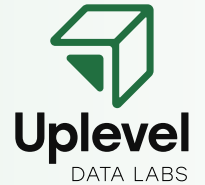


Can Generative AI Improve Developer Productivity?

HERE'S WHAT THE REAL-LIFE DATA SUGGESTS



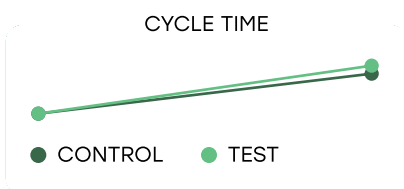
More engineering teams are experimenting with generative AI tools like Copilot to ostensibly improve productivity and developer experience. **But what does the quantitative data say?**

Using actual engineering data across a sample of nearly 800 developers in Uplevel's customer population, Uplevel Data Labs analyzed the difference in how teams with and without Copilot access performed according to objective metrics like cycle time, PR throughput, bug rate, and extended working hours ("Always On" time).

The expectation is that Copilot helps developers write code faster and smarter, which should lead to lower cycle time, more PRs, and fewer bugs without increasing the risk of burnout. Here's what we found:

Copilot access provided no significant change in efficiency metrics.

When comparing PR cycle time, throughput, and complexity along with PRs with tests, Copilot neither helped nor hurt the developers in the sample, and also did not increase coding speed.

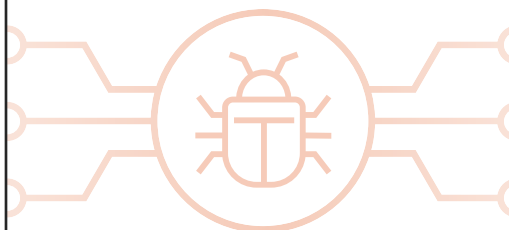


While some of these metrics were statistically significant, the actual change was inconsequential to engineering outcomes, e.g. cycle time decreased by 1.7 minutes.

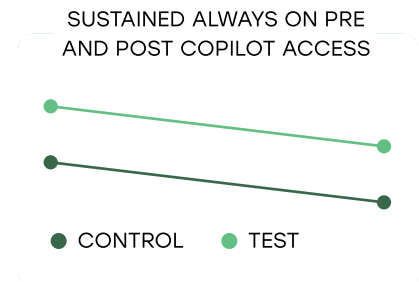
Developers with Copilot access saw a significantly higher bug rate while their issue throughput remained consistent.

+41%
IN BUG RATE

This suggests that Copilot access may impact code quality. (The fact that PR throughput was unchanged further supports this possibility.)



Copilot access was not effective in mitigating the risk of burnout.



Uplevel's "Sustained Always On" metric (extended working time outside of standard hours and a leading indicator of burnout), decreased for both groups. But it decreased by 17% for those with Copilot access and by almost 28% for those without.

What Does This Data Mean?

Access to generative AI tools like Copilot has raised a number of important questions.

Will AI help developers ship faster? Can it help them write better code and avoid burnout?

Not yet for this population. But innovation moves fast, and GitHub [reports](#) that Copilot does improve developer satisfaction. Engineering leaders may benefit from adopting a conservative Copilot adoption strategy to prepare for further advancements in the tool:



Set specific goals. What specifically are the outcomes that you are wanting to achieve by including Copilot in your team's workflow?



Offer training to your teams. Onboarding can be a good way to lay out where Copilot should and shouldn't be used and what safeguards are in place as an organization.



Continue to experiment with generative AI. Seek out the specific use cases in which Copilot can be helpful and the prompts that yield the best results. Share these findings across your organization so that success can be replicated.



Monitor the engineering effectiveness metrics that Copilot might impact. Start A/B testing on your own to gain objective, quantitative insight into whether AI is actually improving developer productivity and/or helping you reach your operational goals.

About the Study

Metrics were evaluated prior to implementation of Copilot from January 9 through April 9, 2023 versus after implementation from January 8 through April 7, 2024. This time period was selected to remove the effects of seasonality.

Data on Copilot access was provided to Uplevel Data Labs across several enterprise engineering customers for a total of 351 developers in the TEST group (with Copilot access) and 434 in the CONTROL group (without Copilot access). The developers in the CONTROL group were similar to those in the TEST group in terms of role, working days, and PR volume in each period.

The results are based on t-tests for numerical metrics and z-tests for proportions to understand any impacts to each metric. Analysis is based on whether individuals had access to Copilot, not actual usage, because Copilot does not make that data available at the individual level. All results are observational, limited to the developers included, and not causal.

Navigation for Engineering Leaders

Uplevel is the only holistic system of decision for enterprise engineering organizations. Applying advanced data science to tooling and collaboration data, Uplevel surfaces and interprets the hard-to-find signals that you need to focus your efforts, prioritize initiatives, and build an effective engineering culture.



[Learn more at
uplevelteam.com](https://uplevelteam.com)