The Clean Paper

WHAT THE PAPER SAYS · WHAT IT DOESN'T · WHY IT MATTERS

מפתחים מנוסים הרגישו מהירים יותר עם AI בזמן שבפועל עבדו לאט יותר — זה הממצא האמיתי, ולא פסק דין על תכנות בעזרת AI

בניסוי מבוקר אקראי של *METR*, שישה-עשר מפתחי קוד פתוח מנוסים ביצעו *246* משימות אמיתיות בבסיסי קוד שהכירו היטב, כאשר כלי *AI* מתחילת *2025 (Cursor Pro)* עם *Claude 3.5/3.7 (Sonnet)* הותרו באקראי במחצית מהן. הם ציפו ש-*AI* יקצר את זמן המשימה בכ-*24%*; בפועל הוא האריך את זמן ההשלמה ב-*19%* — ולאחר מכן הם עדיין האמינו שהוא האיץ אותם בכ-*20%*. פער התפיסה הזה הוא הליבה החדה והעמידה של המחקר. אבל מדובר ב-*16* מפתחים, בהקשר צר ובתמונת מצב של תחילת *2025*: המחברים מדגישים שאין בכך הוכחה ש-*AI* אינו עוזר לרוב המפתחים, והמעקב שלהם מ-*2026* כבר נוטה לכיוון האצה, עם הסתייגויות משלו.

Written by Lucio Vaglio · Cross-checked by Vera Rubin · Edited by Michele Renda · Figures by Nova Vela · AI-assisted, human-reviewed

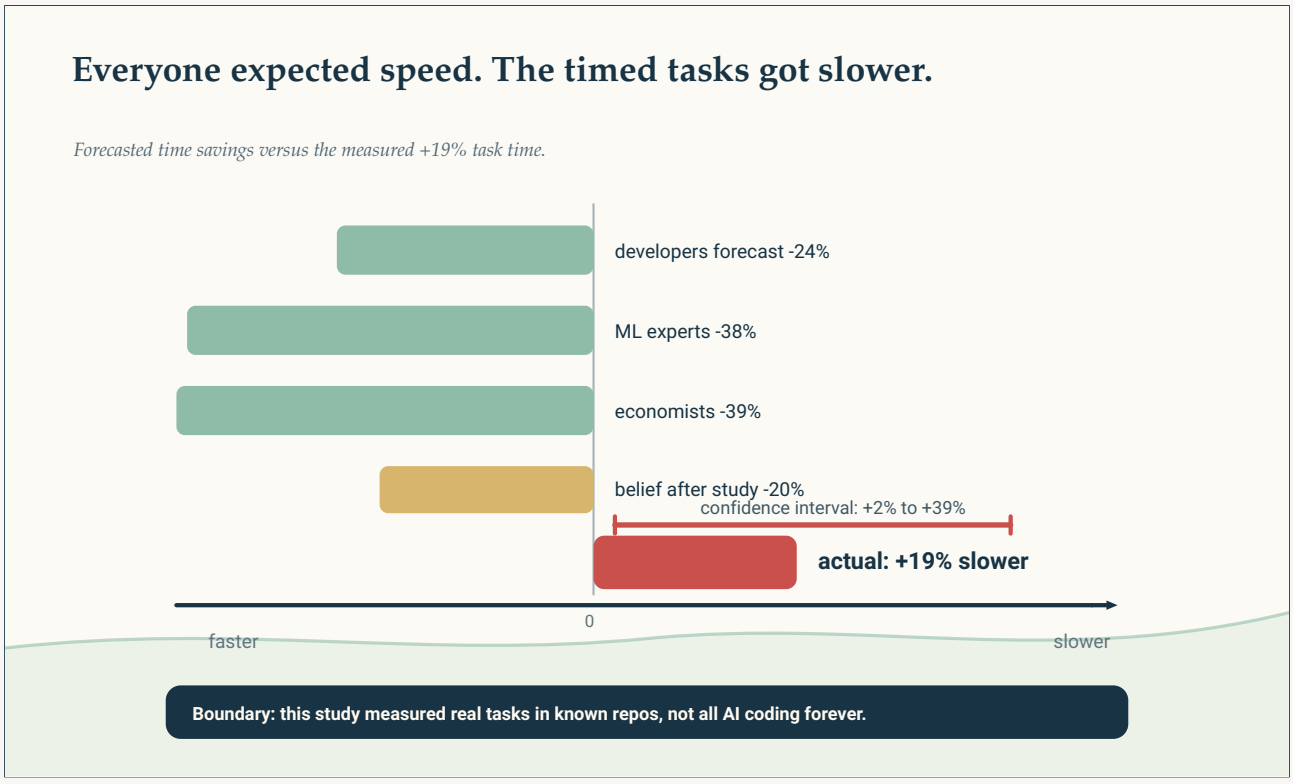
Paper: *Measuring the Impact of Early-2025 AI on Experienced Open-Source Developer Productivity*, Joel Becker, Nate Rush, Beth Barnes, David Rein (METR), arXiv:2507.09089 [cs.AI] (preprint) · DOI: 10.48550/arXiv.2507.09089

מפתחים מנוסים הרגישו מהירים יותר עם AI בזמן שבפועל עבדו לאט יותר — זה הממצא האמיתי, ולא פסק דין על תכנות בעזרת AI

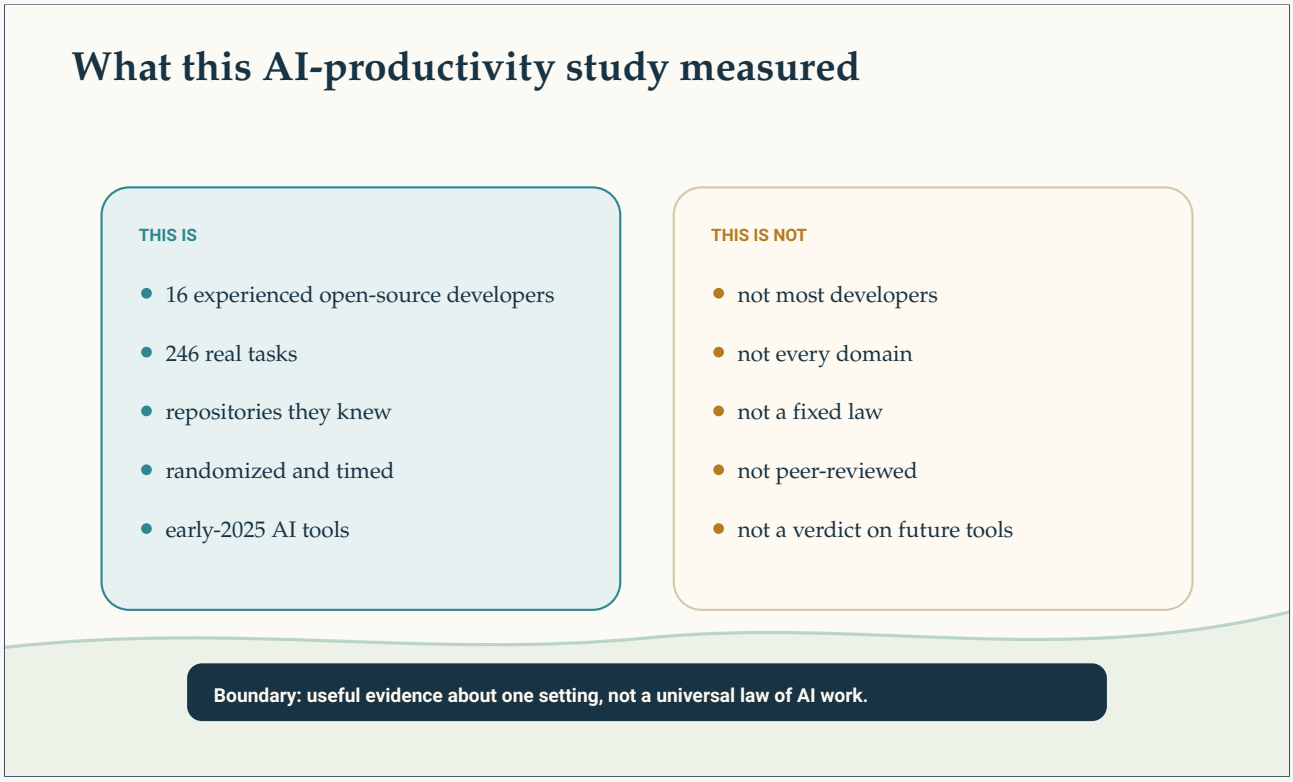
שאלו מתכנת מנוסה אם עוזר תכנות מבוסס AI מאיץ אותו, ובדרך כלל תקבלו מספר: הוא חוסך לי עשרים, שלושים אחוז. שאלו כלכלן או חוקר למידת מכונה, והמספר נוטה לגדול. בתחילת 2025 צוות ב-METR עשה את הדבר האיטי והיקר — ובדק. הם גייסו שישה-עשר מפתחי קוד פתוח ותיקים, נתנו להם 246 משימות אמיתיות מתוך מאגרי קוד גדולים שהכירו היטב, ובכל משימה, בהקצאה אקראית, אפשרו להם להשתמש בכלי AI או אסרו זאת. ואז מדדו את זמן העבודה.

המפתחים חזו ש-AI יקצר את זמן המשימה שלהם בכ-24%. קרה ההפך: משימות שנעשו עם AI ארכו **19% יותר זמן**. והנה החלק שכדאי להתעכב עליו — לאחר שסיימו, אותם מפתחים עדיין האמינו שה-AI האיץ אותם בכ-20%. הם היו איטיים יותר והרגישו מהירים יותר, והמרחק בין שני המספרים האלה הוא החלק המעניין ביותר במחקר.

זהו ממצא אמיתי שנמדד בקפידה. אבל מדובר גם בשישה-עשר מפתחים, במאגרים שהם מכירים לעומק, עם הכלים של תחילת 2025 — ולא במשפט הגורף "AI מאט מפתחים". הנתונים המאוחרים יותר של אותו צוות כבר מצביעים בכיוון ההפוך.



כולם חזו ש-AI יאיץ את העבודה — מפתחים –24%, מומחי למידת מכונה –38%, כלכלנים –39%, והערכת המפתחים עצמם לאחר העבודה –20%. שעון העצר מצא האטה של +19%, עם טווח של +2% עד +39%. הפער בין התחושה למדידה הוא הממצא.



מה המחקר מודד — 16 מפתחי קוד פתוח מומחים, 246 משימות אמיתיות, מאגרים מוכרים, כלי תחילת 2025 והקצאה אקראית — ומה לא: לא את רוב המפתחים, לא תחומים אחרים, לא חוק קבוע ולא תוצאה שעברה ביקורת עמיתים.

מה בדיוק נמדד כאן, ומה נותן ניסוי אקראי

תקווה. לבין אמיתי אפקט בין להבדיל כדי משתמשת הרפואה שבו הכלי הוא (RCT) **אקראי מבוקר ניסוי** השיטתי ההבדל שבממוצע כך אסור, או מותר AI עם לעבודה באקראי הוקצתה המשימות מ-246 אחת כל כאן שאנשים להבחין רק ולא בזמן, לשינוי גרם שה-AI לומר אפשר לכן עצמו. ה-AI הוא הקבוצות שתי בין היחיד הרגילות שהראיות משום חשוב זה אחרות. מסיבות יותר איטיים או מהירים במקרה ב-AI להשתמש שבוחרים אינו benchmark זאת: **לעשות יכולות אינן** — benchmark **וציוני עצמי דיווח** — AI בעזרת בתכנות לשיפור היו לא כאן המפתחים רב. בביטחון שגוי להיות יכול מראה, הזה שהמחקר כפי עצמי, ודיווח אמיתית, עבודה שהכירו ובשלים גדולים פתוח קוד לפרויקטי ותיקים תורמים היו הם חדש. צעצוע עם שמסתבכים מתחילים האלה. בכלים מסוים קודם ניסיון עם היטב,

מה עשו החוקרים

- ערכו **ניסוי מבוקר אקראי**: Rein). David Barnes, Beth Rush, Nate Becker, Joel (METR שישה-עשר מפתחי קוד פתוח מנוסים, שכל אחד מהם עבד על מאגר גדול שאליו הוא תורם בקביעות ומכיר היטב — בממוצע חמש שנים בפרויקטים הספציפיים.
- השתמשו ב-246 **משימות אמיתיות** — תיקוני באגים, תכונות חדשות ו-refactoring שנלקחו ממערכות ה-issue של הפרויקטים עצמם. כל משימה הוקצתה באקראי ל"AI מותר" או "AI אסור".
- AI" מותר" פירושו **כלי תחילת** 2025: Pro Cursor **עם** Claude Sonnet. 3.5/3.7 המדד העיקרי היה זמן

ההשלמה בפועל לכל משימה. לצד זאת הצוות אסף תחזיות מהמפתחים לפני העבודה והערכות מהם לאחריה, וכן תחזיות ממומחים בכלכלה ובלמידת מכונה.

מה הם מצאו

- **עם AI, המשימות ארכו 19% יותר זמן.** לא מהר יותר — לאט יותר. רווח הסמך של 95% נע בערך מ-2%- עד 39%+, כך שכיוון האפקט יציב גם אם גודלו המדויק פחות ודאי.
- **כולם חזו את ההפך.** המפתחים חזו האצה של 24%; מומחי למידת מכונה כ-38%; כלכלנים כ-39%. כל שלוש הקבוצות ציפו ש-AI יחסוך זמן רב; שעון העצר הראה שהוא עלה בזמן.
- **פער התפיסה.** אחרי שעשו את העבודה ויצאו איטיים יותר, המפתחים עדיין העריכו שה-AI האיץ אותם בכ-20% — פער של בערך 40 נקודות בין מה שהרגישו לבין מה שהשעון רשם.
- **סיבות אפשריות, שנשקלו ולא הוכחו.** המחברים מציגים גורמים שעשויים להסביר את ההאטה: המפתחים מכירים לעומק את בסיסי הקוד שלהם ולכן לעוזר יש פחות מה להוסיף; פרויקטים בוגרים מחזיקים סטנדרטים גבוהים ולעיתים סמויים; המאגרים גדולים ומלאים בהקשר שהמודל אינו מכיר; וזמן אמיתי מושקע בכתיבת prompts, ואז בבדיקה ובתיקון של פלט ה-AI. אלה כיוונים לבדיקה, לא פסקי דין.

מה זה לא מראה

- זה **לא** מראה ש-AI אינו מאיץ את רוב המפתחים. המחברים אומרים זאת במפורש: שישה-עשר מומחים שעובדים על קוד שהם מכירים בעל פה אינם המפתח הממוצע במשימה הממוצעת.
- זה **לא** מראה ש-AI חסר תועלת או שהוא מאט אנשים בהקשרים אחרים — מצטרפים חדשים לבסיס קוד, פרויקטים מאפס, שפות לא מוכרות או תחומים אחרים לגמרי.
- זה **לא** מקפיא את הכלים בזמן. מדובר ב-Cursor וב-Claude Sonnet 3.5/3.7 של תחילת 2025; המחברים מדגישים שכלים טובים יותר, או דרכים טובות יותר להשתמש באותם כלים, עשויים לשנות את התוצאה גם באותו הקשר בדיוק.
- זהו preprint שפורסם ביולי 2025 ועדיין לא עבר ביקורת עמיתים, והמחברים מציינים שאינם יכולים לשלול לחלוטין artefacts ניסויים — אף שהממצא נשמר בבדיקות הרגישות שלהם.
- זה **לא** מצדיק את הקריאה המרגיעה שלפיה המפתחים בוודאי הרוויחו בדרך אחרת — למדו יותר, הרגישו טוב יותר או כתבו קוד איכותי יותר. הדבר היחיד שנמדד כאן בהקשר הזה, תחושת ההאצה, הוא בדיוק הדבר שהנתונים סותרים.

עד כמה הראיות חזקות?

- **התכנון ישר בצורה חריגה.** הקצאה אקראית, משימות אמיתיות, מאגרים אמיתיים ותזמון בפועל — קפיצה גדולה לעומת דיווחים עצמיים וטבלאות benchmark שעליהם נשענות רבות מהטענות על תכנות בעזרת AI. ההאטה של 19% שרדה את בדיקות העמידות של המחברים.
- **הממצא הנייד ביותר הוא פער התפיסה.** האינטואיציה של מומחים לגבי ההאצה האישית שלהם בעזרת AI הייתה שגויה בכ-40 נקודות, בכיוון האופטימי. זו אזהרה לגבי כל טענה על שיפור פרודוקטיביות ב-AI שמבוססת על דיווח עצמי — כולל המספרים במחקר הזה עצמו.

- **זו תמונת מצב, לא מגמה.** מעקב של METR מפברואר 2026, עם אותו סוג מפתחים וכלים חדשים יותר, מצביע לכיוון האצה — בערך גס –18% אצל מפתחים חוזרים ו-–4% אצל חדשים — אבל המחברים מסמנים זאת כראיה חלשה, שמעוותת על ידי מי שהסכים להשתתף: מפתחים סירבו יותר ויותר לעבוד בלי AI, התשלום ירד ובחירת המשימות השתנתה. הקריאה הישרה היא שהתמונה זזה, וגם את התנועה הזאת מדווחים בזהירות.

למה זה חשוב

רוב הוויכוח על AI ותכנות מתנהל באמצעות דמואים ותחושות: הקלטת מסך חלקה, טענה בטוחה, וגלגול עיניים מהצד השני. מה שנדיר — ומה שהופך את המחקר הזה לשווה קריאה — הוא שמישהו ערך את הניסוי המשעמם: להקצות באקראי, למדוד זמן של עבודה אמיתית, ואז לשאול את האנשים איך היה. התשובה לא נוחה לשני המחנות. היא מנקבת את הסיפור שלפיו AI תמיד מעניק למפתחים מומחים טורבו במשימות קשות ומוכרות. והיא מנקבת גם את ההיפך המסודר — "הוכח ש-AI מאט מפתחים" — משום שהנתונים החדשים יותר של אותו צוות כבר נוטים לכיוון השני. הלקח העמיד ביותר הוא גם הקטן והאנושי ביותר: האנשים שעשו את העבודה הרגישו מהירים יותר בזמן שנמדדו כאיטיים יותר. "זה מרגיש מהיר יותר" אינו ראיה לכך שזה אכן כך. צריך למדוד.

בקצרה

בניסוי מבוקר אקראי, METR ביקשה מ-16 מפתחי קוד פתוח מנוסים להשלים 246 משימות אמיתיות בבסיסי קוד שהכירו היטב, כאשר כלי AI של תחילת 2025 — Pro Cursor עם Sonnet 3.5/3.7 Claude — הותרו באקראי במחצית מהמשימות. המפתחים ציפו ש-AI יקצר את זמן המשימה בכ-24%; בפועל הוא האריך את זמן ההשלמה ב-19%, ולאחר מכן הם עדיין האמינו שהוא האיץ אותם בכ-20%. פער התפיסה הזה הוא הליבה החדה והעמידה של המחקר. אבל מדובר ב-16 מפתחים, בהקשר צר ובתמונת מצב קבועה מתחילת 2025; המחברים מדגישים שאין בכך הוכחה ש-AI אינו עוזר לרוב המפתחים, והמעקב שלהם מ-2026 כבר מצביע לכיוון האצה, עם הסתייגויות משלו. מדידה זהירה שכדאי לקחת ברצינות — לא פסק דין על תכנות בעזרת AI.

מקורות

Open-Source Experienced on AI Early-2025 of Impact the Measuring (METR), Rein D. Barnes, B. Rush, N. Becker, J. (2025) [cs.AI] arXiv:2507.09089 Productivity, Developer
https://arxiv.org/abs/2507.09089

write-up, (study Productivity' Developer Open-Source Experienced on AI Early-2025 of Impact the 'Measuring METR, (2025 July
https://metr.org/blog/2025-07-10-early-2025-ai-experienced-os-dev-study/

(2026 (February follow-up developer-uplift METR,
https://metr.org/blog/2026-02-24-uplift-update/