

Generative AI and Distributed Work

Evidence from Open Source Software

Manuel Hoffmann¹ Sam Boysel¹ Frank Nagle¹
Sida Peng² Kevin Xu³

¹Harvard Business School, Harvard University

²Microsoft Corporation

³GitHub Inc.



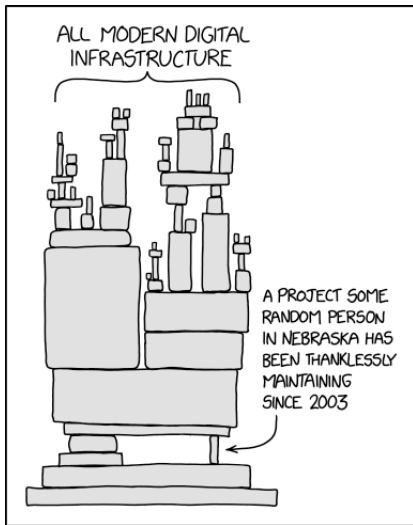
**Harvard
Business
School**

Laboratory for
Innovation Science
at Harvard



Jobs in the Age of Artificial Intelligence
Georgetown, USA – September 4, 2024

The linchpin problem in the production of digital goods



Source: XKCD [PG](#) [XZ Utils](#)

Highlights

How does AI adoption change distributed work patterns?

Context: Maintainers of Open Source Software projects

Natural experiment: A discontinuity in GitHub's Copilot program to offer a free coding AI to "top maintainers"

Main results

- Task reallocation: Coding ↑, project management ↓
- Lower ability maintainers benefit more
- Effect persists for 2 years

Distributed work and the issue of expert reliance

- ⇒ Linchpin problem: more pronounced for **public goods**
 - Example: free **open source software** (OSS) Contributor Funnel
 - $\approx 50\%$ projects rely on single maintainer (Avelino et al. 2017)
 - 96% of value from 5% of developers (Hoffmann et al. 2024)
- ⇒ Further worsened for **distributed work**

Distributed work and the issue of expert reliance

- ⇒ Linchpin problem: more pronounced for **public goods**
 - Example: free **open source software** (OSS) [Contributor Funnel](#)
 - $\approx 50\%$ projects rely on single maintainer (Avelino et al. 2017)
 - 96% of value from 5% of developers (Hoffmann et al. 2024)
- ⇒ Further worsened for **distributed work**
- ⇒ Work in a distributed manner
 - **Digital goods** produced with scarce human capital
 - Heightened salience since pandemic
 - Possible due to **lower communication costs** [Details](#)

Distributed work and the issue of expert reliance

- ⇒ Linchpin problem: more pronounced for **public goods**
 - Example: free **open source software** (OSS) [Contributor Funnel](#)
 - $\approx 50\%$ projects rely on single maintainer (Avelino et al. 2017)
 - 96% of value from 5% of developers (Hoffmann et al. 2024)
- ⇒ Further worsened for **distributed work**
- ⇒ Work in a distributed manner
 - **Digital goods** produced with scarce human capital
 - Heightened salience since pandemic
 - Possible due to **lower communication costs** [Details](#)
- ⇒ The **linchpin problem** under **distributed work**
 - By reducing communication costs on the many, we impose a time burden on the few linchpins (e.g. experts, managers)

A work dichotomy in the information economy



Core Work

Management

A work dichotomy in the information economy

Programmer

Core Work

Management

A work dichotomy in the information economy

Programmer



Core Work

Coding

Management

A work dichotomy in the information economy

Programmer



Coding



Project Management

A work dichotomy in the information economy

Programmer



Coding

Real <<< Ideal



Project Management

Real >>> Ideal

Workload

FOSS Contributor Survey (Nagle et al. (2020))

Can technology ameliorate the linchpin problem?

- ⇒ **New technology:** Artificial Intelligence (AI) may allow us to substitute away from the scarce resource

- ⇒ During **open source software** programming, **AI** can ...
 - ... relieve developers from repetitive work
 - ... substitute advice from other developers (Copilot)
 - ... increase efficiencies (reduce frictions)
 - ... improve learning

Research Question

Broadly: How does AI change the nature of distributed work?

Narrowly: (for open source software production)

1. How does AI change task allocation?
2. How can AI relax the linchpin constraint?

Going under the tip of the iceberg: the contribution

- ⇒ **AI in many areas:** Chat GPT as writing (Noy and Zhang 2023), customer support (Brynjolfsson et al. 2023), consulting (Dell'Acqua et al. 2023), startup assistance (Otis et al. 2024)
- ⇒ **AI in open source:** GitHub Copilot programming assistance
 - Correlational study (Dohmke et al. 2023)
 - Experimental productivity study (Peng et al. 2023)

Going under the tip of the iceberg: the contribution

- ⇒ **AI in many areas:** Chat GPT as writing (Noy and Zhang 2023), customer support (Brynjolfsson et al. 2023), consulting (Dell'Acqua et al. 2023), startup assistance (Otis et al. 2024)
- ⇒ **AI in open source:** GitHub Copilot programming assistance
 - Correlational study (Dohmke et al. 2023)
 - Experimental productivity study (Peng et al. 2023)

- ⇒ **Our contribution**
 - Beyond productivity effects: nature of work
 - AI impact on private provision of public goods
 - Long-term causal evidence of AI from a real-world scenario

- ⇒ **Natural experiment:** Copilot AI on GitHub for top developer
 - Work activities from OSS platform GitHub
 - Local treatment effects via regression discontinuity design

- ⇒ The *de facto* hub for collaborative OSS development
 - Founded in April 2008
 - allows for the private provision of public good
 - time-stamped history of coding

- ⇒ Granular data on distributed work with others on
 - coding and
 - project management

The GitHub Platform: Coding



The screenshot shows the GitHub repository page for `tukaani-project / xz`. The repository is public and has 17 watchers, 37 forks, and 485 stars. The main content area displays a list of files and folders with their commit history. The right sidebar provides information about the repository, including the source code link, license information, activity, and releases.

Repository Information:

- Repository: `tukaani-project / xz` (Public)
- Watch: 17
- Fork: 37
- Star: 485
- Branches: 15
- Tags: 54
- Commits: 2,544

File List:

File/Folder	Commit Message	Time
<code>.github</code>	CI: Don't require po4a on Solaris	2 weeks ago
<code>build-aux</code>	Fix versionsh compatibility with Solaris	2 weeks ago
<code>cmake</code>	Add SPDX license identifier into 0BSD source code files.	4 months ago
<code>debug</code>	debug/translation.bash: Remove an outdated test command	last month
<code>doc</code>	Fix typos	last week
<code>dos</code>	DOS: Omit useless defines from config.h	2 months ago
<code>doxygen</code>	Doxygen: update-doxygen: Support out-of-tree builds	2 months ago
<code>extra</code>	Add SPDX license identifiers to GPL, LGPL, and FSFULLR files.	4 months ago
<code>lib</code>	Add SPDX license identifiers to GPL, LGPL, and FSFULLR files.	4 months ago
<code>m4</code>	Build: Update visibility.m4 from Gnulib	last month
<code>po</code>	Translations: Run "make -C po update-po"	2 weeks ago
<code>po4a</code>	Translations: Run po4a/update-po	2 weeks ago
<code>src</code>	xz: Fix white space	3 days ago

Right Sidebar:

- About**
- XZ Utils
- Source: tukaani.org/xz/
- Tags: `c`, `cli`, `library`, `compression`
- Readme
- Unknown and 3 other licenses found
- Security policy
- Activity
- Custom properties
- 485 stars
- 17 watching
- 37 forks
- Report repository
- Releases 13**
- XZ Utils 5.6.2 (stable) (Latest) - 2 weeks ago
- + 12 releases
- Contributors 22**

The GitHub Platform: Coding



tukaani-project / xz

Search: Type / to search

Code Issues 7 Pull requests 2 Actions Security Insights

Commits

master

All users All time

Commits on Jul 13, 2024

liblzma: Tweak a comment
Laihu committed 3 days ago ✓ 8 / 8 7c292d6

Commits on Jul 11, 2024

CMake: Bump maximum policy version to 3.30
Laihu committed 5 days ago ✓ 8 / 8 6480eda

CMake: Require CMake 3.20 or later
Laihu committed 5 days ago 9231c30

Commits on Jul 9, 2024

Update THANKS
Laihu committed last week ✓ 8 / 8 828185d

Commits on Jul 6, 2024

xz: Remove the TODO comment about --recursive
Laihu committed last week ✓ 8 / 8 ba8cf81

The GitHub Platform: Project Management



tukaani-project / xz

<> Code Issues 8 Pull requests 3 Actions Security Insights

Want to contribute to tukaani-project/xz? Dismiss

If you have a bug or an idea, browse the open issues before opening a new one. You can also take a look at the [Open Source guide](#).

Response to backdoor incident
#103 by thesamesam was closed 2 weeks ago
Closed 41

Filters Labels 14 Milestones 0 New issue

8 Open ✓ 39 Closed	Author	Label	Projects	Milestones	Assignee	Sort
tsan also needs sanitizer nerf for crc64 #122 opened 2 weeks ago by nate-thrivedave						3
no_sanitize_address isn't required #112 opened on Apr 19 by nigeltao						7
Enable sponsorship on your repo #105 opened on Apr 10 by kaapee						6
[Feature Request]: Is there a real-world benchmark for xz? #83 opened on Feb 24 by svenha						6
Where can I download Latest compiled binaries ? #81 opened on Feb 18 by vectors						5

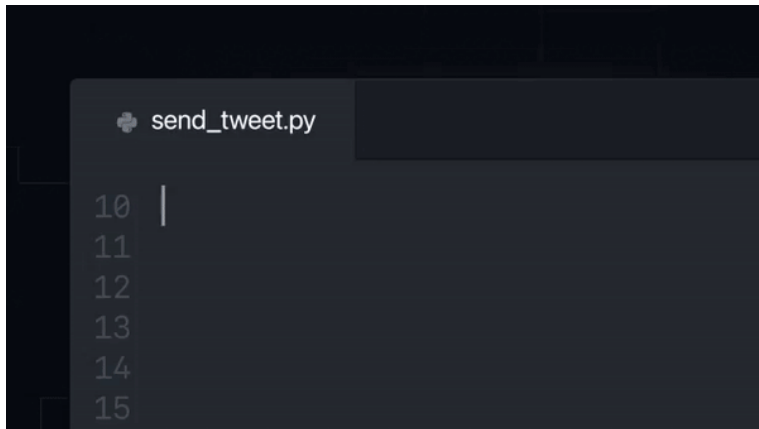
Institutional setting: the GitHub Copilot AI

Copilot Generative AI

LLM to assist programmers to code faster, solve problems more quickly, and learn code that they previously did not know.

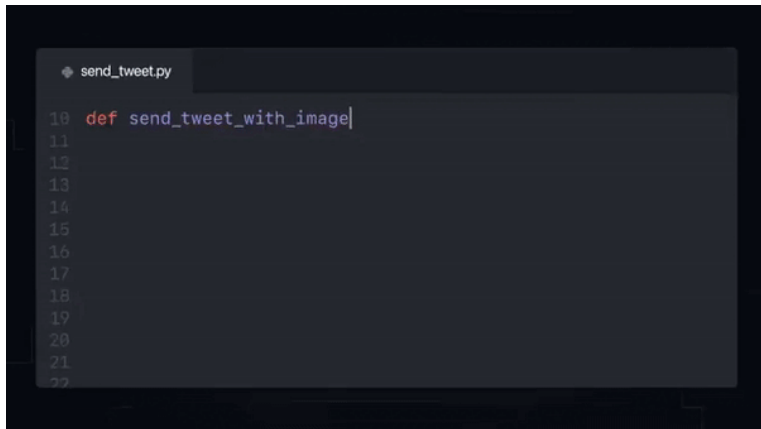
- ⇒ **AI built by** OpenAI and Microsoft/GitHub
- ⇒ **Based on** models underlying ChatGPT
 - Here: Generative Pre-trained Transformer 3 series (GPT-3)
- ⇒ **LLM**. Next Word Prediction (Text Completion).
 - Next Code Prediction (Code Completion)
 - Programmers obtain code-snippets while coding

GitHub Copilot: Artificial Intelligence in Action



```
send_tweet.py  
  
10 |  
11  
12  
13  
14  
15
```

GitHub Copilot: Artificial Intelligence in Action




The image shows a dark-themed code editor window with a file named 'send_tweet.py'. The code is a Python function definition for 'send_tweet_with_image'. The function signature is 'def send_tweet_with_image|' and the cursor is positioned at the end of the line. The editor shows line numbers from 10 to 22.

```
10 def send_tweet_with_image|
11
12
13
14
15
16
17
18
19
20
21
22
```

GitHub Copilot: Artificial Intelligence in Action

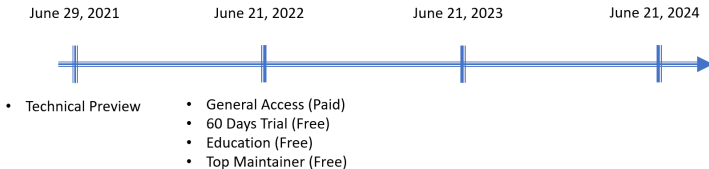
```
◆ send_tweet.py

10 def send_tweet_with_image(message, image):
11     """Send a tweet with an image attached"""
12     # Twitter authentication
13     auth = tweepy.OAuthHandler(CONSUMER_KEY, CONSUMER_SECRET)
14     auth.set_access_token(ACCESS_TOKEN, ACCESS_TOKEN_SECRET)
15     api = tweepy.API(auth)
16
17     # Send the tweet with the image
18     api.update_with_media(image, status=message)
19
20
21
22
```

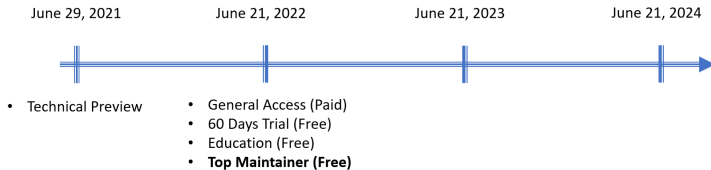


Details

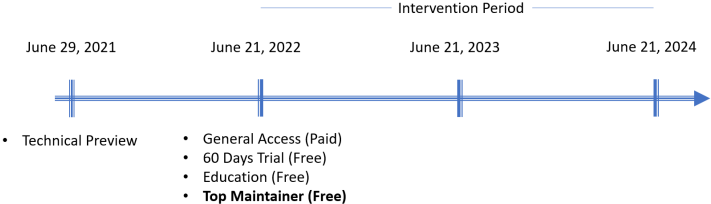
GitHub Copilot AI deployment timeline



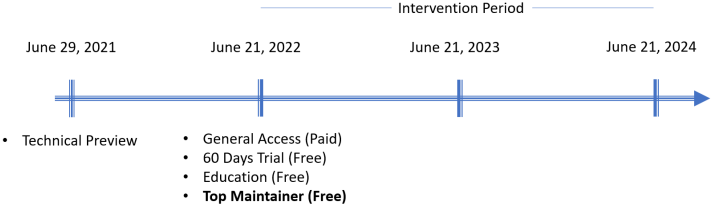
GitHub Copilot AI deployment timeline



GitHub Copilot AI deployment timeline



GitHub Copilot AI deployment timeline



Main Outcomes: Coding & Project Management

GitHub Copilot top maintainer natural experiment

- ⇒ **GitHub Goal:** Reward top open source maintainer
- ⇒ Provide *free access* to Copilot AI for top X maintainers
- ⇒ Internal ranking (R_i) at project (repository) level

$$\text{Eligible} = \begin{cases} \text{AI Free Access for Top Maintainer,} & \text{if } R_i < 0. \\ \textit{else}, & \text{if } R_i \geq 0. \end{cases}$$

- ⇒ Identical maintainer just above and below the threshold

No manipulation of the top maintainer ranking

As a developer of what kind of open source project can continue to use the co-pilot for free? #19754

Unanswered

XiaoYingYo asked this question in Copilot



XiaoYingYo on Jun 30, 2022

...

As a developer of what kind of open source project can continue to use the co-pilot for free?
Fork number of times?
Star number of times?



Category

Copilot

Labels

Copilot Product Feedback

6 participants



Notifications

Subscribe

You're not receiving notifications from this thread.

5 comments

Oldest Newest Top



thomscoder on Jun 30, 2022

...

I do not think there is a full fledged list of projects.
It's probably a combination of factors: stars, forks, contributors, used by etc...

I'd say a size of the impact the Organization would receive if you were to be "slowed down" by a 10\$/month fee



0 replies

Write a reply



D7EAD on Jun 30, 2022

...

Hi!

GitHub Copilot is available for free, as of right now, to verified students and "popular open source projects." What Github defines as a popular open source project is, sadly, not expressly stated.




0 replies

Write a reply



No manipulation of the top maintainer ranking

As a developer of what kind of open source project can continue to use the co-pilot for free? #19754

Unanswered · XiaoYingYo asked this question in Copilot

 XiaoYingYo on Jun 30, 2022

As a developer of what kind of open source project can continue to use the co-pilot for free?
Fork number of times?
Star number of times?

 5 

5 comments

Oldest Newest Top

 thomscoder on Jun 30, 2022


I do not think there is a full fledged list of projects.
It's probably a combination of factors: stars, forks, contributors, used by etc..

I'd say a size of the impact the Organization would receive if you were to be "slowed down" by a 10\$/month fee

 1 




0 replies

Write a reply

 D7EAD on Jun 30, 2022

Hi!

Github Copilot is available for free, as of right now, to verified students and "popular open source projects." What Github defines as a popular open source project is, sadly, not expressly stated.

 3  2 



0 replies

Write a reply

Category

 Copilot

Labels

 Copilot  Product Feedback

6 participants




Notifications

 Subscribe




You're not receiving notifications from this thread.

No manipulation of the top maintainer ranking

 D7EAD on Jun 30, 2022 ...

Hi!

Github Copilot is available for free, as of right now, to verified students and "popular open source projects." What Github defines as a popular open source project is, sadly, not expressly stated.

 3   2 0 replies

Write a reply

Identification: regression discontinuity design

Baseline Model

$$Y_{it} = \alpha_0 + \alpha_1 \mathbb{1}Eligible_{it} + \alpha_2 R_{it} + \alpha_3 Eligible_{it} \times R_{it} + \epsilon_{it}$$

s.t.

$Y = \{Copilot, Activity\}$ (First stage, ITT)

$k = \{i, p\}$ (Individual, Project)

Identifying Assumption

Outcomes change at the threshold due to AI only

Open and Proprietary GitHub Data

⇒ Maintainers (i) observed over time (t)

- Task Allocation

$$Y_{it} = \frac{(\text{cumulative activity } x)_{it}}{(\text{total cumulative activity})_{it}}$$

where $x \in \{\text{coding, project management}\}$

- Copilot usage
- Top maintainer ranking

⇒ Balanced maintainer-week panel

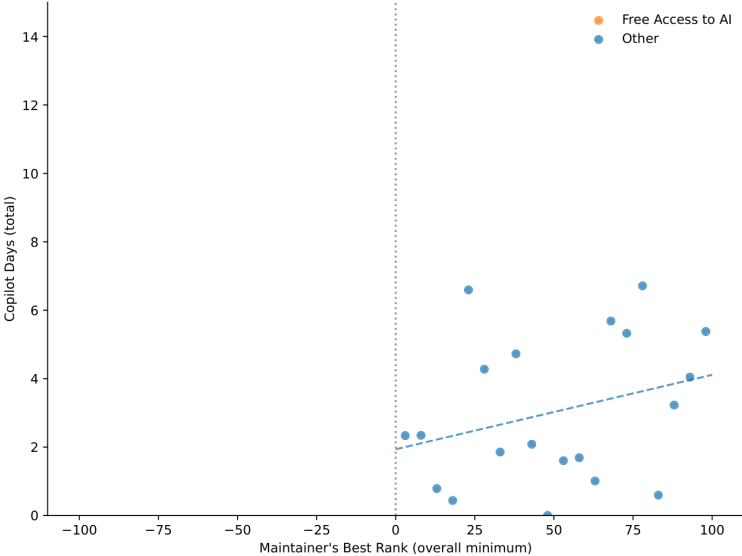
- Over 187k maintainers with 6 mill. observations
- Final sample: 6,885 maintainers with 269,546 obs.

Descriptive Statistics

Restrictions

⇒ Other: project level outcomes

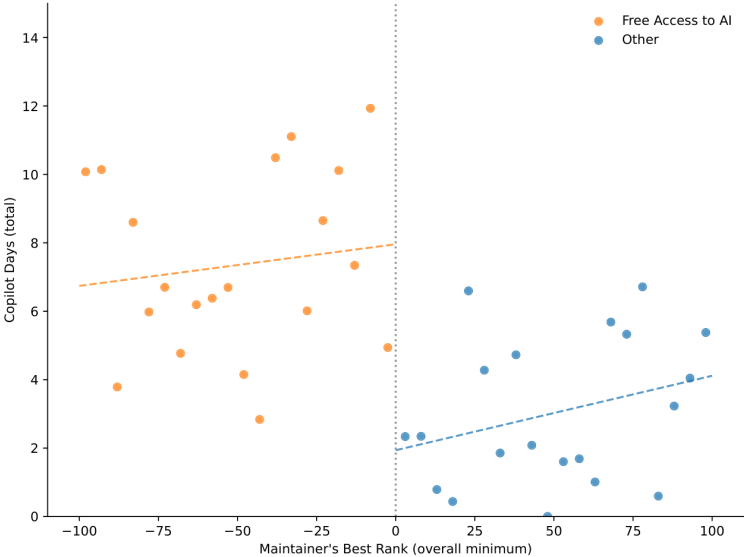
Copilot AI usage increases for free access rankings



Table

Dynamics

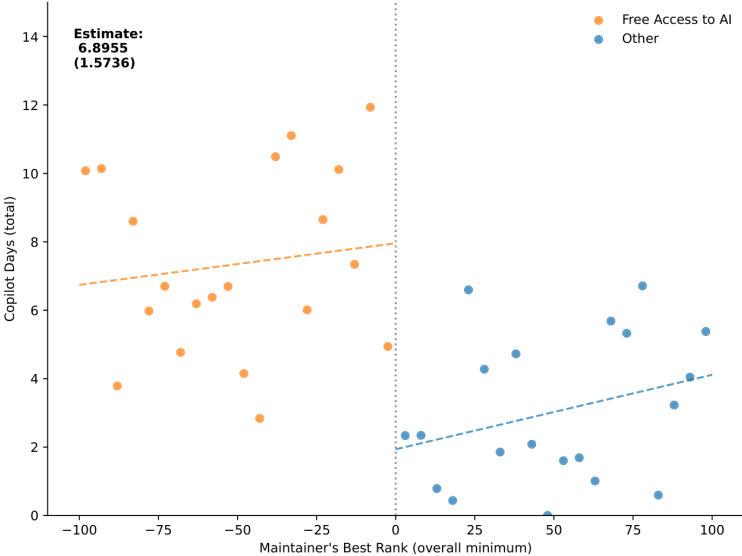
Copilot AI usage increases for free access rankings



Table

Dynamics

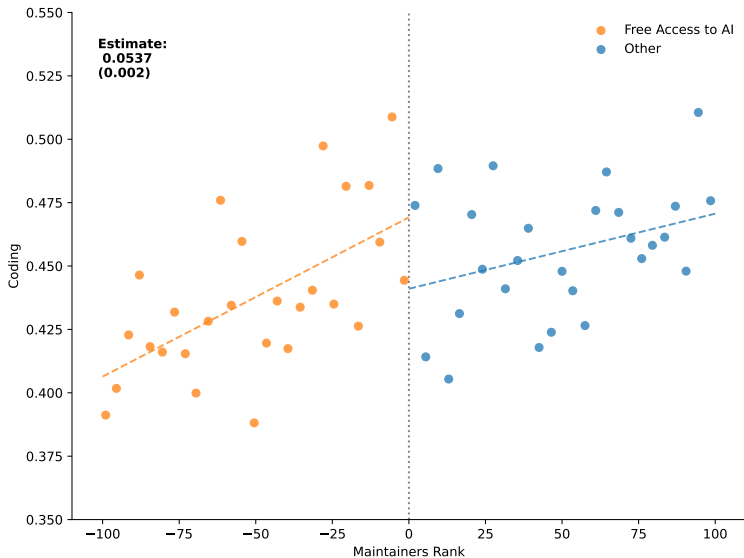
Copilot AI usage increases for free access rankings



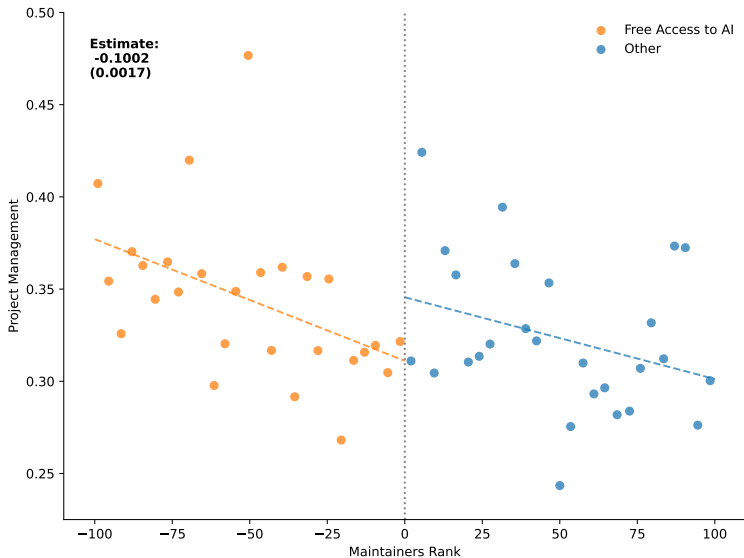
Table

Dynamics

Coding increases for free access rankings



Project management drops for free access rankings



Copilot treatment effects on the compliers

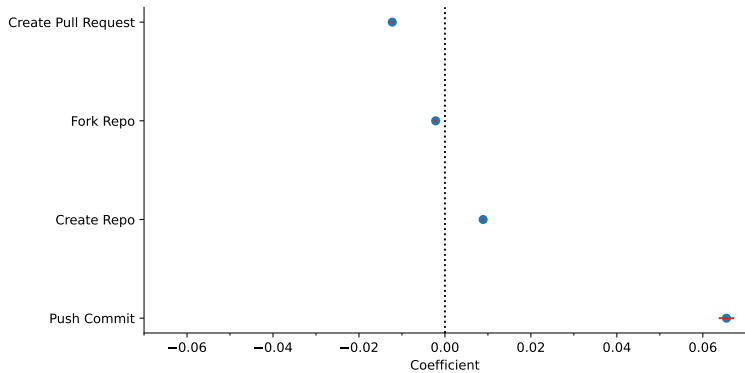
⇒ Ever-Uptake increases by 6.14 pp.

	Coding		Project Management	
	ITT	2SLS	ITT	2SLS
$\mathbb{1}(\text{Eligible}) / \text{Uptake}$	0.0537*** (0.002)	0.8904*** (0.048)	-0.1002*** (0.002)	-1.7311*** (0.082)
Baseline	0.4345*** (0.001)	0.2728*** (0.010)	0.3972*** (0.001)	0.7130*** (0.018)
N	269,546	269,546	269,546	269,546

⇒ Coding increases by 90 pp. for compliers

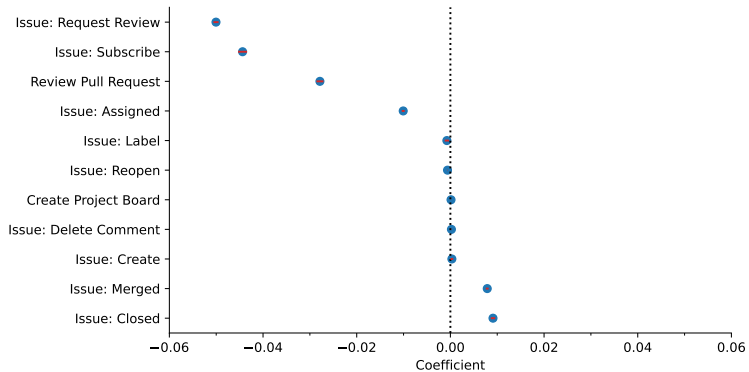
⇒ Project Management drops by 170 pp. for compliers

Individual level coding effects



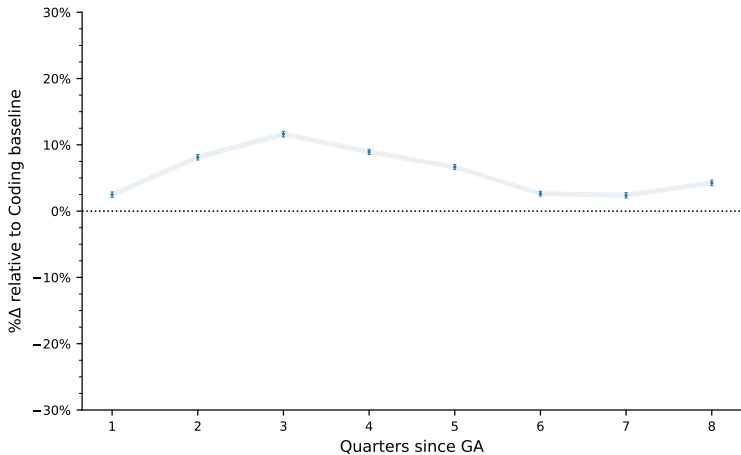
[Back](#)

Individual level project management effects

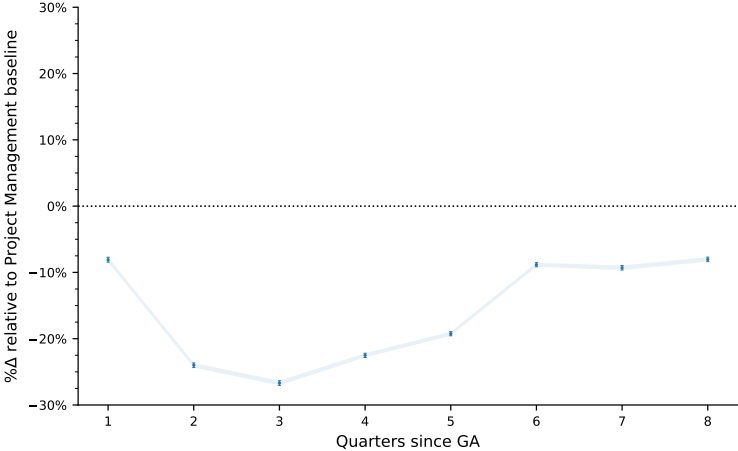


Back

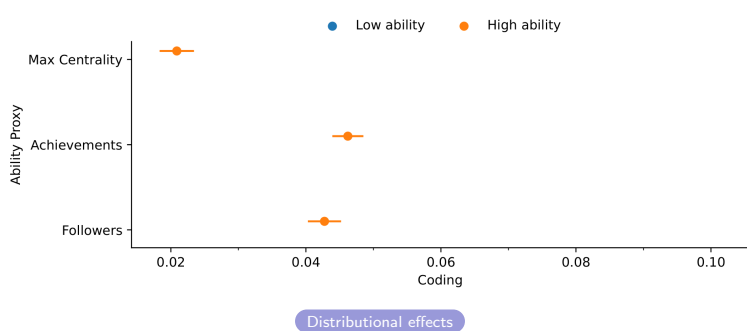
Dynamic effects of free-access AI on coding



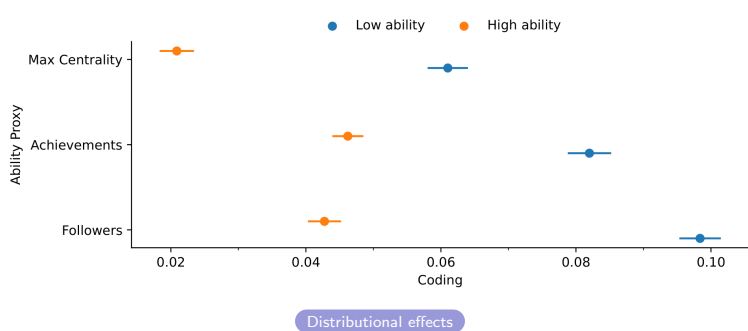
Dynamic effects of free-access AI on project management



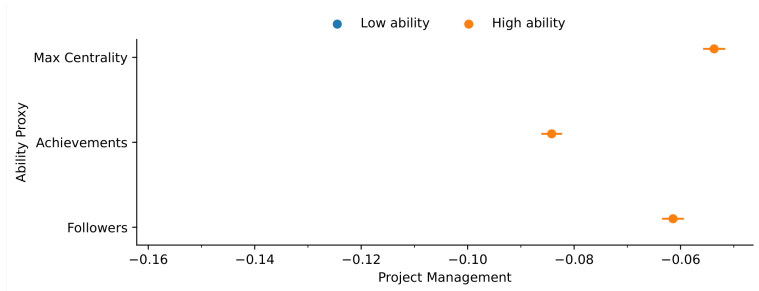
Low ability programmers code more



Low ability programmers code more

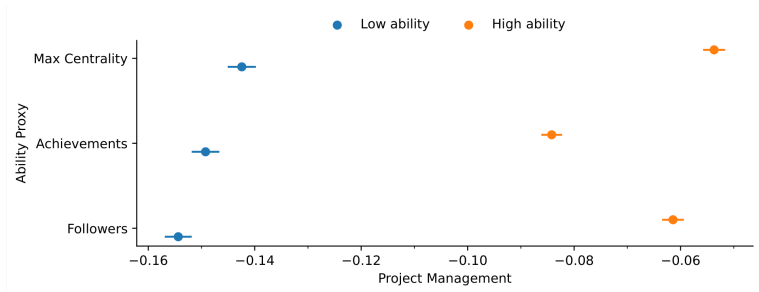


Low ability programmers reduce project management more



Distributional effects

Low ability programmers reduce project management more



Distributional effects

Robustness

Individual Level

- ⇒ No manipulation. No Knowledge of Ranking [Ycombinator](#) [GitHub](#)
- ⇒ No manipulation. Empirical Evidence [Histogram](#) [McCrary Test](#)
- ⇒ No other intervention. Smoothness of covariates [Figures](#)
- ⇒ Stability. Polynomial (Degree 1 & 2) [Polynomial: Table](#)
- ⇒ Stability. Kernel (Uniform, triangle) [Kernel: Table](#)
- ⇒ Stability. Bandwidth (MSE, CER) [Bandwidth: Table](#)

Project Level

- ⇒ Consistent with individual level [Project Level](#)
- ⇒ No manipulation. See above
- ⇒ No other intervention. Smoothness of covariates
- ⇒ Stability. Polynomial (Degree 1 & 2)
- ⇒ Stability. Kernel (Uniform, triangle)
- ⇒ Stability. Bandwidth (MSE, CER)

Other [Residual](#) [Back of the envelope](#)

Conclusion

- ⇒ We place emphasis on **nature of work** instead of productivity
- ⇒ We exploit a natural experiment in AI adoption
 - Causal interpretation
 - 2 years of AI use in real-world setting
- ⇒ Generative AI changes work processes by
 - ↑ coding, ↓ project management
 - Treatment effect remain after 2 years
 - Helping lower ability maintainers more
- ⇒ Generative AI can
 - positively impact the public good (e.g. open source software)
 - mitigate the linchpin problem in the distributed work context!

Thank you!

⇒ Manuel Hoffmann

⇒ mhoffmann@hbs.edu