScript and at least one self-reported skill on their resume are grouped into deciles according to the total lines of code in JavaScript they have uploaded in JavaScript over their lifetime. The mean probability that male and female coders within each of these deciles 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 deciles.

Figure 3: Probability JavaScript is Listed in the Self-Reported Skills Section of the Profiles Grouped by Number of "Watchers" of JavaScript Repositories



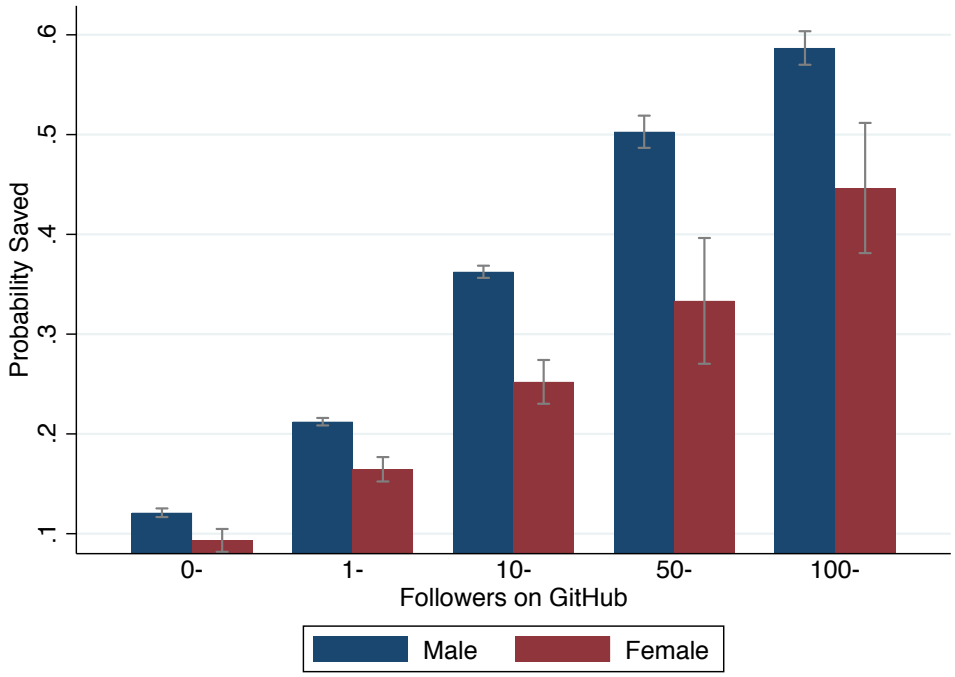
Note: The above graph uses data from candidates with at least one line of open source code in JavaScript and at least one self-reported skill on their resume. These candidates are grouped according to the total number of "watchers" that their JavaScript open source repositories hosted on GitHub have. 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.

Figure 4: Predicted Probability Profile Saved by Gender and Display of JavaScript



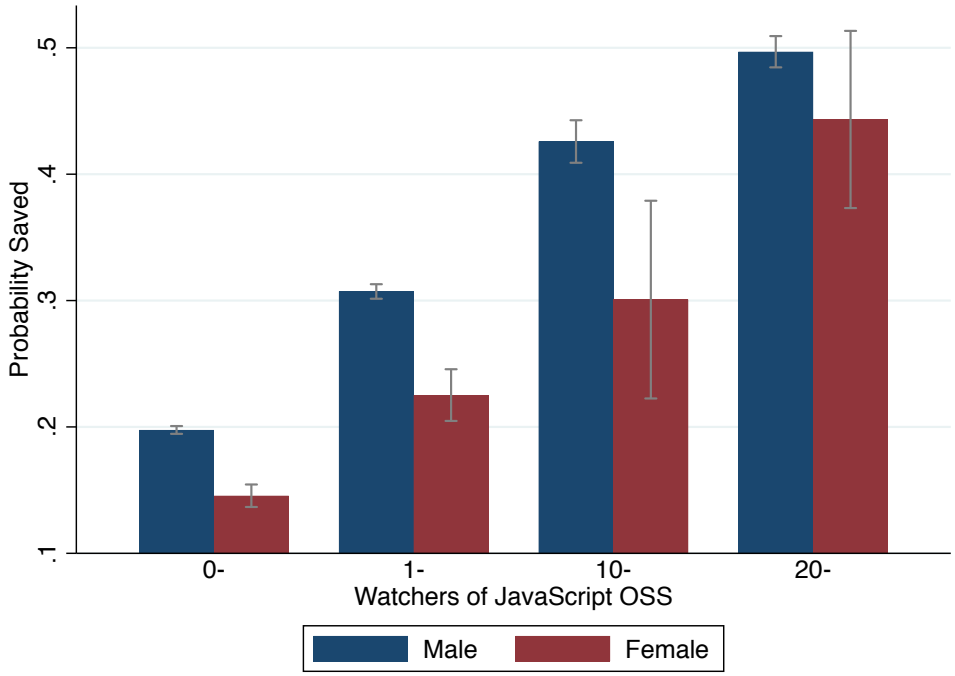
Note: Equation 4 is estimated on the Profiles data. This equation predicts the probability that a candidate is saved on all profile attributes as described in Table 1. In addition, profile attributes about educational credentials, work history, and skills are interacted with the gender of the candidate. The predicted probability that each candidate is saved is then computed at the mean value of all profile attributes. The above figure displays the predicted probability that a candidate profile is saved when JavaScript is displayed in the ways shown on the horizontal axis. In addition, I show the predicted probability if the gender of the candidate is either male or female.

Figure 5: Probability a Profiles Grouped by Number of GitHub Followers are Saved



Note: Among candidates who have contributed JavaScript code to open source, I collect the number of "followers" they have on GitHub. A "follower" is someone who receives updates about the coding work of another programmer. Candidates are group according to the number of "followers" they have on the platform Github. The vertical axis shows the probability that a candidate with a particular number of followers is saved. The horizontal axis show a range for the number of followers within each group. The probability that candidates are saved is computed for each group. Confidence intervals for the mean of each are shown. These represent the 95% confidence interval of the means.

Figure 6: Probability Profiles Grouped by Number of Open Source "Watchers" to JavaScript Projects are Saved



Note: Candidates who have at least one line of open source code in JavaScript posted to GitHub are grouped according to the number of "watchers" to their JavaScript repositories. A "watcher" is someone who subscribes to updates about code changes in these particular open source software projects. "Watchers" are typically people who use the code in the open source project or who find the project interest. The horizontal axis shows the range in the number of "watchers" that candidates within that group have to their projects. The probability that candidates are saved is computed for each group. The 95% confidence interval of the mean for that group is displayed.

8. Tables

Table 1: Variables in the Profiles Dataset

Variable	Description
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 city 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: 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.

Table 2: Mean Value of Attributes on Profiles

	Males	Females
Saved	0.02	0.01
Saves	0.03	0.01
BA Year	2000.65	2001.92
BA in CS	0.19	0.18
Has Masters	0.13	0.15
Has Ph.D.	0.02	0.02
Currently Coder	0.39	0.27
Overall Candidate Score #1	2.15	2.06
Overall Candidate Score #2	2.59	2.56
SR Languages	0.58	0.43
SR Skills	12.26	14.33
Verified Languages	0.23	0.06
Lists BA Degree	0.50	0.54
Lists SR Skills	0.54	0.64
SR Programming	0.01	0.00
SR Software Dev./Engineering	0.02	0.00
SR Web Apps.	0.01	0.00
SR Git/SVN	0.01	0.00
SR REST	0.01	0.00
SR Web Dev.	0.01	0.00
SR Agile	0.01	0.00
SR JavaScript	0.05	0.03
JavaScript in Work Descriptions	0.03	0.02
N	3,116,942	810,208

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 above variables are described in detail in Table 1

Table 3: Fraction of Profiles Displaying JavaScript as Verified and Self-Reported Among those with JavaScript on their Profile

	Verified - High	Verified - Low	Not Verified
Self-Reported	4.20%	9.32%	56.19%
Not Self-Reported	3.44%	26.84%	
% Self-Reporting	55.01%	25.78%	

Note: 263,422 profiles have JavaScript listed as either "Self-Reported" or "Verified" or both. Each cell shows the fraction of those 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 4: Probability a Profile is Saved Conditional on JavaScript Listed in the Self-Reported Skills and "Verified" Languages Lists

	Verified - High	Verified - Low	Not Verified
Self-Reported	0.60	0.33	0.06
Not Self-Reported	0.36	0.14	0.01

Note: Each cell shows the empirical probability that a profile in that cell was saved by at least one recruiter between 2014 and 2016. Profiles are placed into cells according to which lists on the profile the programming language JavaScript appeared. For example, the top left cell contains profiles where the language appeared in both the "Self-Reported Skills" list and the "Verified Languages" list with a score indicating "High Experience."

Table 5: Means of Attributes on Profiles of Open Source Contributors

	Males	Females
Saved	0.20	0.14
Saves	0.37	0.22
BA Year	2006.20	2007.96
BA in CS	0.28	0.26
Has Masters	0.13	0.17
Has Ph.D.	0.03	0.04
Currently Coder	0.46	0.38
Overall Candidate Score #1	3.10	2.95
Overall Candidate Score #2	3.02	3.20
SR Languages	2.17	1.86
SR Skills	11.10	10.09
Verified Languages	4.18	3.64
Lists BA Degree	0.49	0.56
Lists SR Skills	0.42	0.42
SR Programming	0.08	0.07
SR Software Dev./Engineering	0.18	0.12
SR Web Apps.	0.11	0.07
SR Git/SVN	0.16	0.11
SR REST	0.07	0.03
SR Web Dev.	0.13	0.11
SR Agile	0.13	0.09
SR JavaScript	0.25	0.22
JavaScript in Work Descriptions	0.11	0.12
N	168,445	13,858

Note: An observation is a profile. The means of the attributes of profiles with male and female names are shown in the left and right columns respectively. All variables are documented in detail in Table 1.

Table 6: Summary Statistics of Profile-Language Pairs

	Mean	St.Dev.	P10	P25	P50	P75	P90
Code (10k lines) C#	0.77	3.71	0	0	0	0	1
Code (10k lines) Javascript	4.86	17.77	0	0	1	3	11
Code (10k lines) Perl	0.25	2.25	0	0	0	0	0
Code (10k lines) Php	2.92	13.02	0	0	0	1	5
Code (10k lines) Python	0.69	3.85	0	0	0	0	1
Code (10k lines) Ruby	0.80	5.35	0	0	0	0	1
Code (10k lines) Sql	0.40	4.79	0	0	0	0	0
Days C#	97.44	475.23	0	0	0	15	171
Days Javascript	255.07	1,332.53	0	0	8	116	551
Days Perl	231.71	4,198.43	0	0	0	2	87
Days Php	161.81	1,102.76	0	0	1	25	267
Days Python	232.98	1,140.41	0	0	2	60	476
Days Ruby	264.72	1,655.26	0	0	3	62	471
Days Sql	15.99	139.63	0	0	0	1	14
QA Answers C#	6.33	67.51	0	0	0	0	5
QA Answers Javascript	1.47	26.26	0	0	0	0	1
QA Answers Perl	0.70	17.84	0	0	0	0	0
QA Answers Php	2.22	28.17	0	0	0	0	1
QA Answers Python	1.50	31.36	0	0	0	0	0
QA Answers Ruby	2.09	26.68	0	0	0	0	1
QA Answers Sql	0.50	11.78	0	0	0	0	0
Profile-Language Pairs	149,141						
Fraction Profile-Language Pairs Female	0.06						
Profiles	59,394						
Fraction Profiles Female	0.07						

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 7: Linear Probability Model Predicting if a Profile is Saved

	Saved
BA in CS=1	0.008***
	(0.000)
BA School in Top 10 for CS=1	-0.000
	(0.001)
BA in CS=1 × BA School in Top 10 for CS=1	0.013***
	(0.001)
Current Employer in Top 10 for Tech	0.049***
	(0.002)
SR Public Speaking	0.000*
	(0.000)
SR Team Building	0.001***
	(0.000)
SR Node.js	0.057***
	(0.003)
SR Agile	0.008***
	(0.001)
SR Git/SVN	0.043***
	(0.002)
SR Machine Learning	0.012***
	(0.002)
Javascript SR, No V	0.005***
	(0.001)
Javascript V-Lo, No SR	0.049***
	(0.001)
Javascript SR and V-Lo	0.115***
	(0.003)
Javascript V-Hi, No SR	0.148***
	(0.005)
Javascript SR and V-Hi	0.245***
	(0.005)
Controls	Yes
N	3,927,083
Prob. Saved for Not Verified Profiles	0.009
Prob. Saved for Verified - Low Profiles	0.189
Prob. Saved for Verified - High Profiles	0.493
R^2	0.222

Standard errors in parentheses

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

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 1 and include all of the candidate attributes visible to recruiters. Robust standard errors are shown in parentheses.

Table 8: 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	0.063***	0.086*
	(0.010)	(0.047)
Group Fixed Effect	Yes	Yes
N	12,336	1,169
N Groups	1,624	591
Dep. Mean	0.08	0.34
R^2	0.007	0.008

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 9: Predicted Probability a Programmer Lists a Programming Language in the Self-Reported Skills on their Resume

	Self-Reported		
	(1) Code (10k)	(2) Year	(3) SO Answers
Female	-0.060***	-0.063***	-0.061***
	(0.008)	(0.008)	(0.007)
Experience	0.003***	0.010***	0.001***
	(0.000)	(0.002)	(0.000)
Female x Experience	0.001	0.030***	0.004***
	(0.001)	(0.007)	(0.001)
Edu. Controls	Yes	Yes	Yes
N	82,779	82,779	82,779
N Programmers	52,811	52,811	52,811
Dependent Mean	0.61	0.61	0.61
R^2	0.01	0.01	0.00

Standard errors in parentheses

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

Note: The above table shows the estimated coefficients from an OLS regression on pairs of candidates with programming languages that they have uploaded at least one line of open source code in. Only candidates who also list at least one self-reported skill are included in the regression. The dependent variable is an indicator for if the candidate from the candidate-language pair self-reported the language within the self-reported skills on their resume. Female is defined as an indicator for the candidate having a name associated with the female gender. Code represents the number of lines of code uploaded to open source in that programming language divided by 10,000. Finally an interaction of the female indicator and the number of lines of code is included in the regression. Edu. Controls included the year of completion for the first bachelor's degree. This is included in the regression as an indicator for each year as well as an indicator for the bachelors year being missing. Standard errors are clustered at the person level.

Table 10: Predicted Probability a Programmer Lists a Programming Language in the Self-Reported Skills on their Resume Among Those Who Self-Report Skills Associated with Software Engineering

	(1) Code (10k)	(2) Year	(3) SO Answers
Female	-0.031***	-0.033***	-0.031***
	(0.008)	(0.008)	(0.008)
Experience	0.003***	0.009***	0.001***
	(0.000)	(0.002)	(0.000)
Female x Experience	0.001	0.023***	0.003***
	(0.001)	(0.006)	(0.001)
Edu. Controls	Yes	Yes	Yes
N	72,287	72,287	72,287
Dependent Mean	0.66	0.66	0.66
R^2	0.01	0.01	0.01

Standard errors in parentheses

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

Note: The above table shows the estimated coefficients from an OLS regression on pairs of candidates with programming languages that they have uploaded at least one line of open source code in. Only candidates who also list at least one self-reported skill are included in the regression. The dependent variable is an indicator for if the candidate from the candidate-language pair self-reported the language within the self-reported skills on their resume. Female is defined as an indicator for the candidate having a name associated with the female gender. Code represents the number of lines of code uploaded to open source in that programming language divided by 10,000. Finally an interaction of the female indicator and the number of lines of code is included in the regression. Edu. Controls included the year of completion for the first bachelor's degree. This is included in the regression as an indicator for each year as well as an indicator for the bachelors year being missing. Standard errors are clustered at the person level.

Table 11: Predicted Probability a Programmer Lists a Programming Language in the Self-Reported Skills on their Resume Using Bachelor's Degree Graduates in Computer Science Between 2010 and 2015

	Self-Reported		
	(1)	(2)	(3)
	Code (10k)	Year	SO Answers
Female	-0.032*	-0.031*	-0.036**
	(0.017)	(0.017)	(0.017)
Experience	0.005***	0.030***	0.003***
	(0.001)	(0.003)	(0.001)
Female x Experience	-0.002	0.003	0.001
	(0.002)	(0.015)	(0.001)
Edu. Controls	Yes	Yes	Yes
N	13,862	13,862	13,862
N Programmers	8,691	8,691	8,691
Dependent Mean	0.64	0.64	0.64
R^2	0.01	0.01	0.00

Standard errors in parentheses

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

Note: The above table shows the estimated coefficients from an OLS regression on pairs of candidates with programming languages that they have uploaded at least one line of open source code in. Only candidates who also list at least one self-reported skill are included in the regression. The dependent variable is an indicator for if the candidate from the candidate-language pair self-reported the language within the self-reported skills on their resume. Female is defined as an indicator for the candidate having a name associated with the female gender. Code represents the number of lines of code uploaded to open source in that programming language divided by 10,000. Finally an interaction of the female indicator and the number of lines of code is included in the regression. Edu. Controls included the year of completion for the first bachelor's degree. This is included in the regression as an indicator for each year as well as an indicator for the bachelors year being missing. Standard errors are clustered at the person level. Must have graduated with a bachelor's in Computer Science between 2010 and 2015

Table 12: Predicted Probability a Profile is Saved Using Profile Attributes and Interactions of Self-Reporting

	Saved
Female	-0.002***
	(0.000)
SR Software Dev./Engineering=1	0.006***
	(0.000)
Female × SR Software Dev./Engineering=1	-0.001*
	(0.001)
SR Web Design=1	0.001***
	(0.000)
Female × SR Web Design=1	0.002***
	(0.000)
SR Web Dev.=1	0.000
	(0.001)
Female × SR Web Dev.=1	0.001
	(0.001)
SR Web Apps.=1	0.010***
	(0.001)
Female × SR Web Apps.=1	-0.000
	(0.002)
SR Agile=1	0.011***
	(0.001)
Female × SR Agile=1	-0.004***
	(0.001)
SR Machine Learning=1	0.014***
	(0.002)
Female × SR Machine Learning=1	-0.002
	(0.005)
SR Git/SVN=1	0.044***
	(0.002)
Female × SR Git/SVN=1	-0.011**
	(0.004)
SR REST=1	0.035***

	(0.002)
Female × SR REST=1	0.016**
	(0.007)
SR PostgresQL=1	0.016***
	(0.003)
Female × SR PostgresQL=1	-0.004
	(0.009)
SR MongoDB=1	0.025***
	(0.004)
Female × SR MongoDB=1	-0.009
	(0.012)
SR MySQL=1	0.017***
	(0.001)
Female × SR MySQL=1	0.000
	(0.003)
SR Node.js=1	0.059***
	(0.003)
Female × SR Node.js=1	0.010
	(0.013)
SR Hadoop=1	0.050***
	(0.003)
Female × SR Hadoop=1	-0.009
	(0.009)
SR Hive=1	0.036***
	(0.008)
Female × SR Hive=1	0.029
	(0.022)
SR Redis=1	0.051***
	(0.007)
Female × SR Redis=1	-0.047
	(0.029)
SR Project Management=1	-0.002***
	(0.000)

Female × SR Project Management=1	0.001***
	(0.000)
SR Management=1	-0.001***
	(0.000)
Female × SR Management=1	0.000
	(0.000)
SR Customer Relations=1	-0.001***
	(0.000)
Female × SR Customer Relations=1	0.001***
	(0.000)
SR Leadership=1	-0.001***
	(0.000)
Female × SR Leadership=1	0.001***
	(0.000)
SR Program Management=1	-0.002***
	(0.000)
Female × SR Program Management=1	0.001***
	(0.000)
SR Social Media=1	-0.001***
	(0.000)
Female × SR Social Media=1	0.000
	(0.000)
SR Public Speaking=1	0.000**
	(0.000)
Female × SR Public Speaking=1	0.001*
	(0.000)
SR Team Building=1	0.001***
	(0.000)
Female × SR Team Building=1	-0.000
	(0.000)
Javascript SR, No V	0.006
	(0.001)
Female × Javascript SR, No V	-0.001

	(0.001)
Javascript V-Lo, No SR	0.048***
	(0.001)
Javascript SR and V-Lo	0.112***
	(0.003)
Female × Javascript SR and V-Lo	-0.003
	(0.011)
Javascript V-Hi, No SR	0.147***
	(0.005)
Javascript SR and V-Hi	0.242***
	(0.005)
Female × Javascript SR and V-Hi	-0.035
	(0.023)
Controls	Yes
N	3,927,083
Prob. Saved	0.015
Prob. Saved for Not Verified Profiles	0.009
Prob. Saved for Verified - Low Profiles	0.189
Prob. Saved for Verified - High Profiles	0.493
R^2	0.223

Standard errors in parentheses
* $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

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... Robust standard errors are used.

Table 13: Predicted Probability a Profile is Saved Using All Profile Attributes and Interactions

	Saved
Female	0.001***
	(0.000)
Has Masters=1	-0.001***
	(0.000)
Female × Has Masters=1	-0.001***
	(0.000)
Has Ph.D.=1	-0.011***
	(0.000)
Female × Has Ph.D.=1	0.002***
	(0.001)
BA in CS=1	0.009***
	(0.000)
Female × BA in CS=1	-0.004***
	(0.000)
BA in CS=1 × BA School in Top 10 for CS=1	0.012***
	(0.002)
Female × BA in CS=1 × BA School in Top 10 for CS=1	-0.003
	(0.003)
Currently Coder=1	0.011***
	(0.000)
Female × Currently Coder=1	-0.004***
	(0.000)
Currently Coder=1 × Current Employer in Top 10 for Tech=1	0.026***
	(0.005)
Female × Currently Coder=1 × Current Employer in Top 10 for Tech=1	0.014
	(0.011)
Controls	Yes
N	3,927,083
Prob. Saved	0.015
Prob. Saved for Not Verified Profiles	0.009
Prob. Saved for Verified - Low Profiles	0.189
Prob. Saved for Verified - High Profiles	0.493
R^2	0.224

Standard errors in parentheses

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

Note: The above table shows a subset of the coefficients from estimating Equation 4 using OLS. The dependent variable is an indicator for whether or not a profile was saved by any recruiter between 2014 and 2016. As controls, I include all visible profile attributes as described in Table 1. In addition, the interaction of these variables with an indicator for if