Project Level Effects of Gender on Contribution Evaluation on Github

Pascal Brokmeier (B.Sc.)
Universität zu Köln, Germany
pbrokmei@smail.uni-koeln.de

Abstract

Distributed open source software development has largely turned to GitHub, a pull-based software development collaboration platform. Recent studies have deployed data science techniques on the large datasets available about millions of projects on Github. Some research has focused on pull request (PR) acceptance predictors and some evidence was found of sexual discrimination among members, more specifically in the probability of acceptance of pull requests by different genders. In this paper we analyzed the influence of gender on PR acceptance on a project level, comparing different popular projects regarding their discrimination factors. Several projects were identified that have significant differences between male and female PR acceptance rates.

1. Introduction

1.1. Background

Software development has adapted to the needs for distributed development through the concepts of social coding and pull based software development pushed by platforms like Github which provide a platform for some of the biggest Open Source Software (OSS) projects existing. Projects like rails, docker, angular, node or swift are publicly hosted with some of them having thousands of followers and contributors. Open Source software development has been described as meritocracies [1], however recent research has identified social factors to influence decisions of project managers. This holds also true for the acceptance of contributions by others through PRs. [2]. Inevitably, a social coding environment such as GitHub is accompanied with social interaction that influences the project progress.

1.2. Motivation

An obvious factor in social interaction is gender. Research has shown that women are being treated unequal in professional environments, receive promotions less likely than men and are less likely to be hired when compared to male competition [3, 4]. and more specifically OSS projects have been shown to exhibit sexist behavior. About 1.5% of the total number of members in communities of *'free/libre/open source software (F/LOSS)'* were determined to be female compared to 28% in proprietary software as determined in a 2005 report by the University of Cambridge [5]. More current research shows a percentage of about 9% female users on Github [6].

Hoogendoorn et al. found equal gender mix teams to perform better than male or female dominated teams in the context of business students. Increased mutual monitoring, a form of informal Clan Control [8], was found to be a strong beneficial factor in mixed gender teams.

Vasilescu et al. more specifically found gender diversity to have beneficial effects on OSS team performance. This creates an economic incentive for organizations to promote diversity in their teams and communities.

Tsay et al. found project managers to use social cues to evaluate contributions and Vasilescu et al. found almost half of the project members to be aware of other users gender. Consequently the effect of the perceived gender on this contribution process can be of interest in the ongoing debate of gender inequality. Current research performed by Terrell et al. has identified a significant difference between the acceptance rate of pull requests created by men and women. Women whose profile publicly display their gender have a 4.1% lower chance of their PR being accepted compared to mens and a 10% lower chance than those women that did not disclose their gender. This is especially interesting as this percentage only holds true for 'visible' women. Those that decided to withhold information about their gender in their profile have a higher chance of having their request accepted.

In their research, Terrell et al. have analyzed a variety of different topology factors on the dataset. They have analyzed their results for different biases most notably an extensive covariate analysis which showed no explanations for the bias towards men. They also analyzed different programming languages and their relation to the acceptance rates. They however did not compare gender-dependent acceptance rates on a project level, allowing project specific environment and culture to explain the discovered differences.

1.3. Research Question

These factors, the underrepresentation of women on Github, the observed sexist behavior within OSS communities as well as the social influences on decisions that were believed to be purely lead by meritocratic reasoning raise the following question:

Can project level differences in acceptance rates of PRs between genders be observed on Github?

1.4. Research Method

To answer the research question and due to the large amount of data accessible, data analytics techniques need to be applied. To ensure sufficient data is available per project, the biggest 100 projects on Github are selected [1].

This paper quantitatively analyzes differences of gender participation between projects. It does not try to determine the relevance of gender in comparison to other social cues in the process of deciding whether to merge or decline a PR. Instead, this work is looking for project level differences in gender dependent PR statistics.

The structure of the paper is as follows: First, some background on gender inequality research in professional settings and more specifically in OSS will be provided. Afterwards, general research on GitHub data is reviewed and the most recent research of gender influences on GitHub will be introduced. In the next chapter, the research method and data acquisition will be described to facilitate the reproducibility of this work. Subsequently the results will be presented and discussed. Finally limitations and ideas for further research complete the paper.

2. Background and Theory

Past research on online communities has analyzed effects of gender, tenure, network embeddedness and other social factors on individual participation, team performance and project success. [6, 10, 11]. Results show influences of social cues on peer performance evaluation as well as effects of network embeddedness to generally positively influence project success.

Gender has been a particular topic in past research as the ratio of male to female participants in OSS has historically always been low with surveys ranging from 1-5% in 2006 and recent surveys from 2013 showing results of about 10% female participants [6, 5]. This chapter will first touch on gender inequality in OSS and then summarize previous research performed on GitHub data.

2.1. Gender inequality in OSS

The Free/Libre and Open Source Policy Support (FLOSSPOLS) from 2006 has clearly described the under-representation of women in OSS. Studies between 2002 and 2006 reported low one digit percentages of women participating in OSS and these numbers have increased slightly in the last years [6]. Reasons for these numbers have been summarized in the FLOSSPOLS by Krieger and Leach. The authors report cultural and social arrangements such as a 'hacker ethic' and 'individuals as carriers of agency' as strong reasons for this inequality. The culture of OSS is being described as code-centric instead of product-centric, leading contributions to be evaluated as less relevant if they are not code-based. The culture revolves around online community inherent concepts such as 'flaming' which 'can be off-putting for newcomers [and] is particularly pronounced in the case of women, who [...] have a shorter history in computing' [5, p.6].

1. the projects were sorted based on their stars which is a good proxy for their popularity and therefore their community size

2.1.1. Gender and Tenure diversity in GitHub Teams. Aside from the ethical reasoning of investigating reasons for gender inequality, Vasilescu et al. have studied the influence of gender and tenure diversity on project productivity output in GitHub teams. Diversity is a significant predictor for team productivity. A survey of 4,500 GitHub users showed about half the users were aware of most of their teammates gender, making it the second most salient attribute after programming skills. This "contradicts earlier claims of obscurity of gender in OSS" [6]. Furthermore, there are differences in the subjective importance of diversity in teams. Some respondents did not consider diversity to be relevant as it is "more about the contributions to the code than the 'characteristics' of the person" while others characterize diversity as a "source of creativity". Overall, gender diversity is positively correlated with project productivity and highly significant. Finally, gender diversity negatively impacts turnover, helping projects to sustain their developer base.

2.2. Using Github as a research data set

With the changing of the tools used by developer communities in recent years, many older research used websites such as SourceForge or StackOverflow as their sources of data for empirical research [10, 11]. Newer research has moved towards GitHub as it is now the biggest source for publicly available software projects [6].

Bird et al. and later Kalliamvakou et al. have analyzed the data available on git based projects and GitHub Application Programming Interface (API) data to define a number of guidelines for researchers when approaching these types of data. While the research by Bird et al. focused on Git, the underlying versioning system of GitHub, Kalliamvakou et al. focused on GitHub specifically. The perils defined should be taken into account when analyzing GitHub based data and are therefore listed below:

1) A repository is not necessarily a project
2) Most projects have very few commits
3) Most projects are inactive

4) A large portion of repositories are not for software development
5) Two thirds of projects (71.6% of repositories) are personal
6) Only a fraction of projects use pull requests. And of those that use them, their use is very skewed
7) If the commits in a pull-request are reworked (in response to comments) GitHub records only the commits that are the result of the peer-review, not the original commits
8) Most pull requests appear as non-merged even if they are actually merged
9) Many active projects do not conduct all their software development in GitHub

Peril 1-3,5 are not applicable to this research as it focuses on the most popular projects.

Peril 4 is relevant. It confronts the fact that GitHub, although mainly considered to be a software development code sharing platform, actually hosts many different projects as well. As an example, the top 20 projects on GitHub include "free-programming-books" and "You-Dont-Know-JS", repositories containing books, "awesome", a repository containing a list of links to resources and "gitignore", a project including templates for a type of file often used on GitHub. If significant differences in PR gender/acceptance rates can be observed, the affected projects need to be controlled for being actual software development projects.

Peril 6 is relevant and results also need to be checked against this problem The most notable project is "linux" which is hosting a mirror of the linux kernel git repository. PRs are not accepted via GitHub and all PRs are closed without merge and the creators references to the propper hosting site.

Peril 7 is not relevant as we are not analyzing the contents of PRs but only their states.

Peril 8 and 9 are relevant. The fact that many PRs appear as non-merged although they actually were introduces a bias in the data and reduces the overall percentage of merged PRs. There is however no obvious reason to believe there are more PRs that have this false negative value for one or the other gender. Peril 9 shows a limitation of this work which is the narrow focus on PRs as an indicator for gender differences. It suggests a big part of software development occurs in many different areas, be it forums, chats or mailing lists [13].

The final conclusion of Kalliamvakou et al. is to carefully select the repositories analyzed and ensure the data acquired is actually suitable to answer the research question.

2.2.1. Research into PR acceptance factors. This work leans on the work of Tsay et al. who analyzed factors that may predict the acceptance of a PR as well as Terrell et al. who focused this analysis further towards gender differences.

Tsay et al. showed the number of comments as well as the age of a repository were the best predictors for determining whether a PR would be accepted or rejected. Also, the amount of previous interaction between the PR submitter and the project manager deciding to merge the PR influences the likelihood of a merge. In summary they observed a social factor in the evaluation processes of project managers and therefore gave reason to the research investigating detailed social factors in the space of OSS on GitHub.

Terrell et al. have continued this research path and investigated the influence of gender on the evaluation process. They used similar data sources as others and enriched those with gender information for users, a process repeated in this work. Based on this data they could correlate gender and PR acceptance statistics, finding womens' PR acceptance rates were higher but only for those women that did not reveal this gender in the space of GitHub. If they did so and their gender was clearly visible on the platform (through names or photos), their acceptance rate was lower than those of men. Due to the large number of PRs analyzed, statistical significance is easily achieved. Davison and Burke's meta-analysis on sex discrimination found an average Pearson correlation of $\rho = 0.07$ between gender and job selection which, compared with the results of Terrell et al. of $\rho = 0.02$ is higher. The results are therefore less impactful than the average sexual discrimination observed, yet they can still be interesting to further understand, especially in seemingly meritocratic environments. Also, a project level investigation might reveal a strong variance in this correlation, revealing projects with stronger correlations.

3. Methods and Data

To determine project level differences, three subtasks can be identified that need to be completed for the question to be answered adequately. Acquiring the data, preparing it and analyzing it. To retrieve and prepare the data, a NodeJS tool has been written that combines data from GHTorrent as well as using the Github API to acquire all necessary data. All data was locally stored in a NoSQL Mongo database and later transferred to comma separated values (csv) files that could easily be imported to Microsoft Excel for the final statistical calculations and validations.

3.1. Data acquisition

Fetching the data included first fetching the PRs for each repository. This was performed using the Github REST API, a public interface for programs and applications to interact with Github. Each batch of PRs was then processed and placed into a queue which processed the PRs one at a time. For each PR, the GitHub user that created it was queried from the GHTorrent database and cached locally since many PRs were created by similar users. The entire software to collect, analyze and store this data was written in NodeJS, a JavaScript (JS) based runtime environment that allows for easy handling of many parallel web requests as well as handling JavaScript Object Notation (JSON) documents easily. Both underlying database as well as the GitHub API consume and produce JSON documents and as such, a JS based technology was reasonable.

3.1.1. GHTorrent data set. Gousios created the GHTorrent project. It watches the event stream of Github and stores everything in two separate databases, a relational database and a document based database, namely a MongoDB database. Researchers can get access to these databases by donating a Github API key and perform queries which are more versatile than the Github API itself. It also allows for a higher number of queries per hour, which GitHub limits to 5000 every 60 minutes. This allowed for a much quicker determination of the genders as a request was required for each user profile and over 40,000 users were involved in the analyzed PRs.

3.2. Determining Gender

After the necessary data was collected the users profile was used to determine the gender. For each user a gender inference was attempted from their login name, their email address and their full name. Not every user has all three attributes added to their profile. 74% of all PRs were created by users with a full name added to their profile, allowing for a much higher success rate in inferring gender from profiles than by simply using the login name as was done in previous analyses.

3.2.1. Gendercomputer. To infer the gender, the `genderComputer` by Vasilescu et al. was used. This algorithm uses several heuristics such as the origin/country of the user as well as common name patterns and ultimately a name-gender dictionary to infer the gender from a given user. It has a reported success rate of about 32% [6] using profile names. In this study a higher success rate was achieved due to the availability of full names which can be added by users on Github and was done by 66% of the users that were analyzed.

3.2.2. Social network profile matching. Terrell et al. suggested an additional external verification of gender through the consultation of social network APIs such as Google+ to determine gender of those users that are not identifiable through the previous step. This work only used the `genderComputer` tool as it is simpler to user and offered already high success rates due to the now available data of real names of users. An additional verification with facebook would have been desirable as it is much more adopted than Google+ [15, 16]. However, the company does not offer an

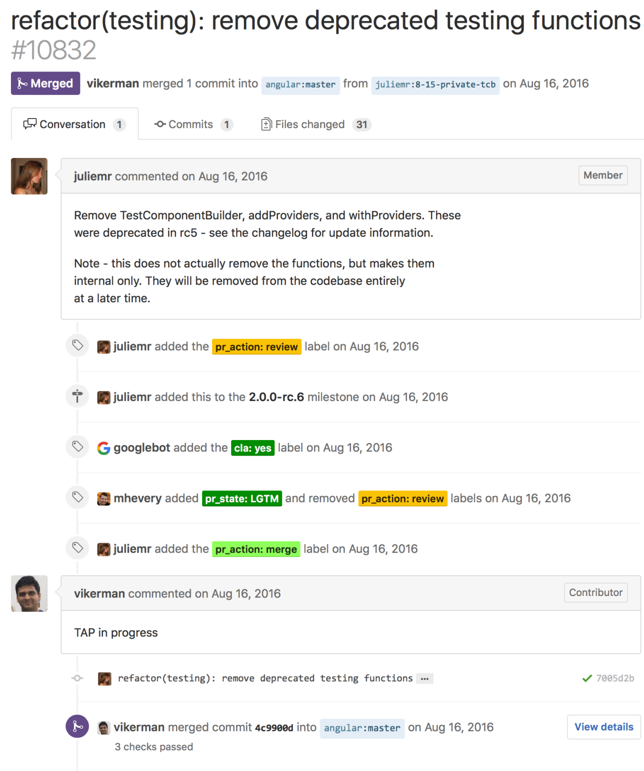

Figure 1. Screenshot of PR

API that allows for resolving profiles from email addresses. Hence the previously mentioned tool was the most efficient and effective approach to analyze all 43130 users that created the PRs.

3.3. Determining PR acceptance

To ensure higher precision of the merged status Gousios et al. suggest a manual approach for determining the ultimate merging decision for each merge. This approach ensures that all merged PRs are captured. This is due to the setup of Github, allowing for pull requests to either be merged through Github's own facilities but also through Git, the underlying technology of Github, using its native merge tools. The GitHub API provides a flag ”merged_at” flag which indicated those PRs that have been merged through the GitHub UI. Since the UI transfers information about gender more easily through profile pictures and linked profiles than a command line based tool, those PRs are actually of higher interest for this study. To rate the acceptance of a contribution, we therefore consider a merged PR that has been flagged so by the GitHub systems to be an accepted contribution and a closed but not merged PR to be a rejected contribution.

The entire processing chain is summarized in Figure 2. In the actual implementation, a parallel architecture was used as

Table 1. Sample of repo data

Attribute	**angular**	**docker**	**ghost**
PR Count	2369	15146	1738
PR merged count	720	12024	1323
PR declined count	1649	3122	415
Male PRs	1586	9445	867
Female PRs	161	980	280
unknown gender PRs	622	4721	591
male merged count	494	7664	640
male declined count	1092	1781	227
female merged count	66	820	256
female declined count	95	160	24

well as a database backed prioritized queue to better handle the large amount of PRs and repositories while allowing for pausing and restarting of the processing engine.[2]

After the data preparation, the analysis of the data can be performed. To analyze the differences between projects, all past PRs of each project are analyzed. For each PR, a tuple of id, repository, creator's gender and merge result is created.

4. Results

A total of 43,039 users and 183,249 PRs were analyzed. The `genderComputer` determined the gender of 63.23% of the total number of users, determining the gender from their provided real name, their username and then their email address in descending order of result prioritization. The success rate was higher than those reported by Vasilescu et al., most likely because we focused on PRs that were in the largest projects on GitHub. Users in these repositories have more complete profiles in comparison to the large long tail of small and empty or personal projects, with 66.15% of all users analyzed having entered their name to their profiles, although its an optional field. Of the gendered users, a total of 1985 or 7.27% were female which is slightly lower than the 9% reported by [10]. Women have created on average 4.18 and men 4.69 PRs.

Using the gender linked PRs and user profiles, summary statistics were created for each project. A sample of these results is shown in Table 1.

To analyze the results, the chi-squared test for statistical significance was used. This test offers the ability to determine correlation between two variables [18, p.102ff.]. In the study the correlation between the expected number of declined or merged pull requests for each gender and the actual number was to be determined.

To evaluate correlation and calculate statistical significance, the chi-squared pearson test requires the calculation of a test statistic χ^2 which is then compared to a chi-squared probability of a defined probability barrier (0.005 in this study). The formula for calculating the test statistic is visible in Equation 1.

$$\chi^2 = \sum_{i=1}^{r} \sum_{j=1}^{c} \frac{(O_{i,j} - E_{i,j})^2}{E_{i,j}} \tag{1}$$

In this study, both r and c were 2, causing $df = 1$ and $\alpha = 7.879$ with $\chi^2(p < 0.005, df = 1)$.

Out of the 100 projects analyzed, 53 offered usable results that complied with the rule of having a count of at least 5 in each of the fields used for the χ^2 test [18, p.104]. Of those 53 projects, 18 projects had results that were statistically significant with $\chi^2(p < 0.005, df = 1)$. The results of those 18 projects are summarized in Table 2.

Table 2. Results of statistically significant projects

Project	**Languages**	χ^2	$p - value$	n
docker	Go	28.803	2.46302E-06	15146
angular.js	JS	21.782	7.23989E-05	6570
swift	Swift	27.285	5.12939E-06	6699
node	JS	105.113	1.23554E-22	5721
elasticsearch	Java	10.378	0.0156108	4412
react	JS	35.257	1.07507E-07	4350
react-native	JS,Java,Obj-C	8.021	0.045586391	4518
django	Python	39.850	1.14627E-08	3427
atom	CoffeeScript	32.610	3.893E-07	2973
electron	C++	14.238	0.002598504	2714
angular	TypeScript	8.980	0.029554946	2369
tensorflow	C++,Python	11.875	0.00782457	2367
Ghost	JS	38.624	2.08483E-08	1738
discourse	Ruby,JS	8.341	0.039460895	1843
moment	JS	27.567	4.47637E-06	1234
Modernizr	JS	9.299	0.025571851	479
free-programming-books	None	21.220	9.47533E-05	414

$\alpha = 0.005$

$\chi^2_{(0.95;1)} = 7.879$

As Terrell et al. noted, simple statistical significance is not a good indicator for such results alone and therefore, those projects with a strong statistical significance have been controlled for the Pearson correlation. The results are summarized in Table 3. Negative values describe projects in which women are less likely to have their PR accepted and positive values describe projects in which women are more likely to have their PR accepted.

Referring back to the perils of mining GitHub defined by Kalliamvakou et al., the results need to be controlled for peril 4, 6 and 8. Peril 4 (usage of GitHub for non-software projects) is applicable to 'free-programming-books', the last project in Table 2 which is a information collection project using GitHub as a host.

2. The entire code of the processing chain can be found at https://github.com/pascalwhoop/github-gender-processing

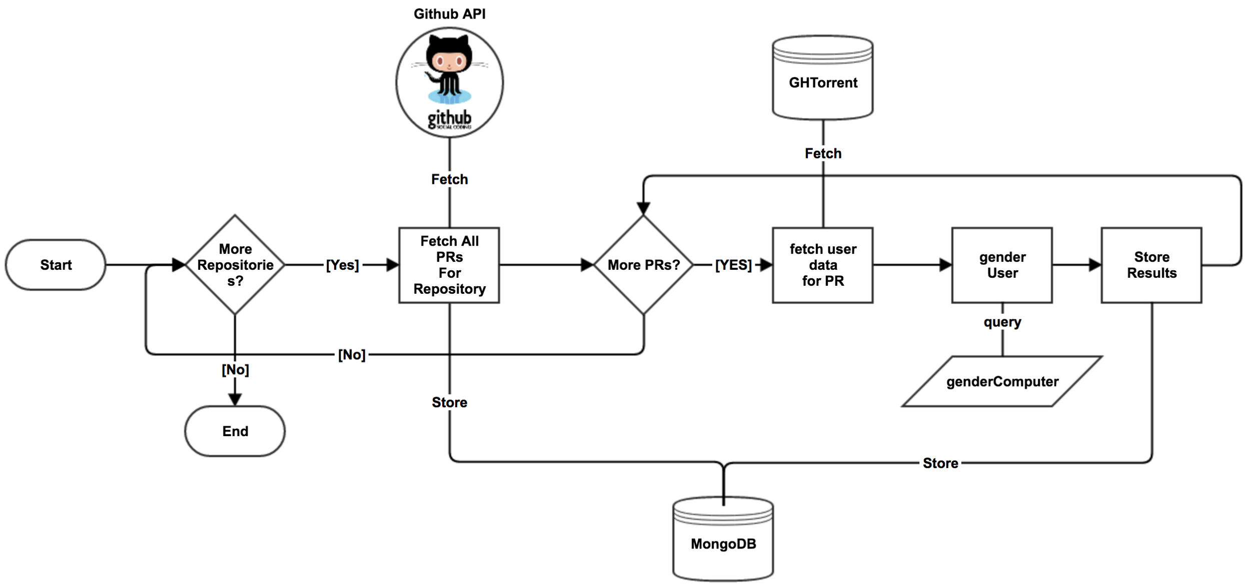

Figure 2. Flowchart describing the processing flow

Peril 6 (not all projects use PRs) is applicable to a few of the projects investigated, however non of the statistically significant results is affected. This is intuitive, since a large amount of PRs is required to ensure statistical significance which in itself is what needs to be controlled for.

Peril 8 (most PRs appear as non-merged but were using non GitHub methods) can be observed in several projects. For this the average merge rate per project was a good indicator to find those projects that tend to not use GitHubs PR system systematically. "angular.js", "node", and "jquery" all had a merging average of 15% or less with an average of 50.24% across all PRs. While jquery is a more mature project, predating GitHub, angular.js has been developed mainly on GitHub infrastructure which therefore suggests that this project has not used the PR concepts or has used other forms of including contributions (such as commit squashing or Git based git-request-pull commands). This has changed with "angular", which is the second major version of angular.js, which has a 30% PR merge average, a 23% increase. At this point, a correlation analysis between gender and type of PR acceptance technique was not performed, referring to the work of Terrell et al. who have not identified this as a possible reason.

Table 3. Pearson correlation for selected outliers

Project	$\rho_{merged,gender}$	n
moment	-0.217	1,234
node	-0.184	5,721
django	-0.173	3,427
atom	-0.169	2,973
react	-0.146	4,350
jquery	-0.138	2,169
electron	-0.121	2,714
angular.js	-0.074	6,570
angular	0.008	2,369
Ghost	0.112	1,738

4.1. Discussion

The quantitative results of the gender bias on GitHub have shown an interesting picture. While previous research has found a gender bias analyzing the overall PR acceptance rate of most repositories on GitHub, the results that focus on a project level difference have offered a more detailed picture. While most repositories only offered statistically insignificant results, 18 showed a significant bias and 7 projects have shown a Pearson correlation between merge result and gender of $\rho > 0.10$, exceeding those results observed by Davison and Burke with an average Pearson correlation of $\rho = 0.07$ between gender and job selection. The reasons for these biases can be various but ultimately require a qualitative inspection of the projects noted to investigate social differences between them. At first, the project "Ghost", which is the only project having a significant and strong bias towards female acceptance probability, stands out. It is also the project with the highest female participation rate (24% of all PRs were created by female users). This however is due to one active woman, accounting for 86.79% of all female PRs. All other projects either showed insignificant results or were biased towards male contributions.

This leads to the conclusion that the overall bias observed in previous research includes some examples of projects which tend to have a rather strong bias towards male contribution acceptance. Although the work of Vasilescu et al. regarding gender and tenure diversity and project success is hardly applicable to the 100 most successful projects on GitHub, the results still leave room for interpretation. The projects listed in Table 3 all, with the previously described exception of Ghost, show strong biases towards male contributions.

Code of Conduct? Moment X Node Y jquery X angular.js Y angular Y Ghost X

5. Conclusion

References

[1] W. Scacchi, "Free/open source software development: Recent research results and emerging opportunities," in *The 6th Joint Meeting on European Software Engineering Conference and the ACM SIGSOFT Symposium on the Foundations of Software Engineering: Companion Papers*, ser. ESEC-FSE companion '07. New York, NY, USA: ACM, 2007, pp. 459–468. [Online]. Available: http://doi.acm.org/10.1145/1295014.1295019

[2] J. Tsay, L. Dabbish, and J. Herbsleb, "Influence of social and technical factors for evaluating contribution in github," in *Proceedings of the 36th International Conference on Software Engineering*, ser. ICSE 2014. New York, NY, USA: ACM, 2014, pp. 356–366. [Online]. Available: http://doi.acm.org/10.1145/2568225.2568315

[3] H. K. Davison and M. J. Burke, "Sex discrimination in simulated employment contexts: A meta-analytic investigation," *Journal of Vocational Behavior*, vol. 56, no. 2, pp. 225 – 248, 2000. [Online]. Available: //www.sciencedirect.com/science/article/pii/S0001879199917116

[4] P. L. Roth, K. L. Purvis, and P. Bobko, "A meta-analysis of gender group differences for measures of job performance in field studies," *Journal of Management*, vol. 38, no. 2, pp. 719–739, 2012. [Online]. Available: http://dx.doi.org/10.1177/0149206310374774

[5] B. Krieger and J. Leach, "Free/libre and open source software: Policy support - gender: Integrated report of findings," University of Cambridge, Tech. Rep., 2006. [Online]. Available: http://flosspols.merit.unu.edu/deliverables/FLOSSPOLS-D16-Gender_Integrated_Report_of_Findings.pdf

[6] B. Vasilescu, D. Posnett, B. Ray, M. G. van den Brand, A. Serebrenik, P. Devanbu, and V. Filkov, "Gender and tenure diversity in github teams," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, ser. CHI '15. New York, NY, USA: ACM, 2015, pp. 3789–3798. [Online]. Available: http://doi.acm.org/10.1145/2702123.2702549

[7] S. Hoogendoorn, H. Oosterbeek, and M. van Praag, "The impact of gender diversity on the performance of business teams: Evidence from a field experiment," *Management Science*, vol. 59, no. 7, pp. 1514–1528, 2013. [Online]. Available: http://www.jstor.org/stable/23443867

[8] L. J. Kirsch, "The management of complex tasks in organizations: Controlling the systems development process," *Organization Science*, vol. 7, no. 1, pp. 1–21, 1996. [Online]. Available: http://dx.doi.org/10.1287/orsc.7.1.1

[9] J. Terrell, A. Kofink, J. Middleton, C. Rainear, E. Murphy-Hill, C. Parnin, and J. Stallings, "Gender differences and bias in open source: Pull request acceptance of women versus men," 2016.

[10] B. Vasilescu, A. Capiluppi, and A. Serebrenik, "Gender, representation and online participation: A quantitative study of stackoverflow," in *Social Informatics (SocialInformatics), 2012 International Conference on*, Dec 2012, pp. 332–338.

[11] R. Grewal, G. L. Lilien, and G. Mallapragada, "Location, location, location: How network embeddedness affects project success in open source systems," *Management Science*, vol. 52, no. 7, pp. 1043–1056, 2006. [Online]. Available: http://dx.doi.org/10.1287/mnsc.1060.0550

[12] C. Bird, P. C. Rigby, E. T. Barr, D. Hamilton, D. M. German, and P. Devanbu, "The promises and perils of mining git," in *Proceedings of the Sixth Working Conference on Mining Software Repositories*. Washington, DC, USA: IEEE Computer Society, May 2009. [Online]. Available: https://www.microsoft.com/en-us/research/publication/the-promises-and-perils-of-mining-git/

[13] E. Kalliamvakou, G. Gousios, K. Blincoe, L. Singer, D. M. German, and D. Damian, "The promises and perils of mining github," in *in MSR. ACM*, pp. 92–101.

[14] G. Gousios, "The ghtorrent dataset and tool suite," in *Proceedings of the 10th Working Conference on Mining Software Repositories*, ser. MSR '13. Piscataway, NJ, USA: IEEE Press, 2013, pp. 233–236. [Online]. Available: http://dl.acm.org/citation.cfm?id=2487085.2487132

[15]

[16]

[17] G. Gousios, M. Pinzger, and A. v. Deursen, "An exploratory study of the pull-based software

development model," in *Proceedings of the 36th International Conference on Software Engineering*, ser. ICSE 2014. New York, NY, USA: ACM, 2014, pp. 345–355. [Online]. Available: http://doi.acm.org/10.1145/2568225.2568260

[18] R. Hellbrück, "Angewandte statistik mit r," 2016.