Empirical Software Engineering (2025) 30:123
https://doi.org/10.1007/s10664-025-10653-x

Is perceived gender related to contributions and standing in open-source software projects?

Christian Hechtl[1] · **Mitchell Joblin**[1] · **Sven Apel**[1]

Accepted: 30 March 2025

Abstract

To date, the percentage of female developers that actively contribute to open-source software (OSS) projects is less than 10%. In recent years, researchers started searching for reasons for this imbalance. A question that arises in this space is how the (perceived) gender of a developer influences their contributions and standing in the organizational hierarchy of a project. Addressing this question, we have analyzed 20 popular OSS projects on GitHub. We found that the (perceived) gender of developers has only a negligible association with their project contributions (e.g., number of pull requests). In the same vein, women and men take similar positions in the organizational hierarchy, except for the top 10%, where men are still over-represented. So, while our results show a certain degree of gender balance with regard to contributions and standing, the leadership positions of the projects are still male-dominated. This suggests that further countermeasures against gender imbalance shall be directed toward the top of the organizational hierarchy.

1 Introduction

Historically, women have been underrepresented in STEM (science, technology, engineering, and mathematics) disciplines (Nimmesgern 2016). In the last two decades, there has been a movement to make these male-dominated fields more diverse, in particular, in terms of gender diversity. Although seeing the first results of that movement, the IT landscape is still

Communicated by: Meiyappan Nagappan

Mitchell Joblin work was done prior to joining Amazon.

✉ Christian Hechtl
hechtl@cs.uni-saarland.de

Mitchell Joblin
joblin.m@gmail.com

Sven Apel
apel@cs.uni-saarland.de

1 Saarland Informatics Campus, Saarland University, Saarbruecken, Saarland, Germany

Published online: 07 June 2025

male dominated. Big tech companies such as Facebook[1], Google[2], and Apple[3] report about 25% of their tech positions to be held by women.

A substantial part of the IT landscape are open-source software (OSS) projects (Robles et al. 2019; Riehle 2019). Unlike in closed-source in-house projects, there is no mandated process or central control structure and no curated and reliable information on the people who work on the projects, which makes even a diversity report next to impossible, not to speak of applying measures for increasing diversity. Still, since OSS projects gain more and more momentum, it is important to assess gender diversity in these projects. In this vein, studies have shown that gender-diverse teams are usually more productive and show better performance than teams that are dominated by men or women (Vasilescu et al. 2015; Hoogendoorn et al. 2013). So, it is in the very interest of OSS projects to attain and maintain diversity.

In recent years there has been a surge in research about this topic (Padala et al. 2022; Prana et al. 2021; Bosu and Sultana 2019; Vasilescu et al. 2015; Frluckaj et al. 2022). While providing interesting insights, existing studies fall short of incorporating the organizational structure that underlies an OSS project, as we will discuss. In this study, we analyze how men and women are positioned in the organizational hierarchy and, thus, how important they are for a project. We measure this through multiple social-network analysis methods. Ultimately, we operationalize the standing in the social hierarchy with *developer coreness*, which is a measure of how strongly embedded a developer is in the community of our subject projects. In contrast to previous work, we define developer coreness as a continuous quantity arising from the underlying developer network structure. With this measure equipped, we pose the following research questions:

RQ$_1$ *Is there a difference between men and women in terms of general contribution statistics in OSS projects?*

RQ$_2$ *Do the overall coreness values and the coreness distributions differ for men and women in OSS projects?*

For studying gender diversity in OSS projects, it is desirable to find a means to classify developers into men and women just based on repository data. This way, we can *automatically* compare men and women developers in terms of collaboration statistics, the role, and position in the organizational hierarchy of the projects. As, typically, we do not know the developers, we can only base the classification on the developers' activity data, such as names used in commits, developer profiles on GitHub, and the communication via issues. There are different techniques for this, including face-to-gender inference (Lu et al. 2006; Moghaddam and Yang 2002), name-to-gender inference (Santamaría and Mihaljević 2018; Mislove et al. 2011), and classification based on the writing style of developers (Argamon et al. 2003). Of these techniques, name-to-gender inference is best suited for our research based on the data available to us. Since it is a widely used method for gender inference, we gain the benefit of comparability with previous studies. With this we work on the assumption that names of developers are usually perceived as either male or female. We do not classify the developers themselves into a binary gender.

[1] https://about.fb.com/news/2022/07/metas-diversity-report-2022/
(accessed: 2024.01.10)

[2] https://about.google/belonging/diversity-annual-report/2023/
(accessed: 2024.01.10)

[3] https://www.apple.com/diversity
(accessed: 2024.01.10)

For the classification of developers and all other relevant statistics of OSS projects, we use data from public GIT repositories and GITHUB archives of projects. Through GIT, we have access to the commit history; through GITHUB, we have access to the communication and contribution records such as pull requests and issues (Gousios and Spinellis 2017; Kalliamvakou et al. 2014). To compare the role and impact of male and female developers, we investigate some basic statistics of OSS projects. Among these are the number of pull requests that were posted by men and women, the number of issues and issue comments, and the number of commits. Subsequently, we build developer networks based on technical artifacts (commits) (Joblin et al. 2015) and communication records (issues) to determine developer coreness (Joblin et al. 2017a). Then, we compare the coreness of men and women to determine whether there are any differences. By using socio-technical developer networks to examine the organizational hierarchy of OSS projects, we build upon the work of previous studies that have shown that analyzing these networks can provide a richer and more nuanced view of the organizational structure of OSS projects (Joblin et al. 2017a, 2015; Joblin and Apel 2022). With this, we aim at providing a more detailed and novel perspective on the differences of the standing of men and women in the organizational hierarchies of OSS projects. Furthermore, by using developer coreness as a continuous measure of the standing in the organizational hierarchy, instead of a binary core/periphery classification, we are able to provide an even more fine-grained view of the differences between men and women in OSS projects.

To answer our research questions, we use the data of 20 popular OSS projects: ANGULAR, ATOM, BOOTSTRAP, COOKIECUTTER, DBATOOLS, DENO, ELECTRON, GHOST, KERAS, MOBY, NEXT.JS, NEXTCLOUD, REACT, REDUX, REVEAL.JS, TENSORFLOW, THREE.JS, TYPESCRIPT, VS CODE, and VUE. After collecting data and preparing our analyses, we use statistical tests to compare male and female developers.

Our results show that, while there are—in most projects—less than 10% women developers, there are, on average, no significant differences between men and women regarding general contribution statistics. This indicates that contributions made by women have similar success as contributions by men. As for developer coreness, we find that, on average, both groups can be considered largely alike in terms of their position in the hierarchy. The crucial difference is that, at the top of the coreness scale (i.e., the top of the organizational hierarchy), there are significantly more men than women.

Moreover, we find that there is a less significant association between a developers' gender and their position in the organizational hierarchy than between the time the developer has already been active and the number of contributions the developer has made overall.

It is important to note that, while our results indicate a certain degree of gender balance for the distribution of coreness values, the developers in leadership positions in OSS projects are still men. This suggests that further countermeasures against gender imbalance shall be directed toward the top of the organizational hierarchy.

In summary, we make the following contributions:

- We collect and analyze data from 20 popular OSS projects to investigate the influence of the gender of developers on general contribution statistics and developer coreness.
- We use different metrics to measure developer coreness by analyzing different kinds of socio-technical developer networks of an OSS project.
- We provide evidence that, despite a certain degree of gender balance for the distribution of coreness values, the top of the organizational hierarchy of OSS projects is still male-dominated.

– We show that the gender of developers has a weaker association with their standing in the organizational hierarchy than other factors such as their active time in the project.

2 Background

In this section, we provide an overview of the basic concepts and techniques underlying our study.

2.1 OSS Development on GitHub

GITHUB[4] is one of the largest platforms for collaborative software development. On GITHUB, GIT repositories can be hosted and managed. Developers can review and manage the contributions of others and also track and create issues of the project (Dabbish et al. 2012).

Development in GITHUB and with GIT usually follows the same pattern: A developer who wants to contribute to the project has to first fork the main repository into their own private fork. From there, they can apply whatever changes they want. Once they are done, they have to open a pull request on the main repository. This then gets reviewed by the community, which either accepts the change, rejects the change, or requests changes or improvements to the contribution. If, in the end, the pull request is accepted, it gets merged into the main repository. This process of change and review leaves behind technical and communication artifacts. Each change within a pull request is a commit, which can be seen in the main repository if the pull request gets merged. Furthermore, all communications within pull requests are stored and can be mined through GitHub's API. These data can be used for analyzing the contribution structure of OSS projects (Gousios and Spinellis 2017; Kalliamvakou et al. 2014).

2.2 Developer-Network Analysis

One major step of our study is developer-network analysis, which is a commonly used method to infer community structures in open-source software (OSS) projects (López-Fernández et al. 2009). A commonly used type of network to analyze community structures are developer networks. A *developer network* is a network that captures the relation between developers with regard to their communication (via e-mails, issue comments, ...) or technical interactions such as co-changes (i.e., two developers change the same part of the source code with different commits) (Yang 2014). A combination of these relation types is also possible (e.g., issue comment + co-change) (Joblin et al. 2017a). An example of such a combined network is shown in Fig. 1. The edges between developer A and the other two developers are edges created through issue comments. The edge between developers B and C is created through co-changes.

Formally, a developer network is a graph $G = (V, E)$. The set of vertices V is the set of developers. The set of edges E is either the set of co-change edges $E_{\text{co-change}} \subset V \times V$, the set of issue comment edges $E_{\text{issue}} \subset V \times V$, or a combination of both $E_{\text{comb}} = E_{\text{co-change}} \cup E_{\text{issue}}$. In the network on the right side of Fig. 1, the dotted lines denote elements of E_{issue} and solid lines elements of $E_{\text{co-change}}$.

After building a developer network, it is possible to analyze the structures using common network analysis metrics such as degree centrality, eigenvector centrality, or hierarchy (Joblin

[4] https://github.com/

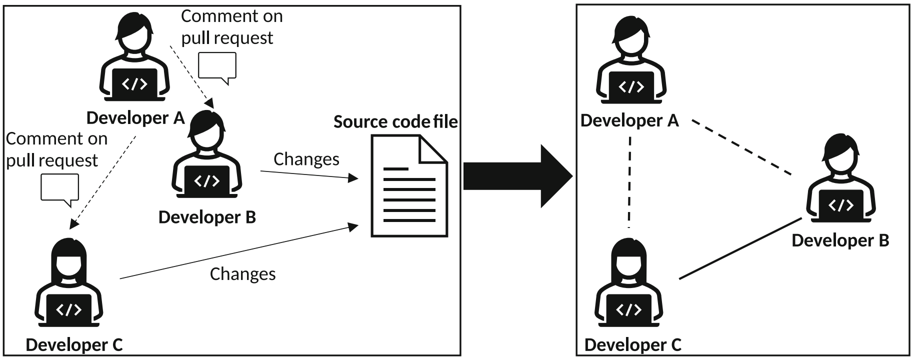

Fig. 1 Illustration of an OSS development process including commits and comments on the left. Developers B and C both change the same code file with separate commits. Developer A comments on both changes in their respective pull requests. The resulting combined network is shown on the right (dotted lines represent issue comment edges and solid lines co-change edges)

et al. 2017a). In our study, we use developer-network analysis to analyze the importance of developers in an OSS project in terms of developer coreness.

2.3 Developer Coreness

Analyzing the organizational hierarchy of OSS projects, developers are usually classified as either core or peripheral (Joblin et al. 2017a). The *core group* is sometimes referred to as the group of gatekeepers of a project. These are developers that have been with the project for a long time and are responsible for a majority of the contributions to the project. It is commonly assumed that this typically small group of developers (say the top 20%) are responsible for the majority (say 80%) of the contributions to OSS projects. The *peripheral group* comprises occasional contributors or one-time bug fixers (i.e., people that only work on the project short-term or occasionally). This group is assumed to make up the majority of the developers of an OSS project (Crowston et al. 2006; Terceiro et al. 2010).

The classification of the developers into the two groups is usually done by calculating the centrality of the developers in the developer network and by dichotomizing these values. Since the dichotomization can lead to memory loss (Fedorov et al. 2009), we choose to use a metric called *developer coreness* to represent the importance of the developers in the projects.

The rationale behind developer coreness is that importance is often linked to the experience/expertise that developers have gained in an OSS project. Since experience is not a binary concept (i.e., one has it or not), it makes sense to represent this as a continuous value. With developer coreness we do exactly this. We use social network analysis to calculate the usually used metrics for the core/peripheral classification but leave out the dichotomization step, thus providing a continuous coreness value for each developer. This value serves as an operationalization of the importance/experience of a developer in the OSS project and represents the organizational hierarchy.

Specifically, the two metrics we use to operationalize developer coreness are *eigenvector centrality* and *the position in the hierarchy*. The centrality of a node using eigenvector centrality is determined by the centralities of the nodes in its direct neighborhood. This means that developers that are connected to other influential and important developers have higher centrality values than developers that are connected to less important developers in the network. The higher the centralities of the surrounding nodes, the higher the centrality of the node (Joblin et al. 2017a). The formula to calculate the centrality of a node i is:

$$x_i = \frac{1}{\lambda} \sum_{j \in N(i)} x_j \tag{1}$$

$N(i)$ represents the set of all neighbors of node i and λ is a proportionality constant (Joblin et al. 2017a; Brandes and Erlebach 2005).

Hierarchy is a more complex metric. It describes how local groups within a network are organized relative to each other. The hierarchy metric of a single node is comprised of the node degree (i.e., the number of edges connected to that node) and the clustering coefficient of the node. The higher the node degree and the lower the clustering coefficient, the higher the node is in the hierarchy (Joblin et al. 2017b). So the top of the hierarchy are those developers that communicate with lots of other developers and are not clustered into local communities. The clustering coefficient for a node i here is calculated as follows:

$$c_i = \frac{2n_i}{k_i(k_i - 1)}, \tag{2}$$

e with k_i representing the number of edges that are connected to node i and n_i being the number of edges between the neighbors of i (Boccaletti et al. 2006).

2.4 Perceived Developer Gender

The gender of people is not a binary concept but rather resides on a spectrum. Some people identify themselves within the traditional binary classification of genders (i.e., men and women), some identify as non-binary. In OSS development, however, the developers often do not know their fellow developers in person but only by their name, and names are culturally tied to genders and perceived as such (even if somebody identifies as non-binary). For example, the name *Mary* is usually perceived as female, while the name *Thomas* is perceived as male. Since developers in most cases only know the name of their collaborators, it is fair to assume that they classify them as either men or women in their minds without any further personal knowledge. This means that, when analyzing gender bias in OSS projects, looking at the perceived gender arising from developers names suffices to make statements about the influence of developer gender on the development process. We use the notion of perceived gender to clarify that this is how a name is perceived as either female or male in most cases. While there are names such as *Cameron* that can be perceived as both male and female, there is no reliable database for these names, and the perception of such names depends on the cultural background of the person perceiving the name. Therefore, detecting names as gender-neutral can introduce additional biases, which we discuss in Sections 6.6 and 7.1.

The *name-to-gender inference* method is also one of the most commonly used when analyzing the influence of genders in OSS development (Santamaría and Mihaljević 2018; Mislove et al. 2011). Alternatively, face-to-gender inference (Lu et al. 2006; Moghaddam and Yang 2002) uses the profile pictures of developers across social networks and machine learning to infer gender. Gender inference through writing style (Argamon et al. 2003) is based on the hypothesis that women and men have different writing styles and can thus be identified by their comments and messages. For our study, we choose name-to gender inference method as this provides the most reliable results given the data we have access to. Furthermore, since name-to-gender inference is widely used in studies with related goals (Bosu and Sultana 2019; Prana et al. 2021; Qiu et al. 2019), we gain the benefit of comparability.

In what follows, we refer to developers with male identified names as men and to developers with female identified names as women. We do this for simplicity and not to classify the person behind the name.

3 State of the Art

There has been considerable research on the topic of gender and women in OSS projects over last decade. Trinkenreich et al. (2021) find that, since 2010, there have been, at least, two new publications on the topic per year, with 2019 showing a peak of 15 new studies. They categorize these publications into different topics. In this section, we follow the general structure of four of these topics: who the women that contribute to OSS projects are, what types of contributions they make, what challenges they face when contributing, and what strategies were proposed to mitigate the challenges.

Vasilescu et al. (2015) find that, of over 800,000 contributors on GitHub, only 9% can be identified as female. Similar to that, Qiu et al. (2019) find that, of 300,000 randomly sampled GitHub accounts, only 9.7% belong to women.

The number of contributions of female developers is also a much researched topic. Terrell et al. (2017) report that only 5.2% of 158.464 analyzed pull requests were submitted by women, which is similar to the results of Kofink (2015), who states that of over 1.8 million pull requests only 4.5% were submitted by female contributors. When looking at the top of the organizational hierarchy, Canedo et al. (2020) conclude that of 711 analyzed projects on GitHub, only 5.24% even have core developers and, of all detected core developers across all these projects, only 2.3% are women. In the same vein, Bosu and Sultana (2019) find that less than 10% of core developers in the 10 analyzed projects are female. In contrast to Canedo et. al and Bosu et. al, we employ a network perspective, which allows us to define a continuous measure of developer coreness based on the project's socio-technical structure.

Robles et al. (2016) investigate the types of contributions female developers make in OSS projects. They find that 31% of women are only coders (i.e., only contribute code changes), 45% are only non-coders (i.e., contribute only non-code changes), and 24% of women contribute to both coding and non-coding activities. Men, on the other hand, are predominantly coders, with over 50% of the men that answered the survey, not doing anything else in the project. Only 25% of them are strict non-coders, and the other 25% perform both coding and non-coding actions. In our study, we investigate what specific tasks (e.g., commenting on pull requests) are done more often by women and men.

The goal of many studies on gender differences in OSS projects is to find the challenges that women face as well as to propose solutions for increasing the number of women. Vasilescu et al. Vasilescu et al. (2015) find that diverse teams perform overall better than one-sided teams. Trinkenreich et al. (2021) define eight categories of challenges women face when contributing to OSS projects. The first of the categories is a lack of peer parity: women feel outnumbered, as the number of other female contributors is still low. Other challenges include the stereotyping of female contributors, which can lead to biased reviews of their contributions and thus foster a more or less toxic project culture (Kuechler et al. 2012). Frluckaj et al. (2022) find that especially in the beginning stages of participation, women are faced with visibility challenges and fear of standing out negatively. These challenges can all lead to a discouragement of women to join OSS projects. In our study, we also identify that there are overall much more men than women participating in OSS projects. Moreover, we take a deeper look into the social hierarchy of OSS projects to gain insight into how men and women are represented in OSS projects. We aim at helping OSS communities to identify potential reasons for low participation of women and developing better countermeasures by providing insight into where gender imbalance exists in their projects.

In contrast to the body of research on the role of gender in OSS projects, we perform a socio-technical analysis on a network representation of the organizational project structure to provide a more fine-grained and detailed perspective on the differences of men and women. As. Joblin et al. (2017a, 2015); Joblin and Apel (2022) find that using socio-technical developer networks can provide a richer and more nuanced view of the organizational structure of OSS projects than count-based metrics alone. We build upon their insights to study the effect of developer gender on the implied organizational hierarchy of OSS projects. This way, we seek to provide a better understanding of the effect developer gender has on the developers standing in OSS projects.

To address gender imbalance, researchers proposed solutions such as creating events and spaces just for the women in a project (Canedo et al. 2020; Calvo 2021; Singh 2019). This can help to increase the perception of peer parity and make women feel less outnumbered. Another possible solution is to introduce a code of conduct to foster a more inclusive project culture (Tourani et al. 2017). While this is not a perfect solution, it can definitely send a signal to all contributors that non-inclusive behavior is not welcome and thus be a step into the right direction.

4 Study Design

In this section, we describe the research questions and the study design, including data acquisition, case studies, and operationalization.

4.1 Research Questions

The overarching goal is to analyze the relationship between developer gender and the development process as well as the organizational hierarchy in OSS projects. For this purpose, we analyze historical data from the projects' interaction channels and code repositories. This way, we gain insight into the development process itself and how outside factors can influence it.

We start with analyzing the relationship between developer gender and selected general contribution statistics such as the pull request acceptance rate or the issue comment count. These statistics are explained in more detail in Section 4.6. We pose the following question:

RQ$_1$ *Is there a difference between men and women in terms of general contribution statistics in OSS projects?*

The second part of our study refers to the organizational hierarchy of OSS projects as captured by the notion of developer coreness (cf. Sec 2.3). The question is whether the gender of a developer has an effect on their standing in the hierarchy of a project:

RQ$_2$ *Do the overall coreness values and the coreness distributions differ for men and women in OSS projects?*

To analyze this, we relate established network metrics about the organizational hierarchy of OSS projects to the gender of developers, which could give us unique and novel insights by combining different established metrics. We do so, as previous studies (Joblin et al. 2017a, 2015; Joblin and Apel 2022) have shown that analyzing these networks can provide a richer and more nuanced view of the organizational structure of OSS projects. Therefore, we aim at finding a more detailed and intricate perspective on the differences of men and women in the organizational hierarchies of OSS projects.

4.2 Hypotheses

To answer our research questions, we investigate whether there is a difference in the involvement of men and women in the development process and whether there is a difference in their coreness values.

Therefore, we formulate the following hypotheses. Regarding the general contribution statistics, the hypothesis is:

H$_1$ *There is a difference in the involvement of men and women in the development process with regard to basic contribution statistics in OSS projects.*

Concerning the coreness, we formulate the following hypothesis:

H$_2$ *There is a difference in coreness values of men and women in OSS projects.*

In RQ$_2$, we are interested not only in individual or average coreness values, but also in their overall distribution. We hypothesize that the distribution for males and females should not be the same if there is an average difference between the coreness values of male and female developers. Therefore, the fifth hypothesis is:

H$_3$ *The distributions of coreness values of men and women have different shapes in OSS projects.*

4.3 Subject Projects

We analyze twenty subject projects to answer our research questions and to test our hypotheses. We have selected these projects to vary in size and age, as shown in Table 1. There are small projects with only a few thousand developers, commits, and issues, such as REVEAL.JS, but also large projects with over 60,000 developers and commits, such as VS CODE. Furthermore, we have selected three projects that were started by women, namely COOKIECUTTER, DBATOOLS, and GHOST[5]. With the inclusion of these projects, we aim at understanding whether projects that were started by women differ in the contribution statistics and the organizational hierarchy from projects apparently not started by women. The historical data of the projects also differs in age. There are older projects, for which the data reaches back to 2010, but also newer projects dating back to 2018. These dates do not necessarily reflect the real age of the projects, but rather the start of their development process using GitHub.

We further chose a diverse set of projects, based on their popularity and availability of data on GITHUB, as their communities are well developed and matured. As our analyses are quite time-consuming and require a lot of computational resources, we decided to limit the sample size to the selected 20 projects. With our project selection spanning different domains and programming languages, which we suspect has an influence on the collaboration of developers, we aim at generalizability among OSS projects that use GitHub as their development platform. We discuss our choice of subject projects further in Section 7.2.

[5] https://github.com/roxiomontes/Women-in-OpenSource

Table 1 Overview of subject projects, including the number of developers, the number of commits, the number of issues, and the time frame of the data set

Project	# Developers	# Commits	# Issues	Time frame
ANGULAR	23 420	12 403	38 524	2014–2020
ATOM	21 402	32 402	21 163	2012–2020
BOOTSTRAP	25 252	2 267	31 737	2011–2020
COOKIECUTTER	1 444	1 052	1 815	2013–2023
DBATOOLS	1 451	8 889	8 636	2015–2023
DENO	3 198	4 805	8 762	2018–2020
ELECTRON	15 837	10 673	26 737	2013–2020
GHOST	5 210	15 535	16 793	2013–2023
KERAS	13 604	4 626	13 512	2015–2020
MOBY	29 872	14 097	41 735	2013–2020
NEXT.JS	14 449	3 891	15 356	2016–2020
NEXTCLOUD	10 139	16 228	22 726	2016–2020
REACT	16 445	6 922	20 257	2013–2020
REDUX	4 231	701	3 931	2015–2020
REVEAL.JS	3 047	2 242	2 769	2011–2020
TENSORFLOW	36 848	92 432	45 664	2015–2020
THREE.JS	8 623	27 201	20 856	2010–2020
TYPESCRIPT	18 947	17 956	41 251	2014–2020
VS CODE	68 675	68 350	111 126	2015–2020
VUE	9 869	3 124	9 351	2016–2020

4.4 Data Mining and Preparation

We have mined the raw data for our analyses using the open-source tool CODEFACE[6]. Using this tool, we mine the historical data of the projects' version control system (GIT repositories). This includes author names, e-mail addresses, and commit information. To obtain the information needed about pull request and issue data, we use the public API of GitHub. Using these two mining methods, we obtain a list of developers. As the developers might appear with different names or mail addresses, we disambiguate the developer data using the method by Oliva et al. (2012).

As many projects have existed since before the popularity of GitHub as a development platform grew, we now have two data sets (commit data and issue data) with different start dates meaning the first commit may have happened years before the first issue or pull request were opened on GitHub for a certain project. To avoid compromised analyses and convoluted developer networks, we use the open-source tool CORONET[7]. This tool allows us to build networks from these heterogenous data sets, which we describe in Section 4.7 in detail. Furthermore, it is possible to prepare the data for network construction within the tool, which we use to cut the commit and issue data to the same first and last dates, providing us with a consistent base of data for our analyses.

6 https://github.com/siemens/codeface/

7 https://github.com/se-sic/coronet/

Table 2 Number and percentage of developers per gender. All developers that could not be identified as men or women fall into the *unknown* category

Project	#Men(%)		#Women(%)		#Unknown(%)	
ANGULAR	10371	(44.3%)	476	(2%)	12581	(53.7%)
ATOM	15750	(73.6%)	1009	(4.7%)	4643	(21.7%)
BOOTSTRAP	18424	(72.6%)	1173	(4.6%)	5764	(22.7%)
COOKIECUTTER	1382	(71.2%)	310	(16%)	248	(12.8%)
DBATOOLS	1413	(50.7%)	707	(25.3%)	669	(24%)
DENO	2195	(68.6%)	152	(4.8%)	851	(26.6%)
ELECTRON	10963	(69%)	736	(4.6%)	4187	(26.4%)
GHOST	4932	(68.6%)	1268	(17.6%)	990	(13.8%)
KERAS	8268	(60.8%)	886	(6.5%)	4450	(32.7%)
MOBY	22506	(75.3%)	1219	(4.1%)	6173	(20.6%)
NEXT.JS	11128	(76.9%)	643	(4.4%)	2699	(18.7%)
NEXTCLOUD	5751	(56.7%)	356	(3.5%)	4032	(39.8%)
REACT	8573	(52.1%)	404	(2.5%)	7493	(45.5%)
REDUX	3496	(81%)	159	(3.7%)	659	(15.3%)
REVEAL.JS	2268	(74.4%)	162	(5.3%)	617	(20.2%)
TENSORFLOW	22098	(60%)	2319	(6.3%)	12431	(33.7%)
THREE.JS	5521	(64%)	425	(4.9%)	2677	(31%)
TYPESCRIPT	14275	(75.3%)	765	(4%)	3920	(20.7%)
VS CODE	41126	(64.6%)	2950	(4.6%)	19599	(30.8%)
VUE	6249	(63.3%)	469	(4.8%)	3151	(31.9%)

4.5 Gender Detection

The last step of data aquisition is the detection of developers' genders. We upload the list of developer names of Section 4.4 to the Web-based gender detection tool GENDERAPI[8]. Recent studies have found that GenderAPI is among the best name-to-gender inference tools (Santamaría and Mihaljević 2018). It takes the name of the developers and uses a large database of labeled names to categorize whether the name is male or female. If the name is not classifiable, we disregard the developer from our analyses. This can happen as some people opt to use non-classifiable nicknames instead of their real name.

The results of the classification are shown in Table 2. There, we see that, in all projects, the number of women is significantly lower than the number of men. In most cases, the developers identified as women make up less than 10% of the total number of developers.

After gender detection, we save the gender data for further analyses. Using CORONET, we merge the gender data with the other developer data.

4.6 General Contribution Statistics

For our first research question, we analyze the general contribution statistics of our subject projects. A contribution in our case is defined as any action by a developer made in the

[8] https://gender-api.com/

development process. This includes code contributions in the form of commits, commenting on issues or pull requests, and administrative actions such as merging a pull request. Gaining insight into whether the gender of developers has an influence on their contributions is an important first step when analyzing the overall influence of developer gender on OSS development. Contributions are central in OSS development as no project can flourish without an active community.

Specifically, we consider the following statistics: the number of pull requests a developer has created, the number of pull requests that have been merged, the number of opened issues, the number of comments made within an issue or a pull request, the number of pull requests a developer has merged, the number of changed files through commits, the number of commits, and the diff size of a developers' contributions.

For all these statistics, it holds that the larger the number indicates more involvement in the development process. So, a developer with 10 created pull requests is tendentially more involved in the development process than a developer with only 5 created pull requests.

We perform a two-sided *Wilcoxon Mann-Whitney U test* on a significance level of 0.05.

4.7 Analysis of Coreness

As described in Section 2.2, we build three types of developer networks per project to analyze their organizational hierarchy from different angles: co-change networks, issue communication networks, and a combination of both. For this purpose, we use the tool CORONET. We calculate the coreness value for each developer and network type with both eigenvector centrality and hierarchy (see Section 2.3). The coreness values calculated through eigenvector centrality are in the interval [0, 1], whereas hierarchy values are not. To achieve comparability, we normalize the coreness values calculated through the hierarchy metric. We do so by dividing every single hierarchy value by the overall largest hierarchy value measured in the project.

After obtaining the coreness values for all developers, we compare the values of men and women using two-sided Wilcoxon Mann-Whitney U tests on a significance level of 0.05. We apply this comparison for each project, network type, and coreness value type separately.

We visualize the coreness values of the developers in Q-Q plots to determine whether the distributions of the coreness values of men and women follow the same shape. In addition, we visualize the distribution of values using violin plots to obtain a better overview of where the majority of the coreness values for men and women are.

Finally, we analyze whether other factors such as the time a developer has already been active in a project or the number of contributions the developer has made are more strongly associated with the coreness of a developer than their gender. We choose the time a developer has already been active as it is generally conjectured in the literature that developers with a longer tenure are higher up in the organizational hierarchy (Joblin et al. 2023). We further choose the number of contributions a developer made as this is also a strong indicator of a higher standing of developers. While our coreness metrics in part rely on the number of contributions, it is still a useful metric. Joblin et al. (2017a) show that even though the number of contributions are a factor in network-based coreness metrics, they are not the only factors and as such the number of contributions is useful for our analysis. For this purpose, we split the data for each project into 6-month windows and calculate the coreness of each developer in the co-change and issue networks of the different time windows with both eigenvector centrality and hierarchy centrality. Then, we determine for each developer in each time window, how long they are already active in the project and how many contributions (commits

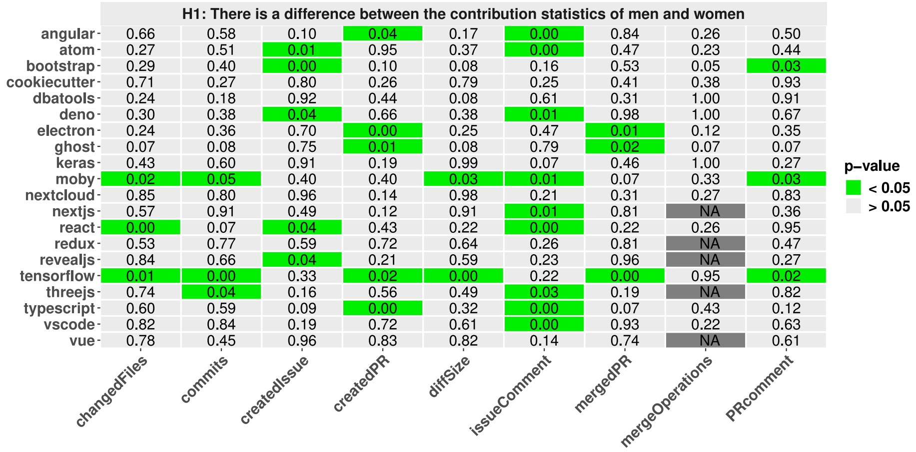

Fig. 2 Results for the Wilcoxon Mann-Whitney U tests of H_1. PR means pull request

for the co-change network and issues for issue network) they have made. We then fit a linear regression model to the data to determine which of the factors have a statistically significant relationship with the coreness value of the developers and how strong this relationship is. We do this for each project individually, which we present in added-variable-plots, which show the influence of each factor on the dependent variable separately and also for all data together to see whether there is a trend over all projects. In addition to the linear regression model, we calculate Spearman correlation coefficients to determine the strength of the relationship between the factors and the coreness values.

5 Results

In this section, we present the results of our analyses of the general contribution statistics, coreness values, and coreness distributions.

5.1 General Contribution Statistics (H_1)

H_1 states that there is a difference between the involvement of men and women in the development process in terms of general contribution statistics. The results of the statistical test are shown in Fig. 2. Most of the results do not show statistical significance.[9] Still, there are some significant results, especially in the issue comment statistic. In addition to the two-sided statistical test, we have also performed statistical tests with the alternative that either men or women are more involved. These results are shown in our supplementary material[10]. There we can see that men seem to be more involved in issue discussions on GITHUB in almost half of the projects. In addition to the statistical tests, we have also calculated the average

[9] The *merge operations* column contains five unavailable values (NA), namely for the projects VUE, THREE.JS, REVEAL.JS, REDUX, and NEXT.JS. This happens because in these three projects there were no women identified that performed merge operations. This means that in these projects the people that merge reviewed pull requests, are all either male or could not be classified as either male or female.

[10] https://se-sic.github.io/paper-perceived-gender/

number of files changed and the average diff size of commits made by men and women. These results are shown in Table 3. Here, we see that the average number of files changed is not consistently higher for men or for women across our subject projects. For some projects (i.e., TYPESCRIPT or GHOST), the average number of files changed is higher for women than it is for men. We see the same pattern for the average diff size of commits. For this statistic, the average is also not consistently higher for either men or women. These results indicate that looking at the technical aspects of the contributions alone is not sufficient to find differences between men and women in OSS projects but that looking at more intricate socio-technical network metrics can provide a more detailed view.

Overall when looking at the results for H_1, we do not obtain a clear picture from the statistical tests. Although there are a few statistically significant results that suggest that there is a difference in general contribution statistics between men and women, these are less than a quarter of all results. Therefore, we **reject H_1**.

5.2 Analysis of Coreness (H_2)

With H_2, we hypothesize that there is a difference in the coreness values of men and women in OSS projects. The results of our statistical tests are shown in Fig. 3. We see that there are many statistically significant results. Most of them are obtained from issue networks and combined networks. Since there are usually a lot more actions performed via issues and

Table 3 Average diff sizes and number of files changed in commits made by men and women

	Files Changed		Diff Size	
Project	Men	Women	Men	Women
ANGULAR	89	43	8 863	12 308
ATOM	27	12	8 726	870
BOOTSTRAP	8	3	4 802	40
COOKIECUTTER	6	19	348	1 627
DBATOOLS	19	116	3 419	2 085
DENO	32	88	4 124	22 026
ELECTRON	25	36	3 654	2 894
GHOST	51	302	5 352	31 700
KERAS	4	9	173	4 417
MOBY	36	24	4 388	2 320
NEXTCLOUD	63	7	11 860	1 481
NEXT.JS	14	4	782	156
REACT	26	37	3 946	1 208
REDUX	7	3	3 626	48
REVEAL.JS	2	3	888	54
TENSORFLOW	57	60	15 648	11 756
THREE.JS	13	6	11 438	4 486
TYPESCRIPT	111	245	45 777	115 476
VS CODE	34	14	8 847	1 625
VUE	10	2	5 788	21

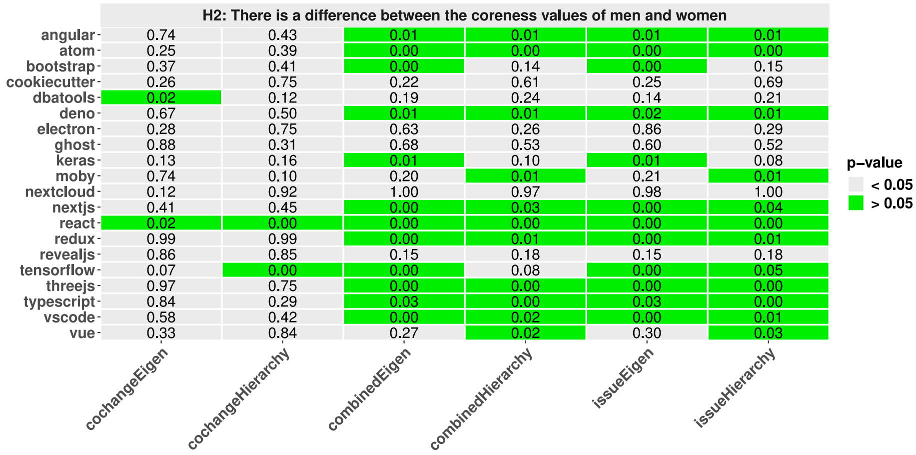

Fig. 3 Results for the Wilcoxon Mann-Whitney U tests of H_2. Eigen means eigenvector centrality

pull requests of a project than there are commits, the issue interaction edges outweigh the co-change edges in the combined network. Therefore, the coreness values of the combined networks are similar to the issue networks. This shows that, when it comes to activities that manifest in issues, there is a difference in coreness values between men and women. In addition to a two-sided test, we have also performed statistical tests with the alternative that men have greater coreness values and that women have greater coreness values. These are shown in our supplementary material and show that in most of the test configurations that show a difference between men and women, men seem to have the overall higher coreness values. This indicates that men are more active when participating in issues (discussing, opening, closing, or editing) in OSS projects.

The results for the co-change networks do not show many significant results that suggest that there is a difference in the coreness values of men and women. The results for the other two network types do show several of such significant results. So, we **partially accept H_2**. The hypothesis holds for issue networks and combined networks.

5.3 Analysis of Coreness Distribution (H_3)

In H_3, we hypothesize that the distributions of coreness values for men and women are dissimilar. We address this question by inspecting violin plots and Q-Q plots of the distribution of the coreness values for both groups. Since we have one violin plot and one Q-Q plot per project and network/coreness value combination, we only show plots of selected projects here. All other plots can be seen on our supplementary Website.

In Fig. 4, we show the plots for TENSORFLOW, VS CODE, BOOTSTRAP, and MOBY. For TENSORFLOW's co-change network with eigenvector centrality as the coreness metric, we see that the distribution of the coreness values is very similar for men and women, confirming the results of the Wilcoxon Mann-Whitney U test (Fig. 3) that the coreness values are not

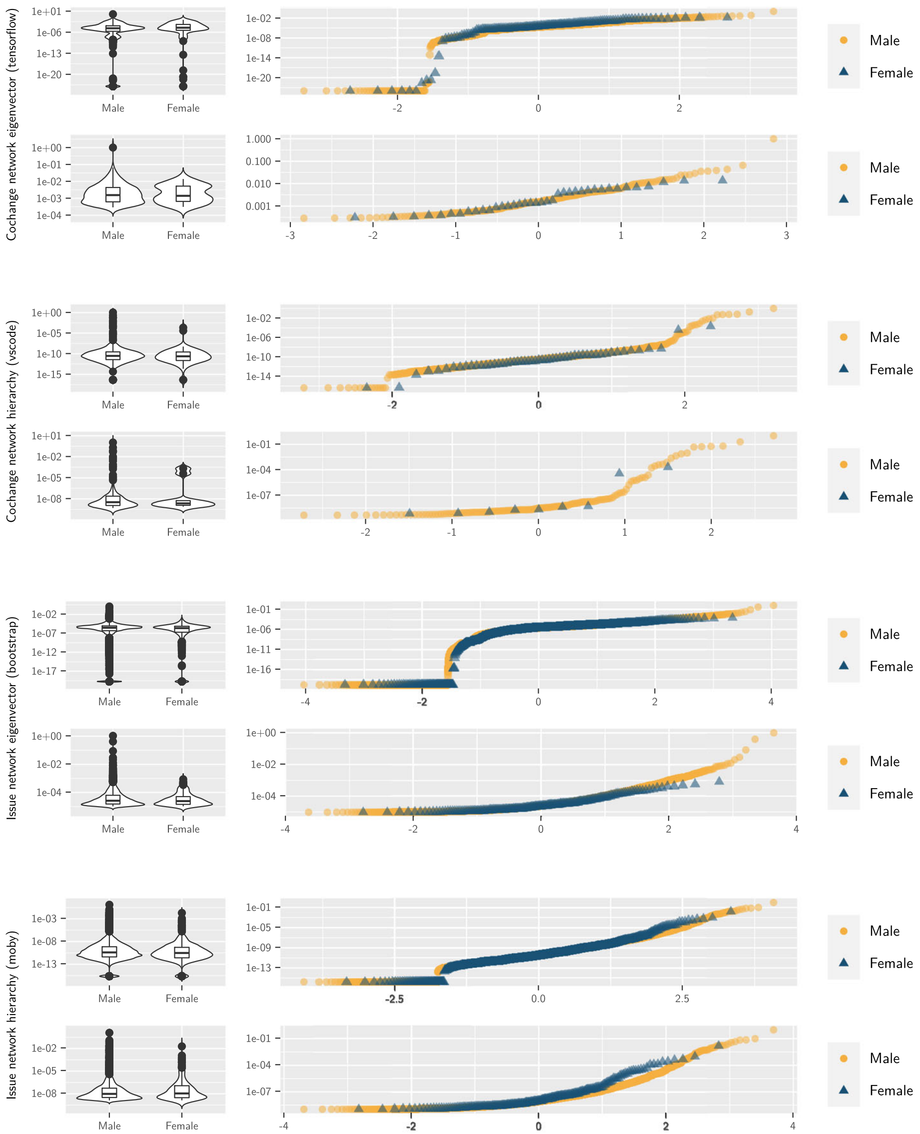

Fig. 4 Coreness value distributions for TENSORFLOW, VS CODE, BOOTSTRAP, and MOBY. Each project is divided into two rows and two columns of plots. In the left column, we show violin plots with nested box plots. In the right column, we show quantile plots. The plots in the upper row are created with all coreness values of the projects and in the lower row we only use the coreness values above the 80th percentile. The y-axes of all plots are the coreness values of developers on a logarithmic scale for better visualization. The x-axes of the quantile plots are theoretical quantiles meaning that they describe a normal distribution with mean 0 and standard deviation 1

significantly different for both groups. Interestingly, the plots show that men tend to reach higher single values, meaning that, although there is no statistical difference in coreness values, individual male developers seem to reach a higher standing in the organizational hierarchy. More importantly, the absolute top (say the top 10%) of the scale is male dominated. This behavior can also be observed in the results for the project VS CODE with coreness values derived using hierarchy from the co-change network. This holds even though the Wilcoxon Mann-Whitney U tests in Section 5.2 show no statistical significance across the whole organizational hierarchy.

The distributions for BOOTSTRAP with coreness values derived from the issue network using eigenvector centrality and MOBY with coreness values derived using hierarchy from the issue network show similar results. In these two projects and configurations, there is a significant difference between the coreness values of men and women (see Fig. 3). Nevertheless, the distribution of values follows the same shape for both men and women except for the very top of the scale.

Overall, we obtain the same picture for the other subject projects. There seems to be a pattern here: men reach higher coreness values although the distribution for men and women is similar until the very top of the coreness scale.

As the coreness values of men and women follow similar distribution shapes below the 90th percentile, but not in the upper 10%, we can neither accept nor reject H_3. Therefore, we conclude H_3 to be **inconclusive.**

5.4 Multiple Linear Regression

To further analyze our results, we perform a linear regression analysis on the association between developer gender, active time of developers, number of contributions per developer, and their coreness values. As described in Section 4.7, we perform this analysis on data that we obtain by splitting our projects into 6-month windows and calculating the coreness for each developer in each of the time windows they were active. We then relate the coreness values to the gender of the developers, the time they were active in the project at each time window, and the number of relevant contributions they made in each time window using a multiple linear regression model.

For the analysis of each separate project, we find that, in most cases, the number of relevant contributions and the time a developer has already been active in a project have a more significant influence on the coreness of a developer than their gender. We show examples of this in Figs. 5 and 6. In Fig. 5 we see that the active time (upper left) and the number of contributions (upper right) are positively related with the coreness values of the developers whereas the gender of developers are neither positively nor negatively related. In Fig. 6, we obtain a similar result with the exception that for this project (TYPESCRIPT) the time a developer has been active in the project does not seem to have an association with the developers coreness. The specific results of the linear regression models for the eigenvector centrality on the co-change networks are shown in Table 4. Here, we see that, in all cases, the number of contributions has a statistically significant association with a developers' coreness value; in most of the cases, the time a developer has already been active is related to the coreness; and the gender of the developers only has a statistically significant association with the coreness in four cases. In addition, we present the standardized beta coefficients of all three factors in Table 4. Here, we see that the number of contributions have the highest influence on the coreness, while the other two variables have only little influence. This could

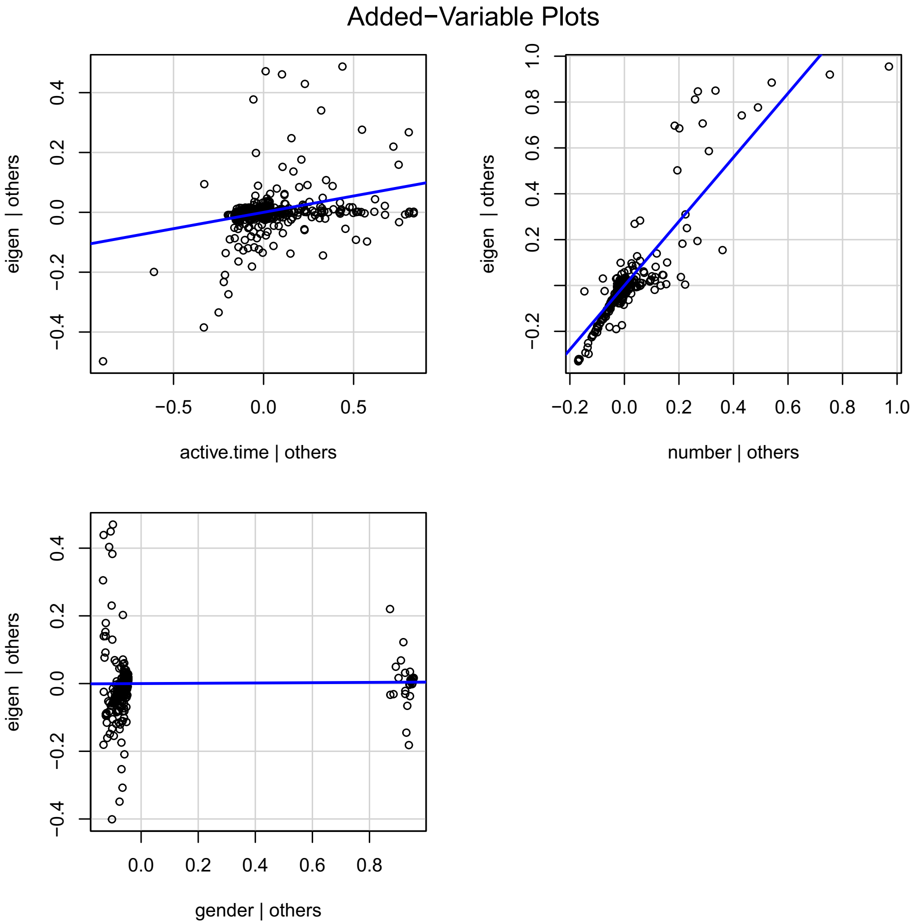

Fig. 5 Added-variable-plot of the association between coreness and the other factors we use for the linear regression model for the project DENO with the coreness values calculated using the eigenvector centrality on the co-change network

be because the coreness of developers is largely influenced by the number of contributions a developer makes. Nevertheless, it is interesting to see, that neither the active time nor the gender of developers seems to influence the coreness values a lot. Finally, we have also performed a Spearman correlation between the active time, the number of contributions, and the gender of developers with their coreness values. These results are also shown in Table 4. Here, we see that the active time and the number of contributions are indeed correlated with the coreness of developers. The gender of developers on the other hand does not seem to be correlated to their coreness values in most cases. The results for the other configurations can be found on our supplementary Website.

Subsequently, we took all data from all projects and added them together to perform a linear regression analysis on the overall results to see whether or not there is a general

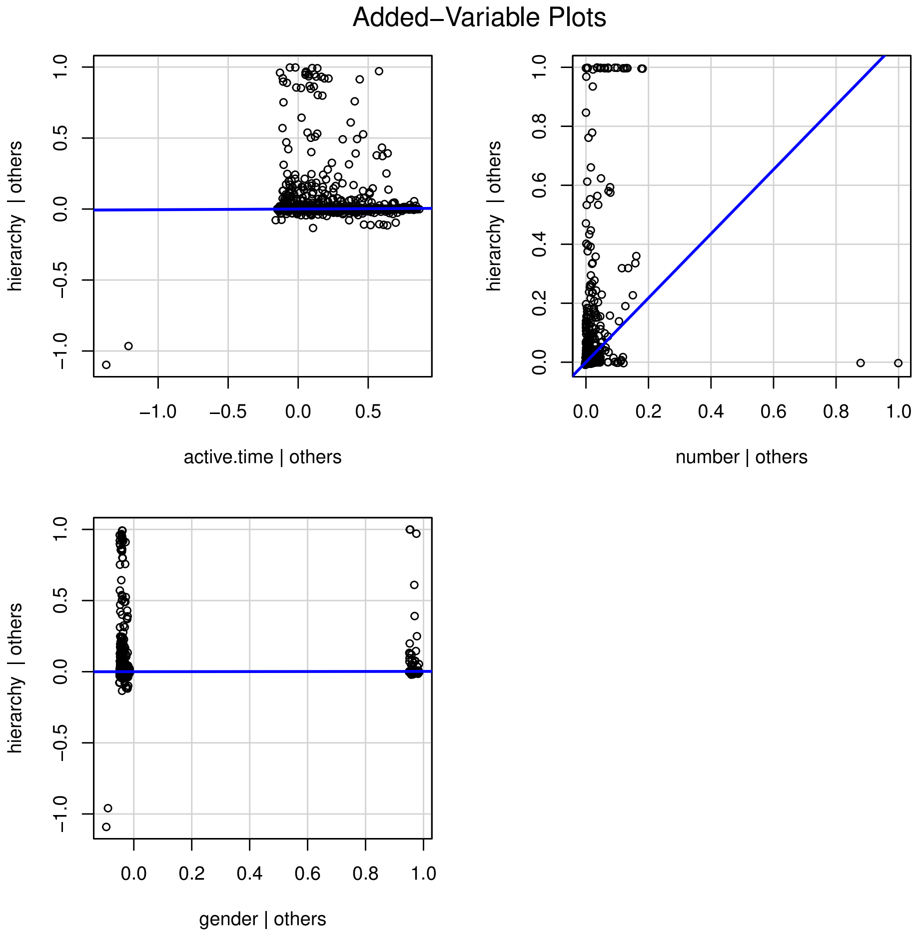

Fig. 6 Added-variable-plot of the association between coreness and the other factors we use for the linear regression model for the project TYPESCRIPT with the coreness values calculated using the hierarchy centrality on the issue network

pattern that we can see. We do this four times, once for each combination of network type and centrality metric. We present the results in Table 5. We find that, for the co-change networks (i.e., the technical interactions between developers), the number of contributions and the time a developer has already been active in the project, have a statistically significant association with the coreness values of the developers, while the gender does not seem to have such an association. The beta coefficients, which we present in Table 6, further confirm this as the gender of developers shows almost no influence on the coreness while especially the number of contributions do. Interestingly, when looking at the correlation coefficients, which we present in Table 7, we see that all three variables are significantly correlated with the coreness of developers. For the issue networks with the hierarchy centrality (i.e., the social interactions between developers), we find that all three factors have a statistically significant association with the coreness of a developer. Interestingly, we find that being a woman has a positive effect on the coreness of the developers, as indicated by the positive regression coefficient.

Table 4 Results of the multiple linear regression for all projects with the coreness values derived from the co-change networks using the eigenvector centrality. The values represent the estimates for each factor and a * means that the result is statistically significant with a significance level of 5%. Also we present the standardized beta coefficient of the factors in the linear regression model and a Spearman correlation coefficient between the factors and the coreness of developers

Project	Linear Regression				Beta Coefficients			Spearman Correlation		
	Active Time	Number of Contributions	Gender	R-Squared	Active Time	Number of Contributions	Gender	Active Time	Number of Contributions	Gender
Angular	0.005	0.997*	0.033*	0.499	0.008	0.702	0.055	0.506*	0.716*	0.027
Atom	0.200*	1.281*	-0.008	0.547	0.161	0.691	-0.008	0.368*	0.648*	0.035
Bootstrap	0.547*	1.497*	0.006	0.698	0.349	0.663	0.004	0.619*	0.640*	-0.121*
Cookiecutter	-0.094	1.312*	0.094	0.314	-0.088	0.561	0.064	0.089	0.514*	0.003
dbatools	-0.015	0.542*	0.002	0.283	-0.04	0.538	0.006	0.362*	0.614*	0.069
Deno	0.109*	1.397*	0.004	0.822	0.149	0.835	0.007	0.578*	0.683*	0.058
Electron	0.270*	1.316*	0.047*	0.656	0.244	0.712	0.063	0.570*	0.702*	0.021
Ghost	0.055*	1.087*	-0.015	0.498	0.079	0.700	-0.025	0.312*	0.729*	0.187*
Keras	0.179*	1.441*	0.026*	0.697	0.169	0.750	0.056	0.535*	0.499*	0.057*
Moby	0.088*	1.527*	-0.006	0.567	0.105	0.725	-0.012	0.474*	0.658*	0.010
Next.js	0.003	1.147*	0.000	0.821	0.004	0.905	0.000	0.334*	0.337*	0.024
Nextcloud	0.009	0.392*	0.004	0.156	0.018	0.389	0.006	0.192*	0.280*	-0.049
React	0.040*	2.022*	0.000	0.542	0.068	0.721	0.000	0.280*	0.565*	0.050*
Redux	0.492*	1.734*	-0.040	0.330	0.319	0.446	-0.024	0.354*	0.433*	-0.023
Reveal.js	0.868*	1.115*	-0.020	0.764	0.538	0.457	-0.013	0.463*	0.537*	-0.047
TensorFlow	-0.012*	1.245*	-0.001	0.797	-0.035	0.898	-0.004	0.431*	0.741*	0.051*
Three.js	0.090*	1.645*	0.007	0.712	0.11	0.798	0.009	0.312*	0.458*	0.025
TypeScript	0.126*	1.287*	-0.032*	0.656	0.139	0.753	-0.043	0.570*	0.723*	0.080*
VS Code	-0.042*	0.635*	-0.003	0.517	-0.087	0.758	-0.008	0.546*	0.642*	-0.001
Vue	0.217*	1.761*	0.003	0.620	0.211	0.734	0.003	0.328*	0.510*	-0.054

Table 5 Results of the multiple linear regression over all projects. The values represent the estimates for each factor and a * means that the result is statistically significant with a significance level of 5%. Also we present the standardized beta coefficient of the factors in the linear regression model in Table 6 and a Spearman correlation coefficient between the factors and the coreness of developers in Table 7

Configuration	Linear Regression			
	Active Time	Number of Contributions	Gender	R-Squared
Co-change-Eigenvector	3.522×10^{-2}*	1.124*	-1.235×10^{-3}	0.477
Co-change-Hierarchy	0.014*	1.011*	-0.002	0.437
Issue-Eigenvector	3.550×10^{-3}*	1.035*	1.582×10^{-4}	0.421
Issue-Hierarchy	1.313×10^{-3}*	8.139×10^{-1}*	2.733×10^{-4}*	0.294

6 Discussion

In what follows, we discuss our results and their implications on gender diversity in OSS projects.

6.1 Research Questions

To understand gender diversity in OSS projects, we have set out to answer two research questions: The first research question is concerned with whether developer gender is related to with general contribution statistics. We found that neither men nor women are significantly more involved in terms of the general contribution statistics of our subject projects. That is, we do not have evidence for an association of gender with the contribution activities of developers. The only notable differences we find are that men seem to be more involved in commenting on issues in some projects, while women seem more likely to create pull requests in a few projects. As for the commits made by men and women, we find that the average number of changed files and the average size of the commit diffs are not consistently higher for any of the two groups. In some cases, the averages are significantly higher for women than for men. This might be due to the fact that there are far fewer women in total, so the average is more easily influenced by outliers. As Frluckaj et al. (2022) find that women are faced with visibility challenges and afraid of standing out negatively, it could be that in these projects only very engaged women join the community which would positively influence the average number of changed files and the average diff size of commits made by women. Why exactly this is the case remains unclear and should be subject to future research. Still,

Table 6 The standardized beta coefficients of the factors in the linear regression model presented in Table 5

Configuration	Beta Coefficients		
	Active Time	Number of Contributions	Gender
Co-change-Eigenvector	0.061	0.674	-0.002
Co-change-Hierarchy	0.027	0.654	-0.004
Issue-Eigenvector	0.023	0.647	0.001
Issue-Hierarchy	0.009	0.541	0.002

Table 7 The Spearman correlation coefficients between the factors of the linear regression model presented in Table 5 and the coreness of developers

Configuration	Spearman Correlation		
	Active Time	Number of Contributions	Gender
Co-change-Eigenvector	0.339*	0.541*	0.042*
Co-change-Hierarchy	0.226*	0.663*	0.051*
Issue-Eigenvector	0.156*	0.399*	-0.016*
Issue-Hierarchy	0.151*	0.466*	-0.012*

our results suggest that, if an association exists, it is rather small. This finding is contrary to what one might expect, given the historical underrepresentation of women as one might assume that the smaller number of involved women that joined, on average, later than men should be less involved in the development process than the overrepresented men. However, the finding is also in line with the findings of Canedo et al. (2020), who find that, there are no statistically significant differences in the commit activities of women and men. Nevertheless, as shown in Table 2, women are still underrepresented in the realm of OSS development. Changing this should be a main concern for the communities of OSS projects, as studies have shown that gender diverse teams perform overall better than one-sided teams (Vasilescu et al. 2015; Shameer et al. 2023). These results alone do not provide a clear overview of the differences between men and women in OSS projects. Therefore, we use socio-technical network analysis to provide a more detailed view of the organizational hierarchy of OSS projects.

Our second research question is concerned with the organizational hierarchy of developers and the relation developer gender has with it. Our results show that there seems to be indeed a relationship of developer gender and the coreness values, especially in the upper 10% of the coreness scale. This suggests that women tend not to rise up to the very top of the organizational hierarchy of OSS projects. This can have many reasons. One is that there might, in fact, be a bias against women that causes this difference. This would mean that OSS communities need to actively check whether such bias exists in their project and handle it. Another possible reason is the huge difference in numbers between men and women. In our subject projects, women only comprised less than 10% of all the developers. This could influence the probability of women rising to the top of the organizational hierarchy, as men outweigh them significantly in numbers. Another explanation is that, historically, the activity of women in STEM is lower than for men (Nimmesgern 2016), meaning men could have an experience advantage. In this vein, Vasilescu et al. (2015) find that women have, on average, six years of experience in programming as opposed to an average of nine years for men. Moreover, they find that most women contributed to OSS projects for less than five years. If there are men that are active in a project for longer than the female developers, they clearly have a better chance to rise to the upper parts of the coreness scale. This is further strengthened by our finding that the time a developer has already been active in a project seems to have a stronger relationship with their coreness value than their gender and that this time is significantly correlated to the coreness. Another reason may be that, as documented in the 2013 survey by Robles et al. (2016), over 53% of women stated that they devote less than 5 hours per week to OSS development. Only 14% devote more than 40 hours per week. In both cases, this is a higher percentage than men that answered the survey. This shows that women are overrepresented among the occasional and the professional full-time developers. This result suggests in turn that men are overrepresented among the volunteer developers that devote 5 to 40 hours to the projects who could be more motivated to rise in these projects. While being a full-time developer should intuitively lead to higher coreness values, the opposite could also hold true as these developers might be too specialized with the needs of their employer in the projects. This could have negative influences on the coreness values.

Regarding the results of our linear regression analysis, we find that being a woman has a positive relationship with their coreness value. This could be because women tend to be more active in social interactions in OSS projects. Robles et al. (2016) find that about 45% of women are strictly non-coders in OSS projects while only 25% of men fall into the same category. Frluckaj et al. (2022) find that women more often than men join OSS projects because of personal invitations. This could mean that women tend to focus more on social interactions, documentation, and helping others.

Our results are also consistent with the findings of Canedo et al. (2020), who find that, in only about 5% of over 700 analyzed OSS projects on GitHub, there are women who can be classified as core developers; and among all core developers of all these projects, only about 2% are women. This has the potential to send a negative message to women that want to join OSS projects. This is also in line with the results of Vasilescu et al. (2015), who show that women are feeling frustrated when there is a lack of peer parity, and core developers are usually the most present people in the projects. A lack of women as core developers could increase the impression of a male-dominated project.

6.2 Projects Started by Women

As described in Section 4.3, we have selected three projects that were apparently started by women, namely COOKIECUTTER, DBATOOLS, and GHOST. We did this to find whether there is a difference in the contribution statistics and organizational hierarchy in these projects if the visibility of some prominent women is higher. Interestingly, we do not find a substantial difference between projects started by women and the other seventeen projects. The only notable result we find is that, when looking at the results of the general contribution statistics in Fig. 2 and the results of the coreness in Fig. 3, there are almost no individual results where there is a difference in involvement of men and women in the development process and only one result where there is a difference in the organizational hierarchy. This may suggest that in these projects, there is a certain gender balance in the development process and the organizational hierarchy. Moreover, when looking at the results of the coreness distribution (as shown on our supplementary Website), we see that, for certain configurations, the coreness distribution of women has a higher distribution (with regards to the values) than the distribution of male developers. Since these are not consistent results, we can not make a general statement about this. However, they could indicate that the higher visibility of women could influence the standing of women in the organizational hierarchy of OSS projects which is a proposed solution to solve the gender imbalance in OSS projects (Frluckaj et al. 2022; Canedo et al. 2020; Calvo 2021; Singh 2019).

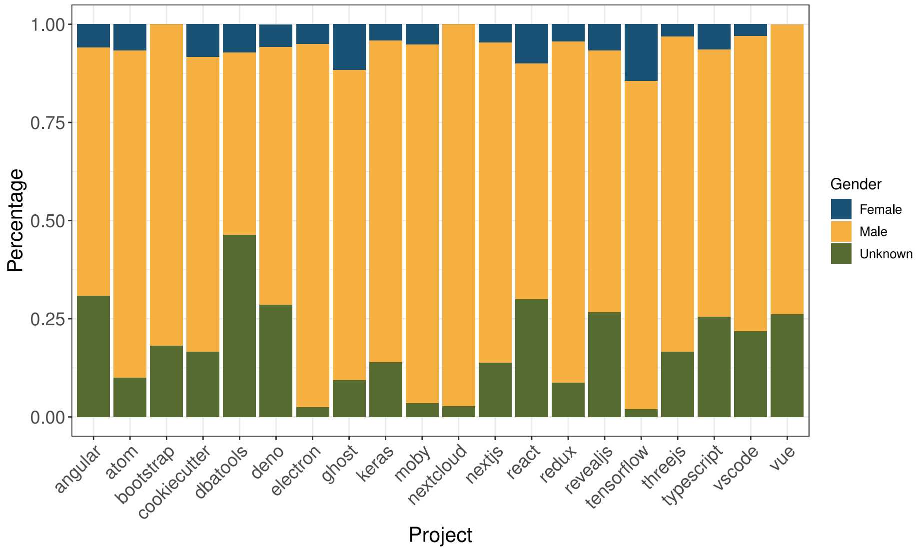

Fig. 7 Distribution of developers in the top 10% of the coreness scale with coreness values derived from the cochange network using eigenvector centrality

6.3 The Top Ten Percent

The results of our analyses of the coreness values and distributions of men and women in OSS projects indicate that the top 10% of developers are still predominately men. As this is a rather consistent result across all of our subject projects and a major conclusion of our study, we take a closer look at this. To do so, we first looked at the composition of developers that have coreness values in the top 10%. In Fig. 7, we show the composition of developers in the top 10% of the coreness scale with coreness values derived from the cochange network using eigenvector centrality.[11] Here, we can see that women only make a small part of the developers and in some cases (e.g., BOOTSTRAP) are completely absent. Men, on the other hand, make up the majority of the developers in the top 10% of the coreness scale. Interestingly, when looking at the composition of the top 10% of developers in the issue network, there is not one project where no women reached the top 10% of the coreness scale. A possible reason is that, as stated before in Section 6.1, women are more often strict non-coders in OSS projects and thus are more active in social interactions and documentation (Robles et al. 2016). To understand the distribution better, we have created violin plots with nested box plots that show the exact distribution of the coreness values of men and women in the top 10%. In Fig. 8, we show the results for the project TENSORFLOW with coreness values derived from the cochange network using eigenvector centrality.[12] Here, we see that the women in the top 10% of the coreness scale, while present, do not reach the highest coreness values. This is consistent across almost all of our subject projects. Nevertheless, there are also a few results that show no difference (apart from the number of men and women) in the top 10% of the

[11] We show the results for the other configurations on our supplementary Website.

[12] We show the results for the other configurations on our supplementary Website.

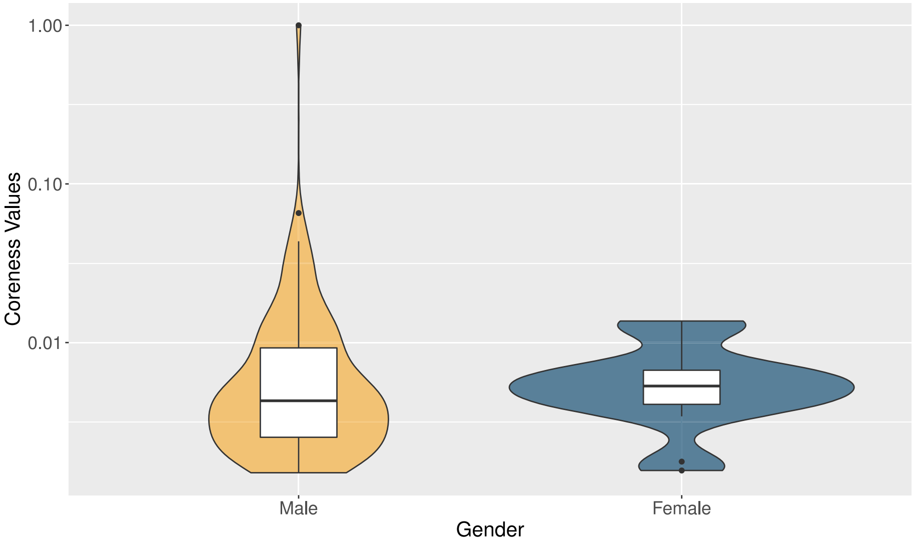

Fig. 8 Violin plots with nested box plots of the coreness values of men and women in the top 10% of the coreness scale with coreness values derived from the cochange network using eigenvector centrality in TENSORFLOW

coreness scale. We show one such example in Fig. 9 for the project ELECTRON with coreness values derived from the issue network using hierarchy. Here, we see that the coreness values of men and women in the top 10% of the coreness scale are not significantly different. This shows, that even though in most cases women do not reach the highest coreness values, there are also cases where they do.

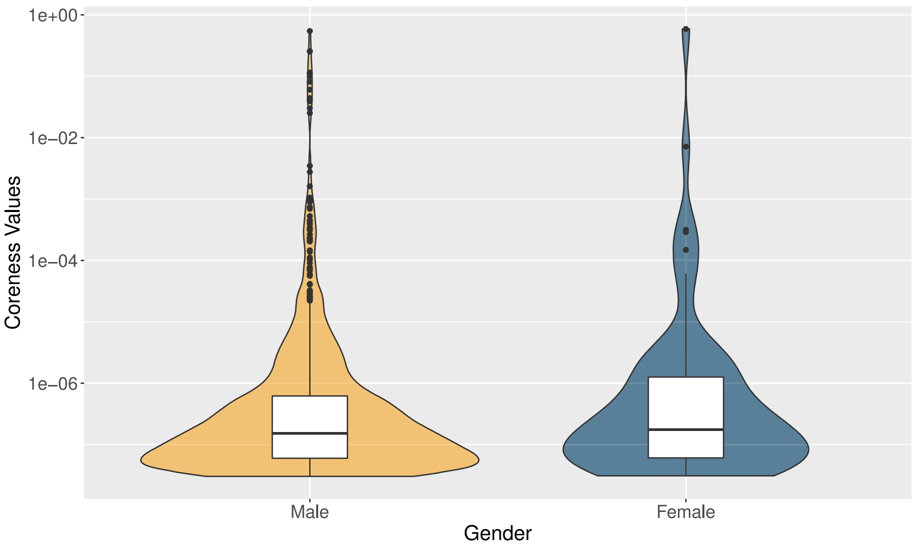

Fig. 9 Violin plots with nested box plots of the coreness values of men and women in the top 10% of the coreness scale with coreness values derived from the issue network using hierarchy in ELECTRON

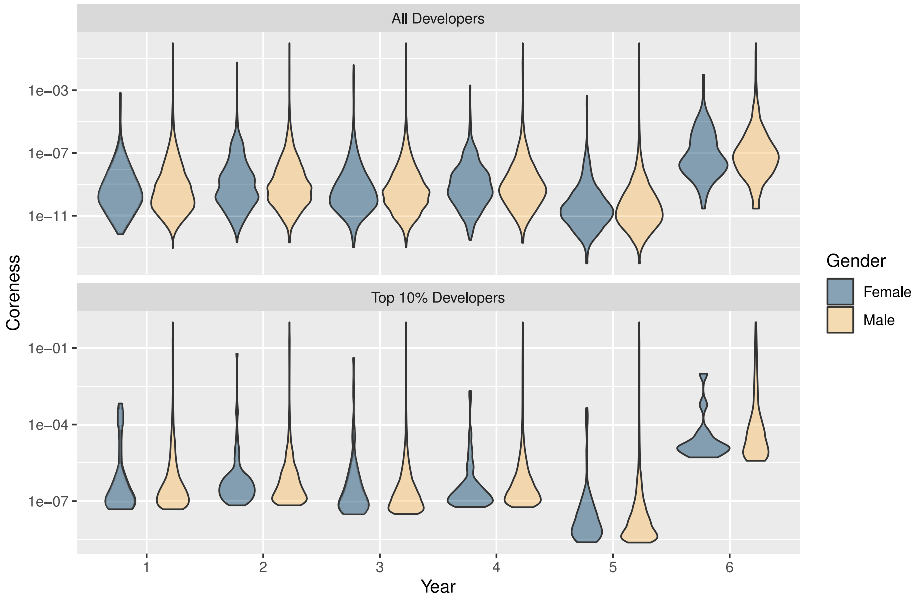

Fig. 10 Evolution of coreness values of men and women in the project VS CODE classified in the issue network with hierarchy. The top plot shows the coreness values of all developers and the bottom plot shows just the top 10% of developers

6.4 Implications of our Results

Our results confirm that there is little to no difference between men and women with regard to general contribution statistics. For their standing in the organizational hierarchy, we find that men tend to reach higher coreness values and seem to be higher up in general in the issue networks. Notably, the difference is not as big as one might expect when looking at the historical underrepresentation of women in STEM (Nimmesgern 2016). A closer look reveals though that, towards the top of the organizational hierarchy, men dominate. This finding is universal across our subject projects and also stable over time. In Fig. 10, we show the evolution of coreness values (all and top 10%) for men and women in VS CODE. While women are present in the top 10% of developers each year, they do not reach as high coreness values as the men. These findings are consistent across all of our subject projects[13]. As a consequence, OSS projects shall target future countermeasures towards the top 10% of developers, to increase peer parity and to motivate more women to join their project.

6.5 Unidentified Developers

Since we have discarded the developers that could not be identified as male or female, we can not say for certain whether our results actually change when these are considered. To mitigate this risk, we have performed an additional analysis with so called "what if" scenarios similar to Vasilescu et al. (2014). These "what if" scenarios are meant to deal with the hypothetical case that all non-classifiable developers are either men or women. With this, we hope to show that even if we include these developers, our assumptions and findings still hold when we

[13] Corresponding plots for all subject projects are available on the supplementary website.

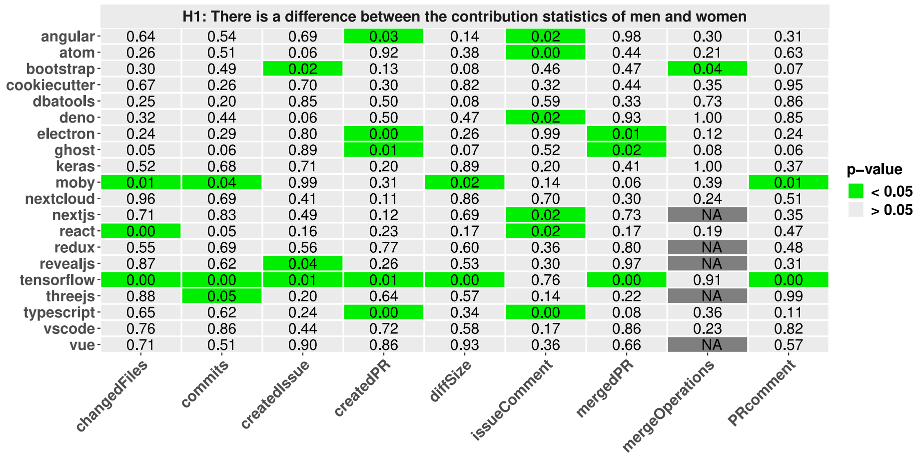

Fig. 11 Results for the Wilcoxon Mann-Whitney U tests of the general contribution statistics if all unknown developers are men. PR means pull request

do not discard these developers. In Fig. 11, we show the results of the general contribution statistics if all unknown developers were men. Here, we can see that the number of statistically significant results actually goes down in comparison to the results in Fig. 2. After further consideration of the results, this indicates that the unknown developers are actually less involved in the development process than men and women, as their contribution statistics pull down the averages of the as men classified developers. Contrary to that, we also see that when the unknown developers would all be women[14], the number of statistically significant results actually increases as the averages of the women are lowered.

For the developer coreness, we perform a similar analysis with the "what if " scenarios. In Fig. 12 we show the results of the analysis if all unknown developers are women. Here, we see that the number of statistically significant results increases drastically. This again is due to the unknown developers having lower coreness values than men and women and thus lowering the averages for both groups if merged with them. This favors the mens' coreness values in this case, as they were already significantly different from the womens' and this gap gets larger through the lowering of the average coreness value of women. Confirming this, when merging the unknown developers with the men, we see that there are drastically fewer statistically significant results[15].

We have also performed the same "what if " scenarios with the multiple linear regression and the correlation analysis. We present these results on our supplementary Website. Here we can see little to no difference.

All of these results taken together, we conclude that the developers that could not be identified are mostly less involved and lower in the organizational hierarchy than the rest of the developers. Nevertheless, we see that while the results do seem to change when including them, it does not change much about our findings for men and women.

[14] These results can be seen on our supplementary website

[15] We show this on our supplementary Website

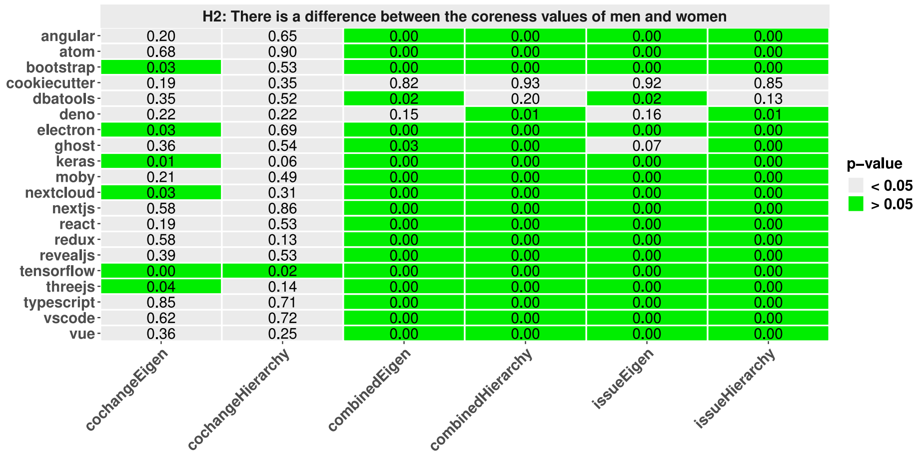

Fig. 12 Results for the Wilcoxon Mann-Whitney U tests of developer coreness if all unknown developers are women. Eigen means eigenvector centrality

6.6 Developers with Gender-Neutral Names

Our definition of the gender of a name is that it is perceived as either male of female. Nevertheless, there are also names that are not primarily associated with either gender. An example of this is the name *Cameron*, which is frequently given to men and women. To further scrutinize the findings of our study, we have performed an additional analysis where we classified developers that have gender-neutral names as their own group. For this purpose, we use a list of the most common gender-neutral names in the United States as reported by *FiveThirtyEight*[16]. The data used for this article is published on GitHub[17]. While we are aware that this selection is biased towards western names and there is no comprehensive list of worldwide gender-neutral names, this analysis still provides a good indication for the robustness of our results.

Equipped with the list of gender neutral names, we re-classify the developers with these names as their own group. In our subject projects, the percentage of developers that could be classified as gender-neutral is, on average, less than 3%. We compared the coreness values of the gender-neutral developers with men and women separately. These results can be seen on our supplementary Website. Overall, we find that the coreness values of gender-neutral developers are not significantly different from the coreness values of men or women. Furthermore, we have also plotted the coreness distribution of gender-neutral developers together with the coreness values of the other three groups of developers (men, women, and unknown) for all subject projects. An example of this is shown in Fig. 13 for the project VS Code with coreness values derived from the issue network with eigenvector centrality. Here, we can see that the coreness value distributions of all four groups are very similar, except for the absolute top of the coreness scale. This shows that, even though gender-neutral names have to be considered, they do not change the results of our study.

16 https://abcnews.go.com/538

17 https://github.com/fivethirtyeight/data/blob/master/unisex-names/unisex_names_table.csv

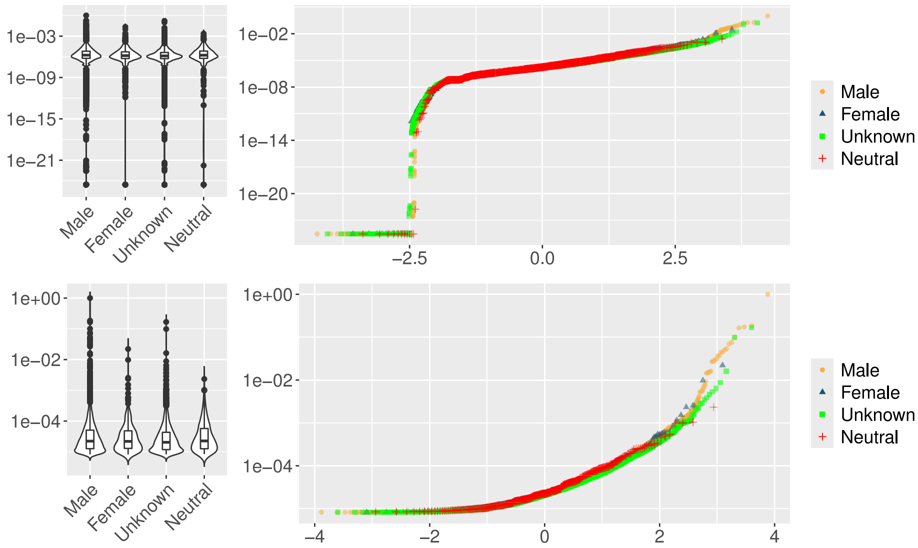

Fig. 13 Coreness value distributions for VS CODE with the coreness values derived from the issue network using eigenvector centrality. In the left column, we show violin plots with nested box plots. In the right column, we show quantile plots. The plots in the upper row are created with all coreness values of the projects and in the lower row we only use the coreness values above the 80th percentile. The y-axes of all plots are the coreness values of developers on a logarithmic scale for better visualization

7 Threats to Validity

Not unexpectedly, our study faces some threats to the validity. In this section, we give an overview of the most relevant threats and how we have addressed them.

7.1 Internal Validity

We use developer networks to capture organizational structures arising in our subject projects. The question is whether these networks do correctly reflect the reality of organizational structures in OSS projects. Previous studies find strong evidence that these networks do accurately reflect the reality (Meneely and Williams 2011), mitigating this risk.

The data we use for our study pose another threat. Specifically, the GitHub and Git data of the projects might be incomplete. Since we use widely used methods to mine our datasets, we achieve comparability to other studies, which use the same or similar methods and therefore lower the risk of false findings.

The biggest threat to internal validity arises from the use of automatic gender detection through developers names. As this is only a heuristic, there is a risk of false classifications. We address this threat by using a well-tested state-of-the-art tool, which was reported to yield highly accurate results (Santamaría and Mihaljević 2018). This lowers the risk of false classification. Furthermore, we have checked a random sample of 170 male and 170 female classified developers (10 per project) to confirm the accuracy. We manually verified the names by applying our common knowledge of usual gender associations of names. In cases where these were not clear, we used a manual Web-search to gather more information. In the sample, we only found 22 misclassified developers (12 men & 10 women), which is well below 10%. Moreover, it is not entirely clear how our chosen name-to-gender tool handles gender-neutral names. To mitigate this risk, we have performed a semi-automated re-evaluation of the developers names using publicly available information about gender-neutral names. Subsequently, we have re-analyzed our data considering gender-neutral developers as their own category. We discuss these results in Section 6.6. There, we find that the developers with gender-neutral names make only a small fraction of the developers in our projects, and their coreness values are not significantly different from the coreness values of men and women. The other problem with automated gender detection are developers that use a pseudonym for their development activities. This results in the high number of unclassified developers as shown in Table 2. This problem can only be solved by asking the people behind the pseudonyms. Since surveys in OSS communities usually yield a low response rate (Smith et al. 20113), we refrain from conducting a survey where the cost outweighs the benefit. However, we have also checked another random sample of 170 unclassified developers with the same method and only found 8 developers that could be manually reclassified. We therefore deem this threat to be sufficiently low for our study. Furthermore, we have addressed the exclusion of these non-classifiable developers with the "What-If "-Scenarios, which we present in Section 6.5.

7.2 External Validity

With our project selection of 20 popular OSS projects, we open up our study to the threat that these projects might not reflect the reality of OSS development. As a mitigation, we selected these projects such that we cover a wide variety of factors: Our projects span across multiple domains, sizes, and project ages. With this, we try to mitigate the risk of reporting a hard-to-generalize study. Moreover, our selection of 20 popular OSS projects yields insightful results without substantial variance, meaning a certain generalizability is present. Furthermore, we only chose projects that use Git repositories hosted on GitHub as their method of development. This bears the risk that we loose the option to generalize to OSS projects that use different development methods such as mailing-list-based development. But since the development through GitHub is one of the most widely used development methods, this threat is not very relevant.

8 Conclusion

We have conducted an empirical study on the influence of perceived developer gender on contributions and standing in OSS projects. In particular, we were interested in whether there are differences between men and women when looking at the general contribution statistics and the position in the organizational hierarchy of OSS projects. Our findings suggest that there is little to no influence of perceived developer gender on their contributions. As for the organizational hierarchy, we found that the distribution of developers along the coreness scale (i.e., their position in the organizational hierarchy) is largely the same. In the top 10% of this scale, however, we found that women are underrepresented when compared to men. Overall, while our results indicate a certain degree of gender balance, the people in leadership positions in OSS projects are still male. As a direct consequence of our results, OSS communities should investigate the apparent difference between male and female developers in the top of the organizational hierarchy. If there is, in fact, a bias against women, this should be addressed to promote a diverse and inclusive project culture, which is usually beneficial for project success as diverse points of view can help in developing complex software. An actionable step would be to direct future countermeasures against gender imbalance toward the top of the organizational hierarchy.

Author Contributions All authors contributed to the study conception and design. Material preparation, data collection and analysis were performed by Christian Hechtl. The first draft of the manuscript was written by Christian Hechtl and all authors commented on and helped revise previous versions of the manuscript. All authors read and approved the final manuscript.

Funding Open Access funding enabled and organized by Projekt DEAL. This work was supported by the German Research Foundation (Grant Number: AP 206/14-1).

Data Availability We provide a replication package including anonymized raw data, scripts, and tooling for our analyses on our website (https://se-sic.github.io/paper-perceived-gender/).

Declarations

Conflict of Interest The authors declared that they have no conflict of interest.

Ethical Approval This declaration is not applicable for this study.

Informed Consent This declaration is not applicable for this study.

References

Argamon S, Koppel M, Fine J, Shimoni A (2003) Gender, genre, and writing style in formal written texts. Text and Talk 23(3):321–346

Boccaletti S, Latora V, Moreno Y, Chavez M, Hwang DU (2006) Complex networks: Structure and dynamics. Phys Rep 424(4):175–308

Riehle D (2019) The innovations of open source. Comput 52(4):59–63

Brandes U, Erlebach T (2005) Network analysis: Methodological foundations. Springer

Calvo D (2021) The (in) visible barriers to free software: Inequalities in online communities in Spain. Stud Commun Sci 21(1):163–178

Padala S, Mendez C, Dias L, Steinmacher I, Steine Hanson Z, Hilderbrand C, Horvath A, Hill C, Simpson LD, Burnett M, Gerosa M, Sarma A (2022) How gender-biased tools shape newcomer experiences in OSS projects. IEEE Trans Softw Eng (TSE) 48(2):241–259

Prana GAA, Ford D, Rastogi A, Lo D, Purandare R, Nagappan N, (2021) Including everyone, everywhere: Understanding opportunities and challenges of geographic gender-inclusion in OSS. IEEE Trans Softw Eng (TSE). https://ieeexplore.ieee.org/abstract/document/9466393

Bosu A, Sultana KZ (2019) Diversity and inclusion in open source software (OSS) projects: Where do we stand? in Proc int symposium empirical software engineering and measurement (ESEM). IEEE, 2019:1–11

Fedorov V, Mannino F, Zhang R (2009) Consequences of dichotomization. Biopharm Stat J Appl Stat Pharm Ind 8(1):50–61

Frluckaj H, Dabbish L, Widder DG, Qiu HS, Herbsleb JD (2022) Gender and participation in open source software development. ACM Trans Human-Comput Interact 6:1–31
Moghaddam B, Yang MH (2002) Learning gender with support faces. IEEE Trans Patt Anal Mach Intell 24(5):707–711
Hoogendoorn S, Oosterbeek H, van Praag M (2013) The impact of gender diversity on the performance of business teams: Evidence from a field experiment. Manag Sci 59(7):1514–1528
Mislove A, Lehmann S, Ahn YY, Onnela JP, Rosenquist J (2011) Understanding the demographics of twitter users. in Proc int conf on web and social media (ICWSM), The AAAI Press, pp 554–557
Argamon S, Koppel M, Fine J, Shimoni A (2003) Gender, genre, and writing style in formal written texts. Text and Talk 23(3):321–346
Gousios G, Spinellis D (2017) Mining software engineering data from GitHub. in Proc int conf on software engineering companion (ICSE-C), IEEE, pp 501–502
Kalliamvakou E, Gousios G, Blincoe K, Singer L, German D, Damian D, (2014) The promises and perils of mining GitHub. in proc int workshop mining software repositories (MSR) ACM, pp 92–101
Joblin M, Mauerer W, Apel S, Siegmund J, Riehle D (2015) From Developer Networks to Verified Communities: A Fine-grained Approach. in proc int conf software engineering (ICSE) ACM/IEEE, pp 563–573
Joblin M, Apel S, Hunsen C, Mauerer W (2017a) Classifying Developers into Core and Peripheral: An empirical study on count and network metrics. in proc int conf software engineering (ICSE) ACM/IEEE,pp 164–174
Joblin M, Apel S (2022) How do successful and failed projects differ? A socio-technical analysis. ACM Trans Softw Eng Methodol (TOSEM) 31(4)
Dabbish L, Stuart C, Tsay J, Herbsleb J (2012) Social coding in GitHub: Transparency and collaboration in an open software repository. in proc int conf computer-supported cooperative work (CSCW), ACM, pp 1277–1286
López-Fernández L, GRobles, Gonzalez-Barahona J, Herraiz I (2009) Applying social network analysis techniques to community-driven libre software projects. Int J Inf Technol Web Eng (IJITWE) 1(3):27–48
Yang X (2014) Social network analysis in open source software peer review. in proc int symposium on foundations of software engineering (FSE), ACM, pp 820–822
Crowston K, Wei K, Li Q, Howison J (2006) Core and periphery in free/libre and open source software team communications. in proc Hawaii int conf system sciences (HICSS), IEEE, pp 118–127
Terceiro A, Rios L, Chavez C (2010) An empirical study on the structural complexity introduced by core and peripheral developers in free software projects. in proc Brazilian symposium on software engineering (SBES), IEEE, pp 21–29
Moghaddam B, Yang MH (2002) Learning gender with support faces. IEEE Trans Patt Anal Mach Intell 24(5):707–711
Nimmesgern H (2016) Why Are Women Underrepresented in STEM Fields? Chem Eur J 22(11):3529–3530
Joblin M, SApel, and WMauerer () Evolutionary trends of developer coordination: A network approach. Empir Softw Eng 22(4):2050–2094
Padala S, Mendez C, Dias L, Steinmacher I, Steine Hanson Z, Hilderbrand C, Horvath A, Hill C, Simpson LD, Burnett M, Gerosa M, Sarma A (2022) How gender-biased tools shape newcomer experiences in OSS projects. IEEE Trans Softw Eng (TSE) 48(2):241–259
Qiu HS, Nolte A, Brown A, Serebrenik A, Vasilescu B (2019) Going farther together: The impact of social capital on sustained participation in open source. In proc int conf software engineering (ICSE), ACM/IEEE, pp 688–699
Trinkenreich B, Wiese I, Sarma A, Gerosa M, Steinmacher I (2022) Women's participation in open source software: A survey of the literature. ACM Trans Softw Eng and Methodol (TOSEM) 31(4):1–37
Riehle D (2019) The innovations of open source. Comput 52(4):59–63
Robles G, Reina L, González-Barahona J, Domínguez S (2016) Women in free/libre/open source software: The situation in the 2010s. Springer, In IFIP int conf on open source systems, pp 163–173
Robles G, Steinmacher I, Adams P, Treude C (2019) Twen-ty Years of Open Source Software: From Skepticism to Mainstream. IEEE Softw 36(6):12–15
Santamaría L, Mihaljević H (2018) Comparison and benchmark of name-to-gender inference services. PeerJ Comput Sci 4:e156
Shameer S, Rodríguez-Pérez G, Nagappan M (2023) Relationship between diversity of collaborative group members' race and ethnicity and the frequency of their collaborative contributions in GitHub. Empir Softw Eng 28(4):83
Kuechler V, Gilbertson C, Jensen C (2012) Gender Differences in Early Free and Open Source Software Joining Process. In int conf open source systems (OSS), Springer,pp 78–93
Calvo D (2021) The (in) visible barriers to free software: Inequalities in online communities in Spain. Stud Commun Sci 21(1):163–178

Singh V (2019) Women-only spaces of open source. In int workshop on gender equality in software engineering (GE), ACM/IEEE, pp 17–20
Terrell J, Kofink A, Middleton J, Rainear C, Murphy-Hill E, Parnin C, Stallings J (2017) Gender differences and bias in open source: Pull request acceptance of women versus men. PeerJ Comput Sci 3:e111
Oliva G, Santana F, de Oliveira K, de Souza C, Gerosa M (2012) Characterizing key developers: A case study with apache ant. In proc int conf collaboration and technology (CRIWG), Springer, pp 97–112
Trinkenreich B, Wiese I, Sarma A, Gerosa M, Steinmacher I (2022) Women's participation in open source software: A survey of the literature. ACM Trans Softw Eng and Methodol (TOSEM) 31(4):1–37
Vasilescu B, Capiluppi A, Serebrenik A (2014) gender, representation and online participation: A quantitative study. Interact Comput 26(5):488–511
Vasilescu B, Filkov V, Serebrenik A (2015) Perceptions of diversity on GitHub: A user survey. In proc int workshop cooperative and human aspects of software engineering (CHASE), IEEE, pp 50–56
Vasilescu B, Capiluppi A, Serebrenik A (2014) gender, representation and online participation: A quantitative study. Interact Comput 26(5):488–511
Meneely A, Williams L (2011) Socio-technical developer networks: Should we trust our measurements? In proc int conf software engineering (ICSE), ACM/IEEE, pp 281–290
Smith E, Loftin R, Murphy-Hill E, Bird C, Zimmermann T () Improving developer participation rates in surveys. In proc int workshop cooperative and human aspects of software engineering (CHASE), IEEE, pp 89–92