გამოცდილ დეველოპერებს AI-სთან მუშაობისას თავი უფრო სწრაფად ეჩვენებოდათ, თუმცა გაზომვით უფრო ნელა მუშაობდნენ — ეს არის ნამდვილი შედეგი და არა განაჩენი AI-კოდირებაზე

METR-ის რანდომიზებულ კონტროლირებად კვლევაში 16 გამოცდილმა ღია კოდის დეველოპერმა 246 რეალური დავალება შეასრულა მათთვის კარგად ნაცნობ კოდურ ბაზებში; დავალებების შემთხვევით ნახევარში 2025 წლის დასაწყისის AI-ინსტრუმენტები (Cursor Pro და Claude 3.5/3.7 Sonnet) ნებადართული იყო. ისინი ელოდნენ, რომ AI დროს დაახლოებით 24%-ით შეამცირებდა, მაგრამ რეალურად შესრულების დრო 19%-ით გაიზარდა — და შემდეგაც კი მონაწილეებს ეგონათ, რომ დაახლოებით 20%-ით დაჩქარდნენ. სწორედ აღქმასა და გაზომვას შორის სხვაობაა კვლევის გამძლე ბირთვი. თუმცა ეს მხოლოდ 16 დეველოპერია, ერთი ვიწრო გარემო და 2025 წლის დასაწყისის კონკრეტული ინსტრუმენტები: ავტორები პირდაპირ ამბობენ, რომ შედეგი არ ნიშნავს, თითქოს AI დეველოპერების უმრავლესობას ვერ ეხმარება, ხოლო მათი 2026 წლის შემდგომი მონაცემები უკვე აჩქარებისკენ იხრება — თავისი შეზღუდვებით.

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

გამოცდილ დეველოპერებს AI-სთან მუშაობისას თავი უფრო სწრაფად ეჩვენებოდათ, თუმცა გაზომვით უფრო ნელა მუშაობდნენ — ეს არის ნამდვილი შედეგი და არა განაჩენი AI-კოდირებაზე

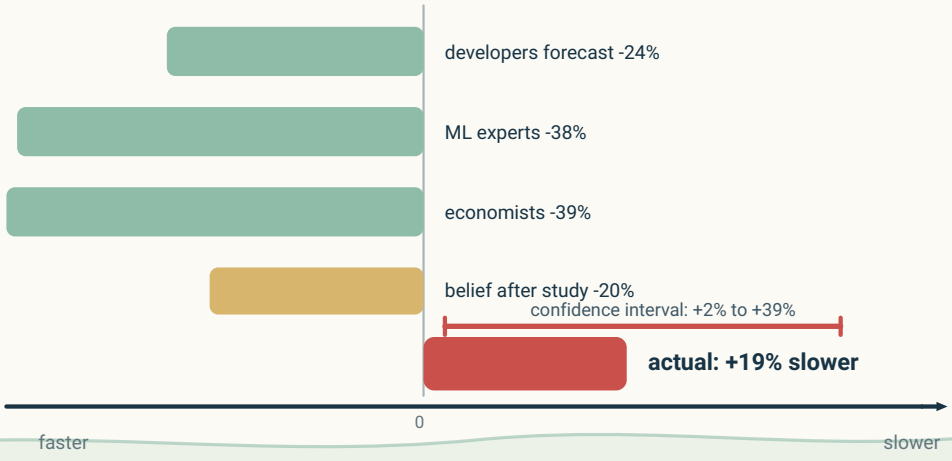
გამოცდილ პროგრამისტს რომ ჰკითხოთ, აჩქარებს თუ არა AI-კოდირების ასისტენტი მის მუშაობას, ხშირად კონკრეტულ რიცხვს მიიღებთ: ოცი პროცენტი, ოცდაათი პროცენტი. ეკონომისტს ან მანქანური სწავლების მკვლევარს თუ ჰკითხავთ, შეფასება კიდევ უფრო მაღალი იქნება. 2025 წლის დასაწყისში METR-ის გუნდმა უფრო ნელი და ძვირი გზა აირჩია — პირდაპირ გაზომა. მათ 16 გამოცდილ ღია კოდის დეველოპერს მისცეს 246 რეალური დავალება დიდი კოდური ბაზებიდან, რომლებსაც ეს ადამიანები კარგად იცნობდნენ; თითოეული დავალებისთვის შემთხვევითი არჩევანით ან აძლევდნენ AI-ინსტრუმენტების გამოყენების უფლებას, ან არა. შემდეგ კი სამუშაოს შესრულების დროს ზომავდნენ.

დეველოპერები წინასწარ ვარაუდობდნენ, რომ AI დავალების დროს დაახლოებით 24%-ით შეამცირებდა. მოხდა საპირისპირო: AI-ს გამოყენებით შესრულებულ დავალებებს **19%-ით მეტი დრო** დასჭირდა. კიდევ უფრო საინტერესოა შემდეგი ნაწილი — სამუშაოს დასრულების შემდეგ იგივე დეველოპერებს კვლავ სჯეროდათ, რომ AI-მ ისინი დაახლოებით 20%-ით დააჩქარა. რეალურად უფრო ნელა მუშაობდნენ, მაგრამ თავი უფრო სწრაფად ეჩვენებოდათ; ამ ორ რიცხვს შორის სხვაობა კვლევის ყველაზე საინტერესო შედეგია.

ეს რეალური და ფრთხილად გაზომილი შედეგია. მაგრამ ის ეხება მხოლოდ 16 დეველოპერს, იმ რეპოზიტორიებში, რომლებსაც ისინი ძალიან კარგად იცნობდნენ, და 2025 წლის დასაწყისის ინსტრუმენტებით მუშაობას. ამიტომ შედეგი არ ნიშნავს მარტივ ფრაზას: „AI დეველოპერებს ანელებს“. იმავე გუნდის უფრო გვიანი მონაცემები უკვე საპირისპირო მიმართულებას მიუთითებს.

Everyone expected speed. The timed tasks got slower.

Forecasted time savings versus the measured +19% task time.



Boundary: this study measured real tasks in known repos, not all AI coding forever.

ყველა ჯგუფი ვარაუდობდა, რომ AI მუშაობას დააჩქარებდა — დეველოპერები −24%-ს, მანქანური სწავლების ექსპერტები −38%-ს, ეკონომისტები −39%-ს, ხოლო თვით დეველოპერებს სამუშაოს შემდეგაც დაახლოებით −20%-იანი აჩქარება ეგონათ. ქრონომეტრმა კი +19%-იანი შენელება აჩვენა, +2%-დან +39%-მდე ინტერვალით. სწორედ განცდილ და გაზომილ შედეგს შორის სხვაობაა მთავარი აღმოჩენა.

Original diagram — The Clean Paper CC BY 4.0

What this AI-productivity study measured

THIS IS

- 16 experienced open-source developers
- 246 real tasks
- repositories they knew
- randomized and timed
- early-2025 AI tools

THIS IS NOT

- not most developers
- not every domain
- not a fixed law
- not peer-reviewed
- not a verdict on future tools

Boundary: useful evidence about one setting, not a universal law of AI work.

რას ზომავს კვლევა: 16 გამოცდილი ღია კოდის დეველოპერი, 246 რეალური დავალება, მათთვის ნაცნობი რეპოზიტორიები, 2025 წლის დასაწყისის ინსტრუმენტები და რანდომიზებული განაწილება. რას არ ზომავს: დეველოპერების უმრავლესობას, სხვა სფეროებს, უცვლელ კანონზომიერებას ან უკვე რეცენზირებულ შედეგს.

Original diagram — The Clean Paper CC BY 4.0

რას ზომავდა ეს კვლევა და რას გვაძლევს რანდომიზაცია

რანდომიზებული კონტროლირებადი კვლევა (RCT) მედიცინაში გამოიყენება იმის გასარჩევად, რეალურ ეფექტთან გვაქვს საქმე თუ მხოლოდ იმედისმომცემ დამთხვევასთან. აქ 246-ვე დავალება შემთხვევითად ნაწილდებოდა ორ რეჟიმს შორის — AI-ს გამოყენება ნებადართული იყო ან აკრძალული. ამიტომ საშუალოდ ამ ორ ჯგუფს შორის ერთადერთი სისტემატური განსხვავება თავად AI უნდა ყოფილიყო. სწორედ ეს გვაძლევს უფლებას ვთქვათ, რომ დროის ცვლილება AI-მ *გამოიწვია*, და არა უბრალოდ ის დავინახეთ, რომ ადამიანები, რომლებიც AI-ს მიმართავენ, სხვა მიზეზებითაც უფრო სწრაფები ან ნელები არიან. ეს მნიშვნელოვანია, რადგან AI-კოდირების სარგებლის შესახებ ჩვეულებრივი მტკიცებულებები — თვითშეფასება და ბენჩმარკის ქულები — ამას ვერ გვაძლევს: ბენჩმარკი რეალური სამუშაო არ არის, ხოლო თვითშეფასება, როგორც ეს კვლევა აჩვენებს, შეიძლება დიდი თავდაჯერებითაც მცდარი იყოს. მონაწილეები ახალბედები არ იყვნენ, რომლებიც უცნობ ინსტრუმენტს პირველად სინჯავდნენ. ისინი დიდი, მომწიფებული ღია კოდის პროექტების დამკვიდრებული კონტრიბუტორები იყვნენ, კარგად იცნობდნენ ამ პროექტებს და AI-ინსტრუმენტების გამოყენების გარკვეული გამოცდილებაც ჰქონდათ.

რა გააკეთეს ავტორებმა

- ჩაატარეს **რანდომიზებული კონტროლირებადი კვლევა** (METR: Joel Becker, Nate Rush, Beth Barnes, David Rein). მონაწილეობდა 16 გამოცდილი ღია კოდის დეველოპერი; თითოეული მუშაობდა დიდ რეპოზიტორიაზე, რომელშიც რეგულარულად შეაქვს წვლილი და რომელსაც კარგად იცნობს — კონკრეტულ პროექტებზე მათი საშუალო გამოცდილება დაახლოებით ხუთი წელი იყო.
- გამოიყენეს **246 რეალური დავალება** — შეცდომების გასწორება, ახალი ფუნქციების დამატება და რეფაქტორინგი, აღებული თავად პროექტების issue tracker-ებიდან. თითოეული დავალება შემთხვევითად ხვდებოდა „AI ნებადართულია“ ან „AI აკრძალულია“ ჯგუფში.
- „AI ნებადართულია“ ნიშნავდა **2025 წლის დასაწყისის ინსტრუმენტებს: Cursor Pro-ს Claude 3.5/3.7 Sonnet-თან ერთად.** მთავარი საზომი იყო თითოეული დავალების რეალური დასრულების დრო. ამასთან ერთად გუნდმა შეაგროვა დეველოპერების წინასწარი პროგნოზები და სამუშაოს შემდეგ გაკეთებული შეფასებები, ასევე ეკონომიკისა და მანქანური სწავლების ექსპერტების პროგნოზები.

რა აღმოაჩინეს

- **AI-ს გამოყენებით დავალებებს 19%-ით მეტი დრო დასჭირდა.** არა ნაკლები — მეტი. 95%-იანი ნდობის ინტერვალი დაახლოებით +2%-დან +39%-მდეა, ამიტომ ეფექტის მიმართულება საკმაოდ საიმედოა, თუნდაც ზუსტი სიდიდე ნაკლებად განსაზღვრული იყოს.
- **ყველა საპირისპიროს ელოდა.** დეველოპერები 24%-იან აჩქარებას პროგნოზირებდნენ, მანქანური სწავლების ექსპერტები — დაახლოებით 38%-ს, ეკონომისტები — დაახლოებით 39%-ს. სამივე ჯგუფი დიდ დროის ეკონომიას ელოდა; გაზომვამ კი დროის ზრდა აჩვენა.
- **აღქმასა და გაზომვას შორის სხვაობა.** მიუხედავად იმისა, რომ სამუშაო უფრო დიდხანს გაგრძელდა, დეველოპერები მაინც ფიქრობდნენ, რომ AI-მ ისინი დაახლოებით 20%-ით დააჩქარა — განცდასა და ქრონომეტრის შედეგს შორის დაახლოებით 40 პროცენტული პუნქტის სხვაობა.
- **შესაძლო მიზეზები შეფასებულია, მაგრამ დამტკიცებული არაა.** ავტორები ასახელებენ რამდენიმე ფაქტორს, რომლებმაც შეიძლება შენელება ახსნას: მონაწილეები საკუთარ კოდურ ბაზებს ძალიან კარგად იცნობდნენ, ამიტომ ასისტენტს ნაკლები დამატებითი ღირებულება ჰქონდა; მომწიფებულ პროექტებში მაღალი და ხშირად ჩუმად ნაგულისხმევი ხარისხის სტანდარტებია; რეპოზიტორიები დიდია და შეიცავს კონტექსტს, რომელიც მოდელს არ აქვს; დრო იხარჯება მოთხოვნების ფორმულირებაზე, შემდეგ კი AI-ს შედეგის შემოწმებასა და გასწორებაზე. ეს მიზეზები განხილვის კანდიდატებია და არა საბოლოო დასკვნები.

რას არ აჩვენებს ეს კვლევა

- ის **არ** აჩვენებს, რომ AI დეველოპერების უმრავლესობას ვერ აჩქარებს. ავტორები ამას პირდაპირ ამბობენ: 16 ექსპერტი, რომლებიც საკუთარი კოდის ბაზებს ზედმიწევნით იცნობენ, საშუალო დეველოპერსა და საშუალო დავალებას არ წარმოადგენს.
- ის **არ** აჩვენებს, რომ AI უსარგებლოა ან სხვა გარემოებებშიც ანელებს ადამიანებს — მაგალითად, ახალი თანამშრომლებისთვის უცნობ კოდურ ბაზაში, ნულიდან დაწყებულ პროექტში, უცნობ პროგრამირების ენაზე ან საერთოდ სხვა სფეროში.
- ის **არ** ნიშნავს, რომ ინსტრუმენტების ეფექტი უცვლელია. კვლევა ეხება 2025 წლის დასაწყისის Cursor-სა და Claude 3.5/3.7 Sonnet-ს; ავტორები ხაზს უსვამენ, რომ უკეთესმა ინსტრუმენტებმა ან მათი გამოყენების უკეთესმა გზებმა შედეგი იმავე გარემოშიც შეიძლება შეცვალოს.
- ეს არის **პრეპრინტი** (გამოქვეყნდა 2025 წლის ივლისში და ჯერ არ გაუვლია რეცენზირება), ხოლო ავტორები აღნიშნავენ, რომ ექსპერიმენტული არტეფაქტების სრულად გამორიცხვა შეუძლებელია — თუმცა შედეგი მათ სხვადასხვა ანალიზშიც შენარჩუნდა.
- კვლევა ასევე **არ** გვაძლევს საფუძველს ვიფიქროთ, რომ დეველოპერებმა აუცილებლად სხვა რამ მოიგეს — მეტი ისწავლეს, უკეთ იგრძნეს თავი ან უკეთესი კოდი დაწერეს.

ერთადერთი ასეთი რამ, რაც აქ გაიზომა — სისწრაფის განცდა — სწორედ ისაა, რასაც რეალური დრო ეწინააღმდეგება.

რამდენად ძლიერია მტკიცებულება

- **კვლევის დიზაინი უჩვეულოდ პირდაპირია.** რანდომიზაცია, რეალური დავალებები, რეალური რეპოზიტორიები და ფაქტობრივი დროის გაზომვა — ეს ბევრად ძლიერი საფუძველია, ვიდრე თვითშეფასებები და ბენჩმარკის ლიდერბორდები, რომლებზეც AI-კოდირების მრავალი პრეტენზიაა დაფუძნებული. 19%-იანი შენელება ავტორების სიმტკიცის შემოწმებებშიც შენარჩუნდა.
- **ყველაზე ფართოდ გადასატანი შედეგი აღქმის სხვაობაა.** ექსპერტების ინტუიცია საკუთარი AI-სარგებლის შესახებ დაახლოებით 40 პროცენტული პუნქტით აცდა რეალობას — ოპტიმისტური მიმართულებით. ეს გაფრთხილებაა AI-ის პროდუქტიულობის *ნებისმიერი* თვითრეპორტირებული ზრდის მიმართ, მათ შორის ამ კვლევის მონაწილეთა შეფასებების მიმართაც.
- **ეს ერთი დროის მონაკვეთია და არა ტენდენცია.** METR-ის 2026 წლის თებერვლის შემდგომმა კვლევამ, მსგავს დეველოპერებთან და უფრო ახალ ინსტრუმენტებზე, უკვე აჩქარებისკენ მიუთითა — დაბრუნებული მონაწილეებისთვის დაახლოებით −18%, ახალი მონაწილეებისთვის კი −4%. თუმცა ავტორები ამას სუსტ მტკიცებულებად აფასებენ, რადგან შედეგზე გავლენას ახდენდა ის, ვინ თანხმდებოდა მონაწილეობაზე: დეველოპერები უფრო ხშირად უარს ამბობდნენ AI-ს გარეშე მუშაობაზე, ანაზღაურება შემცირდა და დავალებების შერჩევაც შეიცვალა. ყველაზე ფრთხილი კითხვა ასეთია: სურათი იცვლება, მაგრამ ცვლილებაც კი დიდი სიფრთხილით უნდა წავიკითხოთ.

რატომ არის ეს მნიშვნელოვანი

AI-სა და პროგრამირებაზე დისკუსიის დიდი ნაწილი დემოებზე და შთაბეჭდილებებზე დგას: გლუვი ეკრანის ჩანაწერი, თავდაჯერებული განცხადება და საპასუხო ირონიული რეაქცია. იშვიათია მოსაწყენი, მაგრამ საჭირო ექსპერიმენტი — დავალებების შემთხვევითად განაწილება, რეალური სამუშაოს დროის გაზომვა და შემდეგ მონაწილეებისთვის კითხვა, როგორ ჩაიარა. პასუხი ორივე ბანაკისთვის მოუხერხებელია. ის აზიანებს ისტორიას, თითქოს AI ყველა გამოცდილ დეველოპერს ავტომატურად აჩქარებს რთულ და ნაცნობ კოდზე. მაგრამ ასევე აზიანებს საპირისპირო, ზედმეტად მარტივ დასკვნას — „დამტკიცებულია, რომ AI დეველოპერებს ანელებს“ — რადგან იმავე გუნდის უფრო ახალი მონაცემები უკვე სხვა მიმართულებას აჩვენებს. ყველაზე გამძლე გაკვეთილი ყველაზე პატარა და ადამიანურიცაა: ადამიანები თავს უფრო სწრაფად გრძნობდნენ მაშინ, როცა გაზომვით უფრო ნელა მუშაობდნენ. „უფრო სწრაფად მეჩვენება“ მტკიცებულება არ არის. უნდა გაზომო.

მოკლე შეჯამება

რანდომიზებულ კონტროლირებად კვლევაში METR-მა 16 გამოცდილ ღია კოდის დეველოპერს 246 რეალური დავალება შეასრულებინა მათთვის კარგად ნაცნობ კოდურ ბაზებში; დავალებების შემთხვევით ნახევარში 2025 წლის დასაწყისის AI-ინსტრუმენტები — Cursor Pro და Claude 3.5/3.7 Sonnet — ნებადართული იყო. დეველოპერები ელოდნენ, რომ AI დავალების დროს დაახლოებით 24%-ით შეამცირებდა, მაგრამ რეალურად დასრულების დრო 19%-ით გაიზარდა — და ამის შემდეგაც კი მონაწილეებს ეგონათ, რომ დაახლოებით 20%-ით დაჩქარდნენ. სწორედ აღქმასა და გაზომვას შორის სხვაობაა კვლევის მკაფიო და გამძლე ბირთვი. თუმცა ეს მხოლოდ 16 დეველოპერია, ერთი ვიწრო გარემო და 2025 წლის დასაწყისის კონკრეტული ინსტრუმენტები; ავტორები პირდაპირ ამბობენ, რომ შედეგი არ ნიშნავს, თითქოს AI დეველოპერების უმრავლესობას ვერ ეხმარება, ხოლო მათი 2026 წლის შემდგომი მონაცემები უკვე აჩქარებისკენ იხრება — თავისი შეზღუდვებით. სერიოზულად მისაღები ფრთხილი გაზომვაა, არა განაჩენი AI-კოდირებაზე.

წყაროები

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

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

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