Productivity Spillovers among Knowledge Workers in Agglomerations: Evidence from GitHub

Lena Abou El-Komboz[1], Thomas Fackler[2]

August 23, 2022

[1]ifo Institute and LMU Munich
[2]ifo Institute, LMU Munich and CESifo

Paper in a Nutshell

Paper in a Nutshell

- **Research Question:** Do high-tech workers become more active when being surrounded by a higher number of other high-tech workers? If so, is there heterogeneity in the effect?
- **Setting:** Relate the activity of GitHub (GH) users with cluster size and estimate the effect of cluster size on user activity; instrument changes in local cluster size by changes originating elsewhere for the most skilled users
- **Data:** Activity stream of GitHub users provided by GHTorrents (Gousios, 2013) over 2015-2021
- **Results:** Positive and significant elasticity between user activity and cluster size of 0.2402 (0.1134)

Research Question

Research Question

- **Open Source Software activity** positively **impact local firm productivity** (Nagle, 2019) and **entrepreneurship** (Wright et al., 2020)
- Work could almost completely be done remotely, however was found to **spatially cluster** - similar to other innovative activity (Wachs et al., 2022)
- GitHub: Worlds **biggest open source online platform** (Lima et al., 2014)
- **Commit:** any code modification
- Provide evidence on agglomeration effects for this type of knowledge worker

Literature Overview

Effects of Exposure to Innovators on Productivity

- Azoulay et al. (2010) explore peer effects in the field of life sciences and find a persistent decline in quality-adjusted publication output of co-authors after the death of a 'superstar' scientist
- Catalini (2018) show that colocation increases the likelihood of collaborations which in turn also tend to work on riskier research

Agglomeration Effects on Productivity

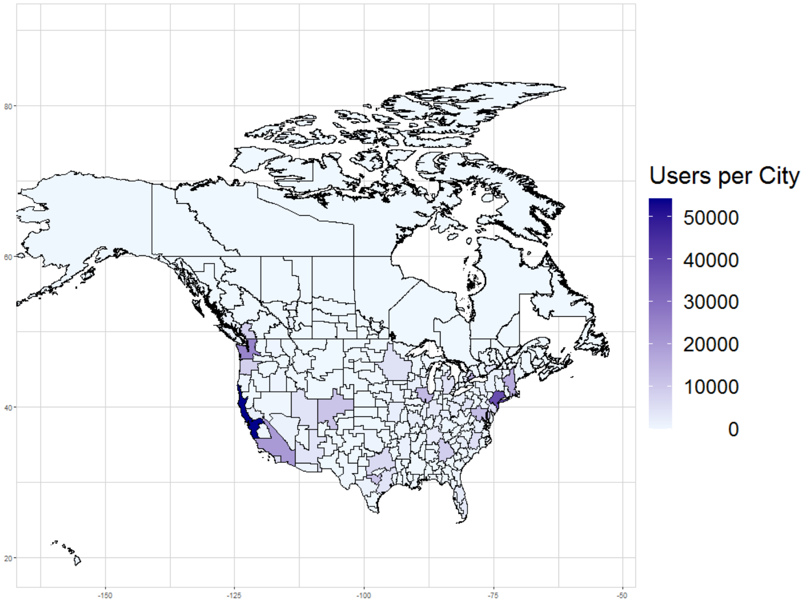
- Seminal paper by Jaffe et al. (1993): Strong localization of patent citations, similarly Atkin et al. (2022) present results on the importance of face-to-face interactions for citation activity
- Moretti (2021) estimate positive elasticity between cluster size and productivity using patent data

Data

GitHub Data

- Combined **10 snapshots** by GitHub Torrents (GHTorrents): 2015/09/25 (201509), 2016/01/08 (201601), 2016/06/01 (201606), 2017/01/19 (201701), 2017/06/01 (201706), 2018/01/01 (201801), 2018/11/01 (201811), 2019/06/01 (201906), 2020/07/17 (202007) and 2021/03/06 (202103) (Gousios, 2013)
- Filter for users **always located in the US or Canada** and with commits in at least two time intervals or, if account created in latest snapshot used, in that time interval User Map Economic Areas
- Consider **18 programming languages** that cover 90 percent of all commits
- **10,785,249 user-snapshot observations with 404,651 US or CA users** User Data User Moves

GitHub Users in the Snapshot 202103



Clusters on GitHub

Cluster Definition

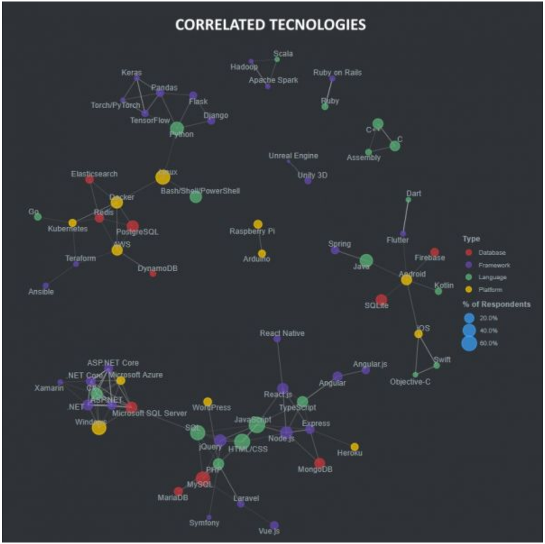
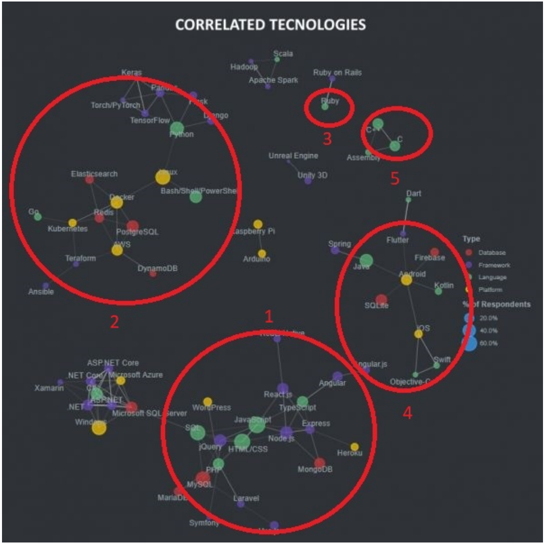


Figure: https://insights.stackoverflow.com/survey/2020#correlated-technologies

Cluster Definition



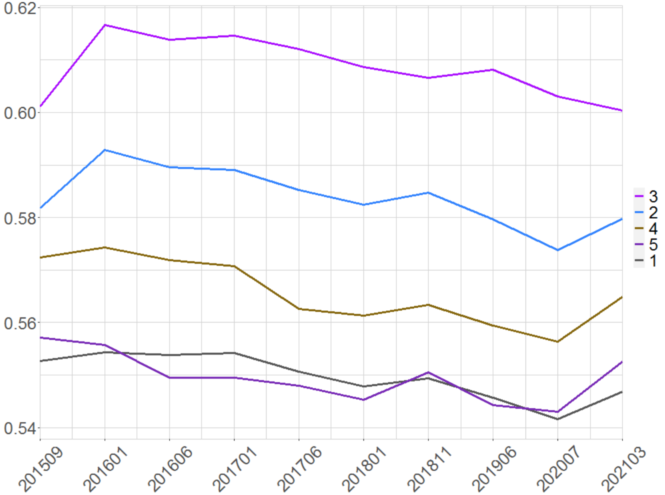
Clusters

Cluster size is calculated as:

$$S_{-ifct} = \frac{\sum_{j \neq i} N_{jfct}}{\sum N_{jft}}$$

- S_{-ifct}: cluster size of user i in city c in technology f in time t, excluding user i
- N_{fct}: number of users j in city c in technology f in time t
- N_{jft}: number of users j in technology f in time t

Share of Top 10 Cities for all Clusters



Clusters Descriptives

Share per City

Empirical Approach

Estimation Strategy

$$ln(y_{ijflct}) = \alpha \; ln(S_{-ifct}) + d_{cf} + d_{cl} + d_{lt} + d_{ct} + d_c + d_f + d_t + d_l + d_i + d_j + \mu_{ijflct} \quad (1)$$

- y_{ijflct}: number of commits of user i in time interval t to project j located in city c in the technology f and programming language l
- S_{-ifct}: cluster size in city c of the technology f in time interval t, excluding user i
- d_{cf}: city $\times$ technology effects
- d_{cl}: city $\times$ programming language effects
- d_{lt}: programming language $\times$ time effects
- d_{ct}: city $\times$ time effects
- d_c: city effects
- d_f: technology effects
- d_t: time effects
- d_l: programming language effects
- d_i: individual effects
- d_j: project effects

Results

Baseline Estimates

	Log(Commit)				
	(1)	(2)	(3)	(4)	(5)
Log(Size)	0.1052	0.2553*	0.2524*	0.1887**	0.2402**
	(0.1206)	(0.1407)	(0.1429)	(0.0766)	(0.1134)
Fixed-effects					
City	Yes	Yes	Yes	Yes	Yes
Time	Yes	Yes	Yes	Yes	Yes
Language	Yes	Yes	Yes	Yes	Yes
Technology	Yes	Yes	Yes	Yes	Yes
Project	Yes	Yes	Yes	Yes	Yes
User	Yes	Yes	Yes	Yes	Yes
City x Technology		Yes	Yes	Yes	Yes
City x Language			Yes	Yes	Yes
Language x Time				Yes	Yes
City x Time					Yes
Adjusted R^2	0.284	0.284	0.284	0.287	0.288
Observations	2,238,606	2,238,606	2,238,606	2,238,606	2,238,606

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1.*

Notes: Standard errors are clustered by city x technology. Every column presents a regression.

Instrumental Variable Estimates - 2SLS

Instrumental Variable Approach - 2SLS

$$IV_{ifct} = \sum_{s \neq j_i} D_{sfc(t-1)} \frac{\Delta N_{sf(-c)t}}{\Delta N_{ft}} \tag{2}$$

- $D_{sfc(t-1)}$: indicator if project s in technology f was present in city c in time interval $t-1$
- $N_{sf(-c)t}$: log sum of users committing to project s in technology f, time interval t in all cities but city c to which user i does not commit to
- N_{ft}: log total sum of users in time interval t in technology f

First Differences

$$\Delta ln(y_{ifct}) = \alpha \Delta ln(S_{-ifct}) + d_{ct} + d_{lt} + d_l + d_j + \mu_{ijflct} \quad (3)$$

- $\Delta ln(y_{ifct})$: change in log number of commits of user i in time interval t to project j located in city c in the technology f and programming language l
- $\Delta ln(S_{-ifct})$: change in log cluster size in city c of the technology f in time interval t, excluding user i
- d_{ct}: city $\times$ time effects
- d_{lt}: programming language $\times$ time effects
- d_l: programming language effects
- d_j: project effects

First Differences - IV Sample

	Δ Log(Commit)		
	(1)	(2)	(3)
Δ Log(Size)	0.0341**	0.0956*	0.0956*
	(0.0159)	(0.0500)	(0.0500)
Fixed-effects			
Language x Time	Yes	Yes	Yes
City x Time	Yes	Yes	Yes
Project		Yes	Yes
Language			Yes
Observations	68,694	68,694	68,694

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1.*

Notes: Standard errors are clustered by city x technology. Every column presents a regression. The sample consists of commits to projects, that receive commits in two consecutive time intervals and users in the upper fourth quartile of the follower per user distribution. The dependant variable is the change in the log of commits to a project between two consecutive time intervals. The model estimated is equation (3).

First Differences - Full Sample | Baseline Estimates - IV Sample

Instrumental Variable Estimates - 2SLS

	Δ Log(Commit) (1)	Δ Log(Commit) (2)	Δ Log(Commit) (3)
First Stage	0.00002***	0.00002***	0.00002***
	(0.00001)	(0.00001)	(0.00001)
Δ Log(Size)	0.85540**	0.89009**	0.89009**
	(0.38402)	(0.43522)	(0.43531)
Fixed-effects			
Language x Time	Yes	Yes	Yes
City x Time	Yes	Yes	Yes
Project		Yes	Yes
Language			Yes
Observations	68,694	68,694	68,694
F-test (1st stage)	62.86	73.84	73.81
Wu-Hausman, p-value	0.07	0.12	0.12

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1.*

Notes: Standard errors are clustered by city. Every column presents a regression. The sample consists of commits to projects, that receive commits in two consecutive time intervals and users in the upper fourth quartile of the follower per user distribution. The dependant variable is the change in the log of commits to a project between two consecutive time intervals. The model estimated is equation (3).

Reduced Form

Heterogeneity Analysis

Heterogeneity in User Productivity

	Log(Commit) (1)
First Quartile (Least Productive)	0.2294**
	(0.1115)
Second Quartile	0.2310**
	(0.1126)
Third Quartile	0.2503**
	(0.1162)
Fourth Quartile (Most Productive)	0.2649**
	(0.1150)
Adjusted R^2	0.288
Observations	2,238,606
Wald (joint nullity), p-value	0.133

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1.*

Notes: Standard errors are clustered by city x technology. User productivity is mesured by the total number of commits per user. The most productive users are users that are in the fourth quartile of the distribution of total commits per user. In all regressions, fixed effects for city, time, programming language, city $\times$ programming language, programming language $\times$ time, city $\times$ technology, technology, city $\times$ time, project and user are included.

Alternative User Samples

Heterogeneity in Cluster Size

	Log(Commit)
	(1)
First Quartile (Smallest)	0.2362**
	(0.1122)
Second Quartile	0.2318**
	(0.1132)
Third Quartile	0.2350**
	(0.1154)
Fourth Quartile (Largest)	0.2146*
	(0.1224)
Adjusted R^2	0.288
Observations	2,238,605
Wald (joint nullity), p-value	0.281

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1.*

Notes: Standard errors are clustered by city x technology. In all regressions, fixed effects for city, time, programming language, city × programming language, programming language × time, city × technology, technology, city × time, project and user are included.

Heterogeneity by Share of Commits made during Business Hours

	Log(Commit)
	(1)
First Quartile (Leisure)	0.2429**
	(0.1121)
Second Quartile	0.2457**
	(0.1143)
Third Quartile	0.2431**
	(0.1134)
Fourth Quartile (Business)	0.2101*
	(0.1135)
Adjusted R^2	0.288
Observations	2,238,606
Wald (joint nullity), p-value	0.020

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1.*

Notes: Standard errors are clustered by city x technology. Business commits are commits created during work days (monday through friday) and work hours (6am untill 8pm). The projects with the largest share of business commits received are projects that are in the fourth quartile of the distribution of business commits per project. In all regressions, fixed effects for city, time, programming language, city × programming language, programming language × time, city × technology, technology, city × time, project and user are included.

Heterogeneity in Project Age

	Log(Commit)
	(1)
First Quartile (Youngest)	0.2241**
	(0.1107)
Second Quartile	0.2307**
	(0.1138)
Third Quartile	0.2392**
	(0.1153)
Fourth Quartile (Oldest)	0.2557**
	(0.1149)
Adjusted R^2	0.288
Observations	2,238,606
Wald (joint nullity), p-value	0.040

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1.*

Notes: Standard errors are clustered by city x technology. Project Age is measured by the years after project start until the date of the latest snapshot, 2021-03-06. The oldest projects are projects that are in the fourth quartile of the distribution of project years. In all regressions, fixed effects for city, time, programming language, city × programming language, programming language × time, city × technology, technology, city × time, project and user are included.

Mechanism

Local vs. Spatially distributed Projects

	Log(Commit)	
	Distributed	Local
	(1)	(2)
Log(Size)	0.1926*	0.2234***
	(0.1143)	(0.0836)
Adjusted R^2	0.364	0.279
Observations	778,668	1,459,938

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1.*

Notes: Standard errors are clustered by city x technology. Every column presents a regression. Locality of a project is measured by the share of users stemming from different cities. A project with a share of 1 means, that all users committing to the project stem from one city. The sample is split by 1, i.e. projects with all users stemming from one city in comparison to projects with more geographically distributed users committing to. In all regressions, fixed effects for city, time, programming language, city $\times$ programming language, programming language $\times$ time, city $\times$ technology, technology, city $\times$ time, project and user are included.

Small and Large Projects

	Log(Commit)	
	Small	Large
	(1)	(2)
Log(Size)	0.2336**	0.7292**
	(0.1171)	(0.3385)
Adjusted R^2	0.299	0.411
Observations	2,176,020	62,586

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1.*

Notes: Standard errors are clustered by city x technology. Every column presents a regression. For every project the total number of users committing to the project over the whole time was calculated. Small projects are projects with at most 40 total users, large projects have more than 40 users in total committing to. In all regressions, fixed effects for city, time, programming language, city × programming language, programming language × time, city × technology, technology, city × time, project and user are included.

Others and Own Projects

	Log(Commit)	
	Others	Own
	(1)	(2)
Log(Size)	0.2314*	0.0923
	(0.1343)	(0.1471)
Adjusted R^2	0.320	0.312
Observations	1,261,215	977,391

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1.*

Notes: Standard errors are clustered by city x technology. Every column presents a regression. Ownership of a project is determined if author id equals project owner id. In all regressions, fixed effects for city, time, programming language, city $\times$ programming language, programming language $\times$ time, city $\times$ technology, technology, city $\times$ time, project and user are included.

Leisure and Business Projects

	Log(Commit)	
	Leisure	Business
	(1)	(2)
Log(Size)	0.2366**	0.2840
	(0.1102)	(0.4828)
Adjusted R^2	0.327	-0.389
Observations	1,752,610	485,996

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1.*

Notes: Standard errors are clustered by city x technology. Every column presents a regression. For every project the share of commits during business hours, i.e. Monday through Friday from 6am to 8pm was calculated. A project is identified as a business project with a share of 1, all commits are made during business hours. In all regressions, fixed effects for city, time, programming language, city × programming language, programming language × time, city × technology, technology, city × time, project and user are included.

Concluding Remarks

Concluding Remarks

- **Positive and significant relationship** between cluster size and number of commits of 0.2402 (0.1134)
- Heterogeneity by project type: **Older projects** and **leisure projects** have a **larger elasticity**
- **Contemporaneous effect** of a change in cluster size on productivity in first differences model with an IV approach **for most skilled users larger**
- Effect mainly driven by commits to **others'** projects, **local** projects and **leisure** projects

References & Appendix

Atkin, D., Chen, M. K., and Popov, A. (2022). The returns to face-to-face interactions: Knowledge spillovers in silicon valley. Technical report, National Bureau of Economic Research.

Azoulay, P., Graff Zivin, J. S., and Wang, J. (2010). Superstar extinction. *The Quarterly Journal of Economics*, 125(2):549–589.

Catalini, C. (2018). Microgeography and the direction of inventive activity. *Management Science*, 64(9):4348–4364.

Gousios, G. (2013). The ghtorrent dataset and tool suite. In *Proceedings of the 10th Working Conference on Mining Software Repositories*, page 233–236, San Francisco. IEEE Press.

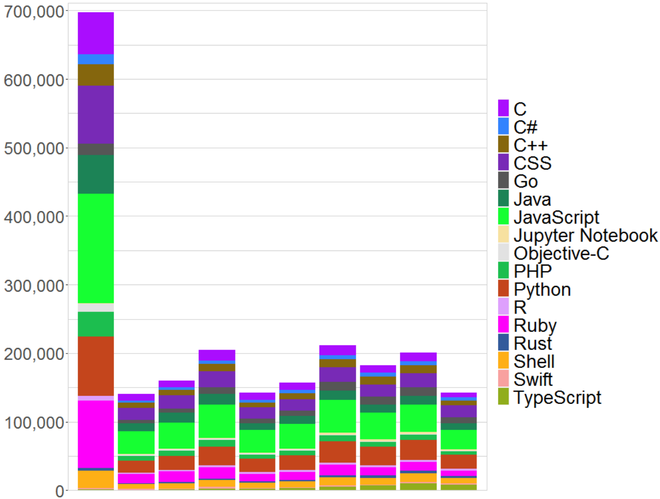
Jaffe, A. B., Trajtenberg, M., and Henderson, R. (1993). Geographic localization of knowledge spillovers as evidenced by patent citations. *The Quarterly Journal of Economics*, 108.

Lima, A., Rossi, L., and Musolesi, M. (2014). Coding together at scale: Github as a collaborative social network. *arXiv preprint arXiv:1407.2535*.

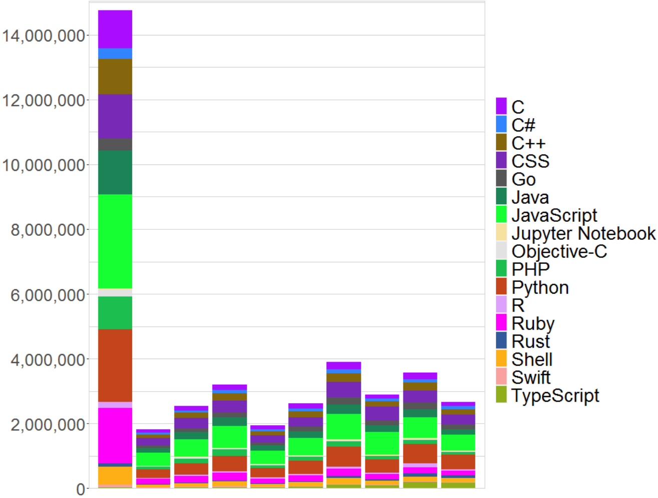
Moretti, E. (2021). The effect of high-tech clusters on the productivity of top inventors. *National Bureau of Economic Research Working Paper*, (w26270).

Nagle, F. (2019). Open source software and firm productivity.

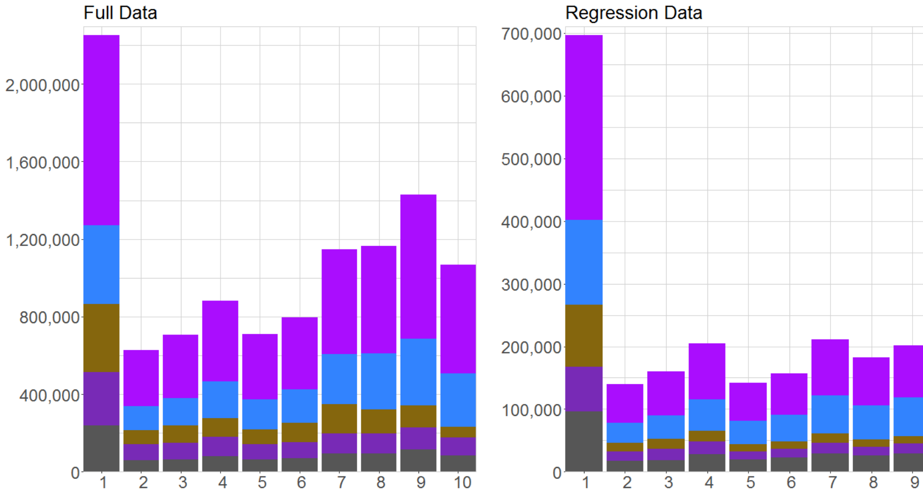
Number of Observation per Snapshot and per Programming Language



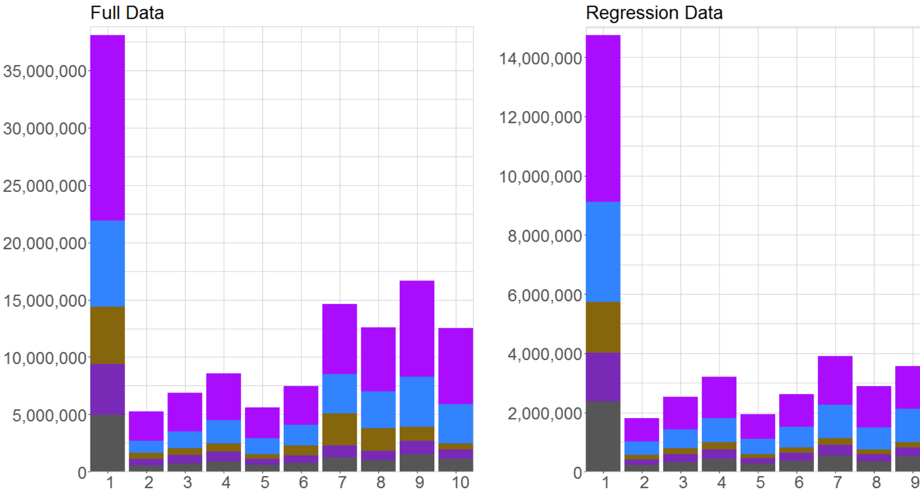
Number of Commits per Snapshot and per Programming Language



Number of Observation per Snapshot and per Technology



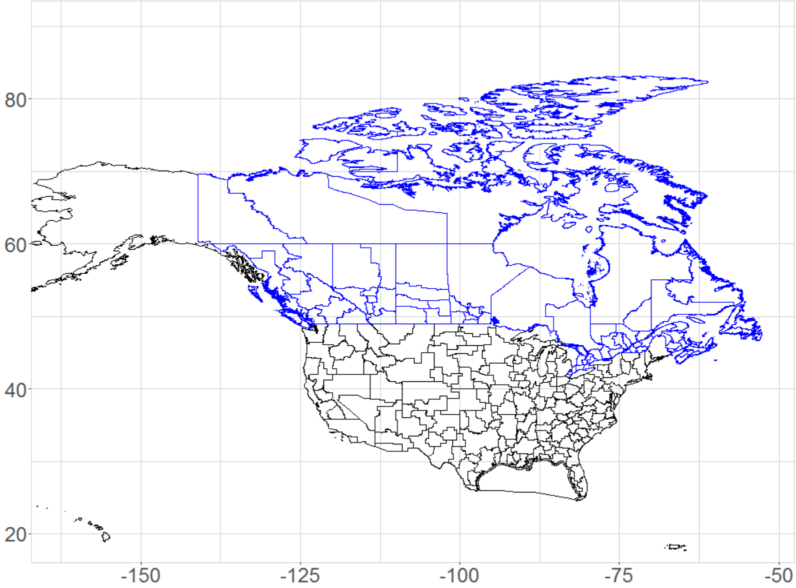
Number of Commits per Snapshot and per Technology



User Moves

- Moves: Out of the 445,230 users in the full data, 92,510 users moved in sum 39,617 times.
- 70,862 users moved once, 18,577 users moved twice, 2,963 users moved three time, 106 users moved four times and two users moved five times.
- Moves occurred in the second time interval 8,324 times, 2,299 times in the third time interval, 16,079 times in the fourth time interval, 24,956 times in the seventh time interval and 65,681 times in the tenth time interval.

US and CA Economic Areas



Baseline Estimates - Excluding Zero Star Projects

	Log(Commit)				
	(1)	(2)	(3)	(4)	(5)
Log(Size)	0.2101	0.4700*	0.4710*	0.3823*	0.5968*
	(0.2233)	(0.2815)	(0.2832)	(0.2251)	(0.3298)
Fixed-effects					
City	Yes	Yes	Yes	Yes	Yes
Time	Yes	Yes	Yes	Yes	Yes
Language	Yes	Yes	Yes	Yes	Yes
Technology	Yes	Yes	Yes	Yes	Yes
Project	Yes	Yes	Yes	Yes	Yes
User	Yes	Yes	Yes	Yes	Yes
City x Technology		Yes	Yes	Yes	Yes
City x Language			Yes	Yes	Yes
Language x Time				Yes	Yes
City x Time					Yes
Adjusted R^2	0.419	0.422	0.422	0.424	0.422
Observations	73,926	73,926	73,926	73,926	73,926

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1.*

Notes: Standard errors are clustered by city x technology. Every column presents a regression. Sample includes only projects with at least 100 stars.

Baseline Estimates - Excluding Large Commits and Large Projects

	Log(Commit)				
	(1)	(2)	(3)	(4)	(5)
Log(Size)	0.0932	0.2228	0.2213	0.1533*	0.2490**
	(0.1089)	(0.1485)	(0.1497)	(0.0850)	(0.1103)
Fixed-effects					
City	Yes	Yes	Yes	Yes	Yes
Time	Yes	Yes	Yes	Yes	Yes
Language	Yes	Yes	Yes	Yes	Yes
Technology	Yes	Yes	Yes	Yes	Yes
Project	Yes	Yes	Yes	Yes	Yes
User	Yes	Yes	Yes	Yes	Yes
City x Technology		Yes	Yes	Yes	Yes
City x Language			Yes	Yes	Yes
Language x Time				Yes	Yes
City x Time					Yes
Adjusted R^2	0.253	0.254	0.254	0.257	0.258
Observations	2,113,098	2,113,098	2,113,098	2,113,098	2,113,098

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1.*

Notes: Standard errors are clustered by city x technology. Every column presents a regression. Sample includes only projects with less than 40 users committing to and commits to projects less than 100.

Baseline Estimates - with Technology × Time FE

	Log(Commit)					
	(1)	(2)	(3)	(4)	(5)	(6)
Log(Size)	0.1052	0.2553*	0.2524*	0.1887**	0.2402**	0.2402**
	(0.1206)	(0.1407)	(0.1429)	(0.0766)	(0.1134)	(0.1134)
Fixed-effects						
City	Yes	Yes	Yes	Yes	Yes	Yes
Time	Yes	Yes	Yes	Yes	Yes	Yes
Language	Yes	Yes	Yes	Yes	Yes	Yes
Technology	Yes	Yes	Yes	Yes	Yes	Yes
Project	Yes	Yes	Yes	Yes	Yes	Yes
User	Yes	Yes	Yes	Yes	Yes	Yes
City x Technology		Yes	Yes	Yes	Yes	Yes
City x Language			Yes	Yes	Yes	Yes
Language x Time				Yes	Yes	Yes
City x Time					Yes	Yes
Technology x Time						Yes
Adjusted R^2	0.284	0.284	0.284	0.287	0.288	0.288
Observations	2,238,606	2,238,606	2,238,606	2,238,606	2,238,606	2,238,606

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1.*

Notes: Standard errors are clustered by city x technology. Every column presents a regression.

Different Lengths of Observation Period

	Log(Commit)				
	1	2	3	4	5
	(1)	(2)	(3)	(4)	(5)
Log(Size)	0.1511	0.1507	0.1411	0.1455	0.1712*
	(0.0976)	(0.0972)	(0.0936)	(0.0925)	(0.0907)
Adjusted R^2	0.178	0.184	0.226	0.245	0.255
Observations	5,677,085	5,640,879	4,761,734	4,180,116	3,655,988

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1.*

	Log(Commit)				
	6	7	8	9	10
	(1)	(2)	(3)	(4)	(5)
Log(Size)	0.1842**	0.1735*	0.1870*	0.2080**	0.2402**
	(0.0921)	(0.0938)	(0.0966)	(0.1011)	(0.1135)
Adjusted R^2	0.262	0.267	0.273	0.278	0.288
Observations	3,355,702	3,119,287	2,873,106	2,656,882	2,238,606

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1.*

Notes: Standard Errors are clustered by city x technology. Every column presents a regression of equation 1. In column 1, users are included that in at least one time interval had commits. In column 2, users are included that commit in at least two time intervals, and so on.

First Differences - Full Sample

	Δ Log(Commit)		
	(1)	(2)	(3)
Δ Log(Size)	0.0138	0.0154	0.0154
	(0.0097)	(0.0181)	(0.0181)
Fixed-effects			
Language x Time	Yes	Yes	Yes
City x Time	Yes	Yes	Yes
Project		Yes	Yes
Language			Yes
Adjusted R^2	0.103	0.018	0.018
Observations	290,363	290,363	290,363

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1.*

Notes: Standard errors are clustered by city x technology. Every column presents a regression. The sample consists of commits to projects, that receive commits in two consecutive time intervals. The dependant variable is the change in the log of commits to a project between two consecutive time intervals. The model estimated is equation (3).

Baseline Estimates - IV Sample

	Log(Commit)
	(1)
Log(Size)	-0.2003
	(0.4737)
Fixed-effects	
City	Yes
Time	Yes
Language	Yes
Technology	Yes
Project	Yes
User	Yes
City x Technology	Yes
City x Language	Yes
Language x Time	Yes
City x Time	Yes
Adjusted R^2	0.501
Observations	68,694

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1.*

Notes: Standard errors are clustered by city x technology. Every column presents a regression. The sample consists of commits to projects, that receive commits in two consecutive time intervals and users in the upper fourth quartile of the follower per user distribution.

Reduced Form

	Δ Log(Commit)			
	(1)	(2)	(3)	(4)
First Stage	-0.000050***	0.000010***	0.000010***	0.000017***
	(0.000016)	(0.000003)	(0.000003)	(0.000006)
Fixed-effects				
Project	Yes	Yes	Yes	Yes
Time		Yes	Yes	Yes
Language			Yes	Yes
Language x Time				Yes
F-test (projected), p-value	0.000	0.017	0.017	0.003
F-test (projected)	148.284	5.679	5.679	8.939
R^2	0.350	0.403	0.403	0.404
Observations	254,582	254,582	254,582	254,582

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1.*

Notes: Standard Errors are clustered by city. Every column presents a regression. The sample includes commits to projects, that have commits in two consecutive time intervals. It is the same sample as used for the first differences estimates.

Clusters based on Regression Data

	Log(Commit)					
	(1)	(2)	(3)	(4)	(5)	(6)
Log(Size)	0.0636	0.0850	0.0886	0.0735	0.0652	0.0599
	(0.0526)	(0.0635)	(0.0641)	(0.0513)	(0.0481)	(0.0451)
Fixed-effects						
City	Yes	Yes	Yes	Yes	Yes	Yes
Time	Yes	Yes	Yes	Yes	Yes	Yes
Language	Yes	Yes	Yes	Yes	Yes	Yes
Technology	Yes	Yes	Yes	Yes	Yes	Yes
Project	Yes	Yes	Yes	Yes	Yes	Yes
User	Yes	Yes	Yes	Yes	Yes	Yes
Technology-City		Yes	Yes	Yes	Yes	Yes
City x Language			Yes	Yes	Yes	Yes
Technology-Time				Yes	Yes	Yes
Language x Time					Yes	Yes
City x Time						Yes
R^2	0.701	0.701	0.702	0.703	0.704	0.705
Observations	2,223,556	2,223,556	2,223,556	2,223,556	2,223,556	2,223,556

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1.*

Standard Errors are clustered by city x technology.

Heterogeneity Gender

	Log(Commit)
	(1)
Female × Log(Size)	0.2143
	(0.1315)
Male × Log(Size)	0.2021
	(0.1231)
Adjusted R^2	0.295
Observations	1,832,422
Wald (joint nullity), p-value	0.257

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1.*

Notes: Standard errors are clustered by city x technology. In all regressions, fixed effects for city, time, programming language, city × programming language, programming language × time, city × technology, technology, city × time, project and user are included.

Absolute Cluster Size

	Log(Commit)				
	(1)	(2)	(3)	(4)	(5)
Log(Absolute Cluster Size)	-0.2491***	-0.2920***	-0.2924***	0.1887**	0.2402**
	(0.0659)	(0.0760)	(0.0763)	(0.0766)	(0.1134)
Fixed-effects					
City	Yes	Yes	Yes	Yes	Yes
Time	Yes	Yes	Yes	Yes	Yes
Language	Yes	Yes	Yes	Yes	Yes
Technology	Yes	Yes	Yes	Yes	Yes
Project	Yes	Yes	Yes	Yes	Yes
User	Yes	Yes	Yes	Yes	Yes
City x Technology		Yes	Yes	Yes	Yes
City x Language			Yes	Yes	Yes
Language x Time				Yes	Yes
City x Time					Yes
Adjusted R^2	0.284	0.285	0.285	0.287	0.288
Observations	2,238,606	2,238,606	2,238,606	2,238,606	2,238,606

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1.*

Notes: Standard errors are clustered by city x technology. Every column presents a regression.

Summary Statistics - Full Data

Variable	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
Length User Observed	1	5	9	7.46	10	10
Commits per User	1	25	78	317.08	236	388,287
Commit per Project per Snapshot	1	1	3	14.20	9	364,392
Stars per Project	0	0	0	29.30	0	259,118
Stars per Project - Star $>$ 0 and non-forked Projects	1	1	4	171.18	24	259,118
Forks per Project	0	0	0	3.56	0	145,997
Forks per Project - Forks $>$ 0 and non-forked Projects	1	1	2	28.80	7	145,997
Programming Language per City	3	16	18	16.45	18	18
Programming Language per City per Snapshot	1	8	13	12.02	16	18
Technology per City	1	5	5	4.87	5	5
Technology per City per Snapshot	1	4	5	4.27	5	5
Programming Language per User	1	3	4	4.25	5	18
Programming Language per User per Snapshot	1	1	2	2.12	3	17
Technology per User	1	1	2	2.24	3	5
Technology per User per Snapshot	1	1	1	1.68	2	5
Own Project	0	0	1	0.73	1	1
Business Share	0	0	1	0.59	1	1
Weekend Share	0	0	0	0.21	0	1
Out of Hour Share	0	0	0	0.32	1	1
Local Share	0	1	1	0.97	1	1
Users per Project	1	1	1	1.24	1	324,321
Project Age (in Years)	0	2	4	3.97	6	13

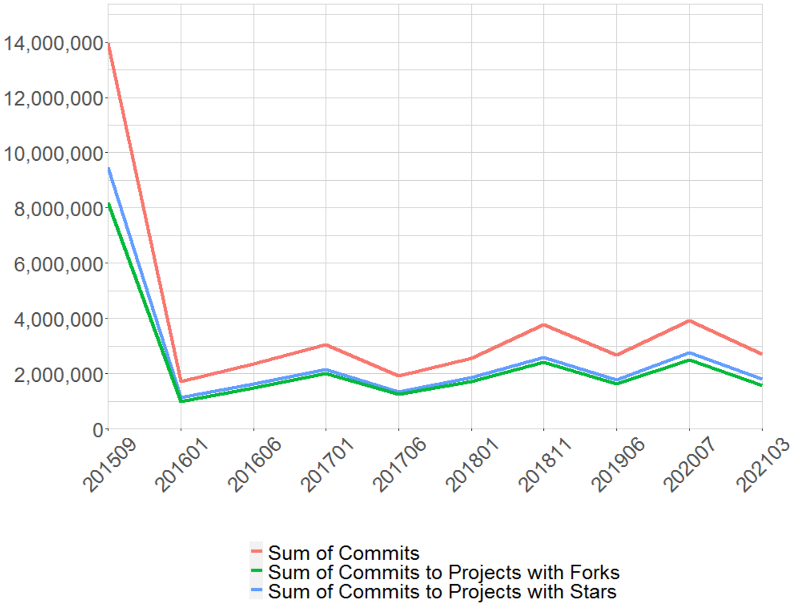
Summary Statistics - Regression Data

Variable	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
Length User Observed	10	10	10	10.00	10	10
Commits per User	25	536	1,089	2,171.40	2,336	235,640
Commit per Project per Snapshot	1	1	3	17.83	10	169,209
Stars per Project	0	0	0	88.01	2	259,118
Stars per Project - Star $>$ 0 and non-forked Projects	1	2	9	295.29	58	259,118
Forks per Project	0	0	0	11.22	0	145,997
Forks per Project - Forks $>$ 0 and non-forked Projects	1	1	3	53.33	14	145,997
Programming Language per City	1	11	16	13.28	17	17
Programming Language per City per Snapshot	1	5	10	9.46	14	17
Technology per City	1	5	5	4.55	5	5
Technology per City per Snapshot	1	3	4	3.62	5	5
Programming Language per User	1	5	6	6.53	8	17
Programming Language per User per Snapshot	1	2	3	3.25	4	16
Technology per User	1	3	4	3.45	4	5
Technology per User per Snapshot	1	1	2	2.26	3	5
Own Project	0	0	1	0.50	1	1
Business Share	0	0	1	0.62	1	1
Weekend Share	0	0	0	0.19	0	1
Out of Hour Share	0	0	0	0.31	0	1
Local Share	0	1	1	0.90	1	1
Users per Project	1	1	1	1.66	1	2,145
Project Age (in Years)	0	3	5	5.00	7	13

Summary Statistics - Programming Language

Language	Min.	Median	Mean	Max.	Projects	N	Commits	Share
C	1	19	340.67	60,823	64,278	7,510	2,558,443	6.41%
C#	1	20	322.03	18,116	30,697	3,473	1,118,404	2.8%
C++	1	23	364.61	56,647	57,220	7,557	2,755,341	6.9%
CSS	1	79	276.28	225,948	155,878	15,840	4,376,271	10.97%
Go	1	20	290.65	24,648	54,765	5,561	1,616,284	4.05%
Java	1	25	392.21	221,308	87,435	8,392	3,291,425	8.25%
JavaScript	1	128	519.07	172,663	323,606	15,571	8,082,383	20.25%
Jupyter Notebook	1	17	116.88	9,212	12,299	2,723	318,267	0.8%
Objective-C	1	10	118.05	9,075	17,207	3,176	374,918	0.94%
PHP	1	23	357.26	218,260	56,859	6,065	2,166,770	5.43%
Python	1	64	471.43	41,771	167,390	12,884	6,073,841	15.22%
R	1	27	415.98	73,039	17,409	1,551	645,181	1.62%
Ruby	1	36	354.62	48,734	127,808	9,412	3,337,658	8.36%
Rust	1	19	218.47	41,287	16,602	2,326	508,171	1.27%
Shell	1	26	150.57	23,715	56,690	10,943	1,647,734	4.13%
Swift	1	15	154.21	30,077	11,639	1,857	286,362	0.72%
TypeScript	1	15	142.87	20,600	26,873	5,224	746,332	1.87%

Commits by Project



Alternative User Samples

	Log(Commit)			
	Upper 25%	Upper 50%	Upper 75%	All
	(1)	(2)	(3)	(4)
Log(Size)	0.5437*	0.2866*	0.2484**	0.2402**
	(0.2776)	(0.1468)	(0.1192)	(0.1134)
Adjusted R^2	0.382	0.321	0.294	0.288
Observations	766,968	1,589,540	2,115,306	2,238,606

*Signif. Codes: ***: 0.01, **: 0.05, *: 0.1.*

Notes: Standard errors are clustered by city x technology. Every column presents a regression. Controls for city, time, language, city $\times$ language, language $\times$ time, user, city $\times$ technology, technology, city $\times$ time and project are included. Users are measured by their share of commits to all commits. Hence, users in the upper 25% sample cover the upper 25% of all commits by their commits.