

The Reliability Gap: Repeated-Trial Consistency and the Deployable Autonomy Horizon of Frontier AI Agents

Alfredo Merlet

alfredo.merlet@gmail.com

Instituto Profesional IACC <https://orcid.org/0009-0007-8558-164X>

Research Article

Keywords: AI agents, time horizon, pass^k reliability, capability evaluation, intraclass correlation, AI governance

Posted Date: October 22nd, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11447587/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: The authors declare no competing interests.

Abstract

Capability evaluations of AI agents increasingly summarise progress as a time horizon: the length of task, measured in human expert time, that an agent completes with 50% probability. Deployment decisions, however, depend on whether an agent succeeds every time a task recurs, not on whether it succeeds on average. This article estimates reliability-adjusted time horizons for frontier AI agents and asks how far they lag behind capability horizons. Using the public run-level records of the METR time-horizon evaluations (60,417 attempts by 30 and 20 models released between 2023 and 2026, on 170 and 228 tasks in two benchmark versions), it combines the unbiased pass^k estimator with per-model fractional logistic curves on log task duration, and quantifies uncertainty with a task-family cluster bootstrap (1,000 replicates). Four hypotheses were specified before estimation. For every model, the horizon at which all four attempts succeed with 50% probability is shorter than the capability horizon (geometric-mean ratio 2.39, 95% CI 2.07–2.84, in the newer benchmark; 2.33, 1.98–2.81, in the older one). The gap narrows by about 17% per year in point estimate, but the trend is not statistically distinguishable from zero, and the reliability horizon doubles at a rate indistinguishable from the capability horizon. Contrary to a pre-specified hypothesis, the 80% horizon does not double more slowly than the 50% horizon. Outcome inconsistency peaks for tasks just below each model's capability horizon. A further, unanticipated result is that repeated attempts are strongly correlated within tasks (intraclass correlation 0.68–0.87): observed reliability horizons are 1.6 to 8.0 times longer than independent attempts would imply. Agent failures are therefore mostly task-determined rather than stochastic. The article discusses, as an implication rather than a finding, why capability disclosures for autonomous agents should report reliability-adjusted horizons alongside average-success horizons.

1. Introduction

Large language model agents now plan, call tools and act over many steps with limited human supervision (Wang et al., 2024a; Sapkota et al., 2026). Evaluations of these systems have moved from short question-answer benchmarks to long, multi-step tasks in software engineering, web navigation and operating-system control (Jimenez et al., 2024; Liu et al., 2024; Xie et al., 2024; Zhou et al., 2024). One influential summary of this progress is the time horizon proposed by Kwa et al. (2025): the duration of task, measured in the time a human expert takes to complete it, that an agent completes with 50% probability. Kwa et al. report that this horizon has doubled roughly every seven months since 2019, a figure that is now widely used in discussions of AI capability trajectories.

A 50% horizon is a measure of what an agent can do. It is not a measure of what an agent can be trusted to do. A deployer who delegates a recurring task cares about whether the agent succeeds each time the task appears, and a regulator who classifies a system by its degree of autonomy cares about the tasks it can carry out without human correction (Chan et al., 2023; Shavit et al., 2023; Feng et al., 2025). Benchmarks of tool-using agents have shown that success measured over several repeated trials, the pass^k metric, falls well below single-trial success (Yao et al., 2024; Wang et al., 2025). These two lines

of work have not been joined: the horizon literature measures average success as a function of task length, and the reliability literature measures repeated-trial success on tasks without a calibrated length.

This article joins them. It defines a reliability horizon, the task length at which an agent succeeds on all k attempts with 50% probability, and estimates it for 50 model-benchmark combinations using the public run-level records of the METR evaluations. The analysis tests four pre-specified hypotheses about the size of the reliability gap, its evolution across model generations, where outcome inconsistency concentrates, and the relative growth of high-success and median-success horizons.

The contribution is threefold. First, it provides the first reliability-adjusted time horizons for frontier agents, with cluster-bootstrap intervals. Second, it shows that the gap between capability and reliability horizons is large (a factor of about 2.4 at $k = 4$) and has not narrowed significantly between 2023 and early 2026. Third, it documents that repeated attempts are strongly correlated within tasks, so that observed reliability is much higher than an independence model predicts. This last result changes how repeated-trial evaluations and retry-based deployment strategies should be interpreted, and it is discussed as an implication for capability disclosure in Section 6.

2. Related literature and research gap

Agent benchmarks and capability horizons. Surveys of agent evaluation describe a rapid proliferation of benchmarks for web, software, operating-system and domain-specific tasks (Mohammadi et al., 2025; Bandi et al., 2025). Most report a success rate over a fixed task set, which is difficult to compare across benchmarks and over time. Kwa et al. (2025) address this by calibrating tasks against human completion time, drawing on three task suites: HCAST (Rein et al., 2025), RE-Bench (Wijk et al., 2024) and a set of short software actions. Their 50% and 80% horizons are computed from average success. They do not separate the variance of outcomes across attempts on the same task from variance across tasks.

Repeated-trial reliability. The $\text{pass}@k$ metric, introduced for code generation, estimates the probability that at least one of k samples succeeds (Chen et al., 2021). Yao et al. (2024) introduced its pessimistic counterpart, pass^k , the probability that all k attempts succeed, and showed that it decays sharply with k for tool-using agents in customer-service domains; ECom-Bench reports similar decay in e-commerce support (Wang et al., 2025). Other work aggregates repeated samples to improve accuracy (Wang et al., 2023; Wang et al., 2024b), treats output non-determinism as a property of the serving stack (Ouyang et al., 2023), or argues that agent benchmarks should report cost and variability alongside accuracy (Kapoor et al., 2024). None of these studies relates repeated-trial reliability to calibrated task length.

Governance of agentic systems. The governance literature argues that the risks of agents grow with their autonomy and that deployers need visibility into what agents do (Chan et al., 2023; Chan et al., 2024; Shavit et al., 2023). Proposals include levels of autonomy (Feng et al., 2025), accountability frameworks for human oversight (Bahidika, 2026; Uriol et al., 2026) and the argument that fully autonomous agents

should not be developed at all (Mitchell et al., 2025). These proposals presuppose that the reliability of an agent can be measured, but they rely on capability measures that do not capture it.

Bibliometric verification. The gap was checked in Scopus on 20 October 2026 with five advanced-search queries (Table 1; exact strings are in the replication package). The query crossing agents, task length and repeated-trial reliability returned 31 records, all of which also appear in the task-length query. Each record was read. Kwa et al. (2025) is the only empirical study of task-length horizons; the other 30 are architectures for long-horizon planning or memory, surveys, or domain applications. The ten records retrieved by the repeated-trial query report consistency on fixed tasks without a time calibration, and none of them appears in the task-length query. The two literatures exist, but they do not intersect.

Table 1
Scopus search funnel (20 October 2026)

Search block	Records	Content after reading
Agents and benchmarks (LLM agents $\wedge$ benchmark or evaluation, from 2023)	3,086	Broad corpus; used for citation verification
Agents and task length (time horizon, task length or duration, long-horizon)	192	One empirical horizon study (Kwa et al., 2025); otherwise architectures and surveys
Agents and repeated-trial reliability (repeated or multiple runs or trials)	10	Consistency on fixed tasks; no calibration by task length
Agents, autonomy and governance (oversight, regulation, reliability)	98	Conceptual frameworks; no reliability measurement
Intersection: agents $\wedge$ task length $\wedge$ repeated-trial reliability	31	All within the task-length block; none estimates reliability as a function of task length
Note. Searches in TITLE-ABS-KEY. Exact query strings are in the replication package. The repeated-trial block and the task-length block share no records.		

3. Data

The data are the public run-level records released by METR with the time-horizon evaluations (repository METR/eval-analysis-public, commit 52cb829, retrieved 20 October 2026). Two report versions are available. Version 1.0 contains 41,629 runs and version 1.1 contains 24,008 runs; version 1.1 re-evaluates most models with a different agent scaffold, adds 73 tasks and drops 15. Each record identifies the model, the task and its family, the binary outcome of the run, and the geometric-mean completion time of human experts who attempted the task (Kwa et al., 2025).

Three filters define the analysis sample. Runs by human baseliners are excluded. Models released before 2023 (GPT-2, GPT-3 and gpt-3.5-turbo-instruct) are excluded because their success rates are close to zero on all but the shortest tasks, which makes their fitted curves degenerate. Model-task cells with fewer than four attempts are excluded so that pass^4 is defined for every cell; this removes 1.2% of cells

in version 1.0 and 1.7% in version 1.1. The resulting sample contains 60,417 attempts in 9,473 model-task cells (Table 2). Task durations range from about one second to 30 hours of human expert time. Release dates are taken from the same repository.

The repository does not include a licence file. For that reason the replication package does not redistribute the run records; it contains a script that retrieves them from the pinned commit and verifies their SHA-256 checksums before any computation, together with all derived variables.

Table 2
Analysis sample

Models (released 2023 or later)	Version 1.0	Version 1.1
	30	20
Tasks	170	228
Task families	50	79
Model-task cells with at least four attempts	5,027	4,446
Attempts	37,395	23,022
Median attempts per cell	8	4
Human completion time, minutes (range)	0.02–1,112	0.02–1,800
Note. Source: METR time-horizon run records (commit 52cb829). Human baseline runs and models released before 2023 are excluded.		

4. Methods

Cell-level estimators. For model i and task j , let n_{ij} be the number of attempts and c_{ij} the number of successes. Average success is estimated by

$$\hat{p}_{ij} = c_{ij} / n_{ij}. \quad (1)$$

Repeated-trial success is estimated with the unbiased estimator of Yao et al. (2024), which counts the share of k -subsets of attempts in which every attempt succeeds:

$$\hat{\pi}_{ij}(k) = C(c_{ij}, k) / C(n_{ij}, k), \quad n_{ij} \geq k, \quad (2)$$

where $C(\cdot, \cdot)$ is the binomial coefficient. If the attempts on a task are exchangeable with success probability p_{ij} , $\hat{\pi}_{ij}(k)$ is unbiased for p_{ij}^k . The baseline uses $k = 4$, the largest value available for almost all cells; $k = 2$ and $k = 3$ are reported as robustness checks.

Horizons. For each model, success is regressed on log task duration with a fractional logit (Papke and Wooldridge, 1996), as in Kwa et al. (2025):

$$\text{logit } E[y_i] = \alpha_i + \beta_i \log_2 m_i, (3)$$

where y_i is either $\hat{p}_i$ or $\hat{\pi}_i(4)$, m_i is the human completion time in minutes, and each task is weighted by the inverse square root of the number of tasks in its family, the weighting used by METR. The horizon at success level q is

$$h_i(q) = 2^{[(\text{logit } q - \alpha_i) / \beta_i]}. (4)$$

The capability horizon $h_i(0.5)$ and the 80% horizon $h_i(0.8)$ come from the average-success curve. The reliability horizon $h_i^{(4)}$ is the 50% point of the curve fitted to $\hat{\pi}_i(4)$. The reliability gap is the ratio $R_i = h_i(0.5) / h_i^{(4)}$.

Independence benchmark. If the attempts on a task were independent with the probability given by the average-success curve, all four would succeed with probability 0.5 exactly when the per-attempt probability is $0.5^{(1/4)} \approx 0.841$. The horizon implied by independence is therefore

$$h_i^{n_i} = h_i(0.5^{(1/k)}). (5)$$

Comparing $h_i^{(4)}$ with $h_i^{n_i}$ measures how far observed reliability departs from independence. Within-task dependence is summarised by an intraclass correlation,

$$\rho_i = 1 - \sum \omega_i s_i^2 / [\bar{p}_i (1 - \bar{p}_i)], s_i^2 = \hat{p}_i (1 - \hat{p}_i) n_i / (n_i - 1), (6)$$

where ω_i are the normalised task weights and $\bar{p}_i$ is the weighted mean success rate. ρ_i is the share of outcome variance that lies between tasks rather than within them (Shrout and Fleiss, 1979); $\rho_i = 0$ corresponds to independent, identically distributed attempts across tasks of equal difficulty, and $\rho_i = 1$ to fully task-determined outcomes.

Trends. For each horizon, $\log_2 h_i$ is regressed on model release date t_i in years:

$$\log_2 h_i = \gamma + \delta t_i + \varepsilon_i, \text{ doubling time} = 12 / \delta \text{ months}. (7)$$

Location of inconsistency. A cell has mixed outcomes when $0 < c_i < n_i$. Task difficulty relative to each model is $z_i = \log_2 m_i - \log_2 h_i(0.5)$, and the share of mixed cells is modelled as

$$M_i = \theta_0 + \theta_1 z_i + \theta_2 z_i^2 + u_i, (8)$$

estimated by weighted least squares with standard errors clustered by task family; the peak lies at $z^* = -\theta_1 / (2\theta_2)$.

Hypotheses. Four hypotheses were fixed before estimation. H1: for every model, the reliability horizon is shorter than the capability horizon ($R_i > 1$). H2: the reliability gap does not narrow with model release date (the slope of $\ln R_i$ on t_i is not negative). H3: outcome inconsistency is highest for tasks of

intermediate difficulty relative to each model's capability, not for the longest tasks ($\theta_2 < 0$, with the peak near $z = 0$). H4: the 80% horizon doubles more slowly than the 50% horizon.

Inference. Uncertainty is quantified with a cluster bootstrap that resamples task families with replacement within each benchmark version (1,000 replicates; Efron and Tibshirani, 1993; Cameron and Miller, 2015) and re-estimates every model and trend in each replicate. Intervals are 2.5th and 97.5th percentiles. The fractional-logit estimates were checked against a reference implementation (maximum absolute coefficient difference 1.9×10^{-6}).

5. Results

5.1 Capability and reliability horizons (H1)

Table 3 reports the horizons for the 20 models evaluated in benchmark version 1.1; Table A1 in the Appendix reports the 30 models of version 1.0. In every one of the 50 model-version combinations the reliability horizon is shorter than the capability horizon, and in every case the bootstrap interval of the ratio excludes one. The ratio ranges from 1.75 to 3.85 in version 1.1 and from 1.58 to 4.75 in version 1.0, with geometric means of 2.39 (95% CI 2.07–2.84) and 2.33 (1.98–2.81). H1 is supported.

For the most capable model in the data, Claude Opus 4.6, the capability horizon in version 1.1 is about 715 minutes, while the horizon at which four consecutive attempts all succeed with 50% probability is about 355 minutes. For GPT-5.2 the corresponding figures are 354 and 192 minutes. Figure 1 shows the three horizons by release date, and Fig. 2 shows the fitted curves for these two models.

Table 3
Capability, reliability and independence-implied horizons, benchmark version 1.1

Model	Release	$h(0.5)$	$h(0.8)$	$h^{(4)}$	h^{indep}	Ratio $h(0.5)/h^{(4)}$ [95% CI]	ICC
GPT-4 0314	2023-03-14	3.9	0.9	1.3	0.6	3.04 [2.16, 4.67]	0.74
GPT-4 1106	2023-11-06	4.0	0.7	1.5	0.5	2.57 [1.74, 3.92]	0.75
Claude 3 Opus	2024-03-04	3.7	0.5	1.3	0.4	2.84 [1.78, 5.40]	0.71
GPT-4 Turbo	2024-04-09	3.7	0.9	1.7	0.7	2.14 [1.43, 3.32]	0.78
GPT-4o	2024-05-13	6.8	1.2	1.8	0.8	3.85 [2.26, 6.87]	0.68
Claude 3.5 Sonnet (Old)	2024-06-20	11.2	1.4	4.8	0.9	2.35 [1.73, 3.30]	0.72
o1-preview	2024-09-12	20.4	4.6	6.1	3.4	3.37 [2.45, 4.64]	0.73
Claude 3.5 Sonnet (New)	2024-10-22	19.8	2.4	6.0	1.6	3.29 [2.16, 5.58]	0.71
o1	2024-12-05	38.7	7.0	12.6	5.0	3.08 [2.10, 4.72]	0.74
Claude 3.7 Sonnet	2025-02-24	62.5	12.1	25.1	8.7	2.49 [1.75, 4.04]	0.81
o3	2025-04-16	128.5	32.4	67.4	24.5	1.91 [1.49, 2.57]	0.80
Claude 4 Opus	2025-05-22	101.1	21.6	42.2	15.8	2.39 [1.76, 3.45]	0.75
Claude 4.1 Opus	2025-08-05	101.2	23.3	49.8	17.3	2.03 [1.55, 2.84]	0.80
GPT-5	2025-08-07	207.2	38.8	103.7	27.7	2.00 [1.52, 2.84]	0.81
Gemini 3 Pro	2025-11-18	224.4	54.8	115.7	41.3	1.94 [1.39, 2.94]	0.79
GPT-5.1-Codex-Max	2025-11-19	224.4	50.7	107.2	37.6	2.09 [1.59, 2.87]	0.79
Note. Horizons in minutes of human expert time. $h(0.5)$ and $h(0.8)$: average-success horizons; $h^{(4)}$: reliability horizon (pass ⁴ = 0.5); h^{indep} : horizon implied by independent attempts (Eq. 5). Intervals from a task-family cluster bootstrap (1,000 replicates). ICC: intraclass correlation (Eq. 6).							

Model	Release	$h(0.5)$	$h(0.8)$	$h^{(4)}$	$h^{n\infty}$	Ratio $h(0.5)/h^{(4)}$ [95% CI]	ICC
Claude Opus 4.5	2025-11-24	292.7	49.9	167.1	34.9	1.75 [1.34, 2.58]	0.84
GPT-5.2	2025-12-11	353.7	64.1	191.6	45.5	1.85 [1.33, 3.01]	0.84
Claude Opus 4.6	2026-02-05	714.6	70.8	354.9	44.5	2.01 [1.37, 4.19]	0.82
GPT-5.3-Codex	2026-02-05	366.0	49.4	179.9	33.1	2.03 [1.45, 3.29]	0.80
Note. Horizons in minutes of human expert time. $h(0.5)$ and $h(0.8)$: average-success horizons; $h^{(4)}$: reliability horizon ($\text{pass}^4 = 0.5$); $h^{n\infty}$: horizon implied by independent attempts (Eq. 5). Intervals from a task-family cluster bootstrap (1,000 replicates). ICC: intraclass correlation (Eq. 6).							

5.2 Repeated attempts are strongly correlated

The reliability horizon is shorter than the capability horizon, but it is much longer than independence would imply. Under independence, the reliability horizon would equal the horizon at which average success reaches 84.1%, which lies below even the 80% horizon. Instead, the observed reliability horizon lies between the 80% and the 50% horizons for every model (Table 3). The ratio of the observed reliability horizon to the independence benchmark ranges from 1.6 to 8.0, and its bootstrap interval excludes one in 49 of the 50 model-version combinations (the exception is GPT-4 0314 in version 1.0).

The intraclass correlation explains why. Between 68% and 87% of outcome variance lies between tasks rather than between attempts on the same task (median 0.79 in both versions). When an agent fails on a task, it tends to fail on every attempt; when it succeeds, it tends to succeed on every attempt. The dashed curves in Fig. 2 show how much faster reliability would decay with task length if attempts were independent. This result was not anticipated in the pre-specified hypotheses and is reported as exploratory.

5.3 Trends across model generations (H2 and H4)

Table 4 reports the trend regressions. The capability horizon doubles every 4.0 months in version 1.1 (95% CI 3.3–5.0) and every 5.5 months in version 1.0 (4.5–7.2). These figures describe the post-2023 models in the sample and are shorter than the seven-month figure that Kwa et al. (2025) estimate from 2019 onwards, consistent with their observation that the trend may have accelerated.

The reliability horizon doubles every 3.7 months (3.1–4.5) and 4.9 months (3.9–6.8), respectively. Its slope exceeds that of the capability horizon by 0.28 doublings per year in both versions, but the bootstrap intervals include zero. Equivalently, the log ratio declines by 0.19 per year, a narrowing of about 17% per year in point estimate, with intervals of $[-0.40, 0.03]$ and $[-0.47, 0.07]$. The gap therefore shows a tendency to narrow, but the narrowing is not statistically distinguishable from zero over the period covered, and H2 is not rejected. Figure 4 shows the ratio by release date.

H4 is not supported. The 80% horizon doubles every 4.1 months in version 1.1 and 5.5 months in version 1.0, virtually the same rate as the 50% horizon; the slope differences are – 0.04 and – 0.01 doublings per year, with intervals centred on zero. The pre-specified expectation that high-success horizons grow more slowly than median horizons is not borne out in these data.

Table 4
Trends across model release dates

Quantity	Version 1.1	Version 1.0
Doubling time of $h(0.5)$, months	4.02 [3.30, 4.95]	5.50 [4.50, 7.17]
Doubling time of $h(0.8)$, months	4.07 [3.35, 5.18]	5.51 [4.04, 9.36]
Doubling time of $h^{(4)}$, months	3.68 [3.10, 4.49]	4.88 [3.89, 6.85]
Slope of $\ln[h(0.5)/h^{(4)}]$ per year (H2)	-0.19 [-0.40, 0.03]	-0.19 [-0.47, 0.07]
Slope of $\log_2 h^{(4)}$ minus slope of $\log_2 h(0.5)$, per year	0.28 [-0.04, 0.57]	0.27 [-0.10, 0.67]
Slope of $\log_2 h(0.8)$ minus slope of $\log_2 h(0.5)$, per year (H4)	-0.04 [-0.57, 0.35]	-0.01 [-0.59, 0.52]
Geometric-mean ratio $h(0.5)/h^{(4)}$	2.39 [2.07, 2.84]	2.33 [1.97, 2.81]
Note. Point estimates from Eq. 7 with 95% cluster-bootstrap intervals in brackets. Doubling-time intervals are percentile intervals of $12/\delta$.		
5.4 Where inconsistency concentrates (H3)		

Table 5 and Fig. 3 report the share of model-task cells with mixed outcomes by task difficulty relative to each model's capability horizon. Far below the horizon ($z \leq -6$) agents almost always succeed on every attempt, and far above it ($z > 6$) they almost always fail; mixed outcomes occur in fewer than 4% of cells at both extremes. The share rises to 31–40% for tasks within a factor of four of the capability horizon. The quadratic term in Eq. 8 is negative and significant in both versions ($p < 10^{-5}$), and the estimated peak lies just below the capability horizon, at 0.69 times the horizon in version 1.0 and 0.86 times in version 1.1. H3 is supported: inconsistency is concentrated near each model's capability frontier, not on the longest tasks, where failure is consistent rather than erratic.

Table 5
Mixed outcomes by task difficulty relative to each model's capability horizon

$z = \log_2(m / h(0.5))$	Cells (1.1)	Mixed	pass@1	pass^4	Cells (1.0)	Mixed	pass@1	pass^4
$z \leq -6$	1,259	0.030	0.980	0.967	1,779	0.026	0.993	0.982
$-6 < z \leq -4$	567	0.073	0.938	0.898	344	0.053	0.958	0.933
$-4 < z \leq -2$	496	0.173	0.886	0.791	489	0.185	0.836	0.751
$-2 < z \leq 0$	548	0.321	0.586	0.438	736	0.392	0.684	0.525
$0 < z \leq 2$	701	0.305	0.389	0.239	710	0.400	0.366	0.215
$2 < z \leq 4$	430	0.294	0.165	0.041	605	0.289	0.115	0.020
$4 < z \leq 6$	227	0.124	0.044	0.000	277	0.067	0.017	0.000
$z > 6$	218	0.033	0.009	0.000	87	0.013	0.002	0.000
Note. Weighted shares. Quadratic model (Eq. 8): $\theta_2 = -0.0028$ ($p = 9.3e-10$) and peak at $0.86 \times h(0.5)$ in version 1.1; $\theta_2 = -0.0040$ ($p = 1.1e-06$) and peak at $0.69 \times h(0.5)$ in version 1.0. Standard errors clustered by task family.								

5.5 Robustness

Table 6 reports alternative specifications. The reliability gap grows with the number of attempts required: the geometric-mean ratio is 1.59 at $k = 2$, 2.01–2.02 at $k = 3$ and 2.33–2.39 at $k = 4$. Weighting every task family equally, excluding the very short software actions, and restricting the sample to cells with at least six attempts leave the ratio between 2.36 and 2.71. In every specification the observed reliability horizon remains well above the independence benchmark, with geometric-mean ratios between 1.57 and 3.26. One instability is noted: with the six-attempt restriction, a model with few remaining long tasks yields a ratio of 28.8 in version 1.1, which reflects a poorly identified curve rather than a substantive change, and the geometric mean is barely affected.

Table 6
Robustness of the reliability gap and of the independence benchmark

Specification	Version	Models	Geometric-mean ratio $h(0.5)/h(\frac{1}{k})$	Range	Geometric-mean ratio $h(\frac{1}{k})/h(\frac{1}{n})$
Baseline: $k = 4$, inverse-square-root family weights	1.0	30	2.33	1.58–4.75	2.37
Baseline: $k = 4$, inverse-square-root family weights	1.1	20	2.39	1.75–3.85	3.26
$k = 2$	1.0	30	1.59	1.27–2.09	1.57
$k = 2$	1.1	20	1.59	1.29–2.02	1.84
$k = 3$	1.0	30	2.01	1.41–3.32	1.99
$k = 3$	1.1	20	2.02	1.55–2.85	2.59
Equal weight per task family	1.0	30	2.44	1.59–4.34	2.14
Equal weight per task family	1.1	20	2.43	1.83–4.13	2.98
Excluding SWAA (tasks under one minute)	1.0	30	2.47	1.53–8.13	2.29
Excluding SWAA (tasks under one minute)	1.1	20	2.71	1.74–7.56	3.06
Cells with at least six attempts	1.0	29	2.36	1.59–4.75	2.23
Cells with at least six attempts	1.1	20	2.60	1.00–28.75	1.65
Note. $k = 4$ unless stated. SWAA: short software actions. The independence benchmark uses the per-attempt probability $0.5^{(1/k)}$.					

6. Discussion

The results support three conclusions about how agent capability should be read. First, a horizon based on average success overstates the length of task that an agent can be relied on to complete. For the frontier models in these data, asking that four attempts all succeed halves the horizon, and the gap has not closed significantly across three years of model releases. Second, the gap is much smaller than a naive independence calculation suggests, because agent outcomes are mostly determined by the task.

Third, inconsistency is concentrated just below the capability frontier, which is where newly delegated tasks are most likely to fail.

Implications for capability disclosure (analogy, not a finding about deployed systems). The following points are implications drawn from benchmark evidence and are offered as hypotheses for future work.

Governance proposals for agentic systems rely on capability measures to decide how much autonomy a system may receive (Shavit et al., 2023; Feng et al., 2025). If such measures report only average-success horizons, they describe a frontier at which roughly half of recurrent executions would include at least one failure. A reliability-adjusted horizon, reported with its k and its interval, would make that difference visible at no additional evaluation cost when repeated attempts are already run. The high intraclass correlation has two further consequences. Retrying a failed task is a weak safeguard, because a task that fails once is likely to fail again; and an audit that runs each task only once already captures most of the reliability risk, so the value of repeated runs lies mainly in identifying the minority of tasks with mixed outcomes. Both consequences point towards task-level disclosure: knowing which kinds of task fail consistently is more informative for oversight than an aggregate success rate (Chan et al., 2024; Kapoor et al., 2024).

Limitations. The evidence comes from one organisation's task suites, which are dominated by software and research-engineering tasks; reliability in other domains may behave differently (Mohammadi et al., 2025). The number of attempts per cell is modest (median four in version 1.1), which limits the precision of individual cell estimates, although the estimator is unbiased. Human completion time is measured with error and some long tasks have estimated rather than observed times (Kwa et al., 2025). The trend analysis uses 20 and 30 models over three years, so the power to detect a narrowing gap is limited; the non-significant narrowing should not be read as evidence that the gap is constant. Finally, scaffolds differ between benchmark versions and model generations, so the horizons describe model-scaffold combinations rather than models alone.

7. Conclusion

Time horizons have become a common language for describing what AI agents can do. This article shows that the horizon at which an agent succeeds reliably across repeated attempts is about 2.4 times shorter than the horizon at which it succeeds half of the time, that this gap has not narrowed significantly between 2023 and early 2026, and that agent failures are largely task-determined. Reporting reliability-adjusted horizons alongside average-success horizons would give deployers and regulators a measure closer to the question they need to answer: not what an agent can do, but what it can be trusted to do each time.

Declarations

Competing interests.

The author declares no competing interests.

Ethics approval.

Not applicable. The study uses only publicly released evaluation records of AI systems and involves no human participants, animals or personal data.

Use of generative AI.

Claude (Anthropic; model Claude Opus 5.5) was used to draft and debug analysis code, to process the data and to edit the language of the manuscript. It was used to accelerate implementation and editing. The author specified the research design and hypotheses, verified every reference against Scopus records, checked all numerical results against the code output, and takes full responsibility for the content of the article.

Funding.

This research received no specific grant from any funding agency in the public, commercial or not-for-profit sectors.

Author contributions.

A.A.M.E. conceived the study, designed the analysis, verified the data and results, and wrote and revised the manuscript.

Data availability.

The analysis uses the public run-level records of the METR time-horizon evaluations (GitHub repository METR/eval-analysis-public, commit 52cb829). Because the repository carries no licence file, the raw records are not redistributed. The replication package, archived at Zenodo (<https://doi.org/10.5281/zenodo.23510553>), contains a script that retrieves them from the pinned commit and verifies their SHA-256 checksums, together with all derived variables, tables, figures and the Scopus search strings.

Code availability.

All Python code needed to reproduce every number, table and figure in this article is included in the replication package at <https://doi.org/10.5281/zenodo.23510553>, released under a CC BY 4.0 licence.

References

1. Bahidika O (2026) Agentic accountability: The buck stops where? Ethical frameworks for human oversight of autonomous AI systems. *Int J Adv Comput Sci Appl* 17(5):13–22. <https://doi.org/10.14569/IJACSA.2026.0170502>
2. Bandi A, Kongari B, Naguru R, Pasnoor S, Vilipala SV (2025) The rise of agentic AI: A review of definitions, frameworks, architectures, applications, evaluation metrics, and challenges. *Future Internet* 17(9):404. <https://doi.org/10.3390/fi17090404>
3. Cameron AC, Miller DL (2015) A practitioner’s guide to cluster-robust inference. *J Hum Resour* 50(2):317–372. <https://doi.org/10.3368/jhr.50.2.317>
4. Chan A, Salganik R, Markelius A, Pang C, Rajkumar N, Krashenninnikov D et al (2023) Harms from increasingly agentic algorithmic systems. In *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency* (pp. 651–666). <https://doi.org/10.1145/3593013.3594033>
5. Chan A, Ezell C, Kaufmann M, Wei K, Hammond L, Bradley H et al (2024) Visibility into AI agents. In *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency* (pp. 958–973). <https://doi.org/10.1145/3630106.3658948>
6. Chen M, Tworek J, Jun H, Yuan Q, Pinto HP, de Kaplan O et al (2021) J.,. Evaluating large language models trained on code. *arXiv:2107.03374*
7. Efron B, Tibshirani RJ (1993) An introduction to the bootstrap. Chapman & Hall
8. Feng KJK, McDonald DW, Zhang AX (2025) Levels of autonomy for AI agents. *arXiv:2506.12469*
9. Jimenez CE, Yang J, Wettig A, Yao S, Pei K, Press O, Narasimhan K (2024) SWE-bench: Can language models resolve real-world GitHub issues? In *The Twelfth International Conference on Learning Representations*
10. Kapoor S, Stroebl B, Siegel ZS, Nadgir N, Narayanan A (2024) AI agents that matter. *arXiv:2407.01502*
11. Kwa T, West B, Becker J, Deng A, Garcia K, Hasin M et al (2025) Measuring AI ability to complete long software tasks. *Adv Neural Inf Process Syst* 38:102382–102435
12. Liu X, Yu H, Zhang H, Xu Y, Lei X, Lai H et al (2024) AgentBench: Evaluating LLMs as agents. In *12th International Conference on Learning Representations, ICLR 2024*
13. Mitchell M, Ghosh A, Luccioni AS, Pistilli G (2025) Fully autonomous AI agents should not be developed. *arXiv:2502.02649*
14. Mohammadi M, Li Y, Lo J, Yip W (2025) Evaluation and benchmarking of LLM agents: A survey. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Vol. 2, pp. 6129–6139). <https://doi.org/10.1145/3711896.3736570>
15. Ouyang S, Zhang JM, Harman M, Wang M (2023) LLM is like a box of chocolates: The non-determinism of ChatGPT in code generation. *arXiv:2308.02828*
16. Papke LE, Wooldridge JM (1996) Econometric methods for fractional response variables with an application to 401(k) plan participation rates. *J Appl Econom* 11(6):619–632

17. Rein D, Becker J, Deng A, Nix S, Canal C, O’Connell D et al (2025) HCAST: Human-calibrated autonomy software tasks. *arXiv:2503.17354*
18. Sapkota R, Roumeliotis KI, Karkee M (2026) AI agents vs. agentic AI: A conceptual taxonomy, applications and challenges. *Inform Fusion* 126:103599. <https://doi.org/10.1016/j.inffus.2025.103599>
19. Shavit Y, Agarwal S, Brundage M, Adler S, O’Keefe C, Campbell R et al (2023) Practices for governing agentic AI systems. OpenAI. [White paper]
20. Shrout PE, Fleiss JL (1979) Intraclass correlations: Uses in assessing rater reliability. *Psychol Bull* 86(2):420–428. <https://doi.org/10.1037/0033-2909.86.2.420>
21. Uriol J, O’Brien E, Hernández I, Viola R, Iradier E, Montalbán J (2026) Accountable autonomy: A governance framework for agentic AI in telecommunication networks. *AI* 7(9):347. <https://doi.org/10.3390/ai7090347>
22. Wang H, Peng X, Huang X, Huang Y, Gong M, Yang C, Liu Y, Jiang L (2025) ECom-Bench: Can LLM agent resolve real-world e-commerce customer support issues? In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track* (pp. 276–284). <https://doi.org/10.18653/v1/2025.emnlp-industry.19>
23. Wang H, Prasad A, Stengel-Eskin E, Bansal M (2024b) Soft self-consistency improves language model agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics* (Vol. 2, pp. 287–301). <https://doi.org/10.18653/v1/2024.acl-short.28>
24. Wang L, Ma C, Feng X, Zhang Z, Yang H, Zhang J et al (2024a) A survey on large language model based autonomous agents. *Front Comput Sci* 18(6):186345. <https://doi.org/10.1007/s11704-024-40231-1>
25. Wang X, Wei J, Schuurmans D, Le Q, Chi E, Narang S, Chowdhery A, Zhou D (2023) Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*
26. Wijk H, Lin T, Becker J, Jawhar S, Parikh N, Broadley T et al (2024) RE-Bench: Evaluating frontier AI R&D capabilities of language model agents against human experts. *arXiv:2411.15114*
27. Xie T, Zhang D, Chen J, Li X, Zhao S, Cao R et al (2024) OSWorld: Benchmarking multimodal agents for open-ended tasks in real computer environments. *Adv Neural Inf Process Syst*, 37
28. Yao S, Shinn N, Razavi P, Narasimhan K (2024) τ -bench: A benchmark for tool-agent-user interaction in real-world domains. *arXiv:2406.12045*
29. Zhou S, Xu FF, Zhu H, Zhou X, Lo R, Sridhar A et al (2024) WebArena: A realistic web environment for building autonomous agents. In *12th International Conference on Learning Representations, ICLR 2024*

Figures

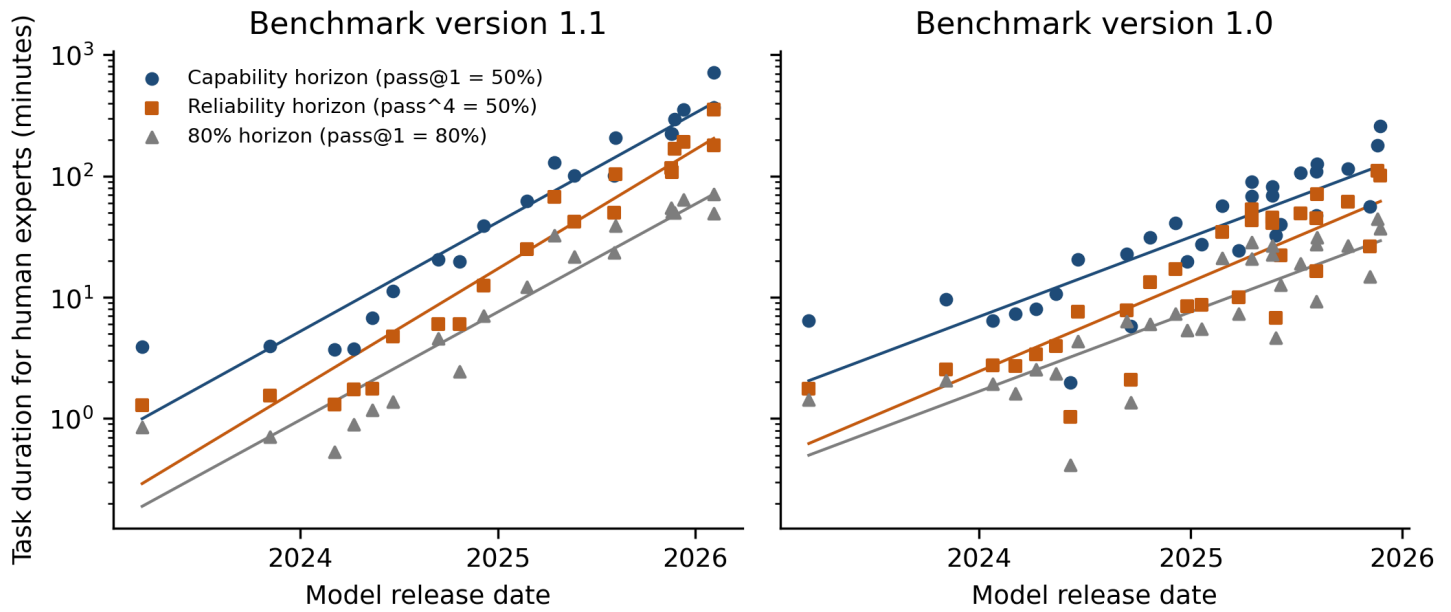


Figure 1

Capability horizon ($\text{pass}@1 = 50\%$), reliability horizon ($\text{pass}^4 = 50\%$) and 80% horizon by model release date, in minutes of human expert time (log scale). Lines are least-squares fits of Equation 7. Left: benchmark version 1.1 (20 models); right: version 1.0 (30 models).

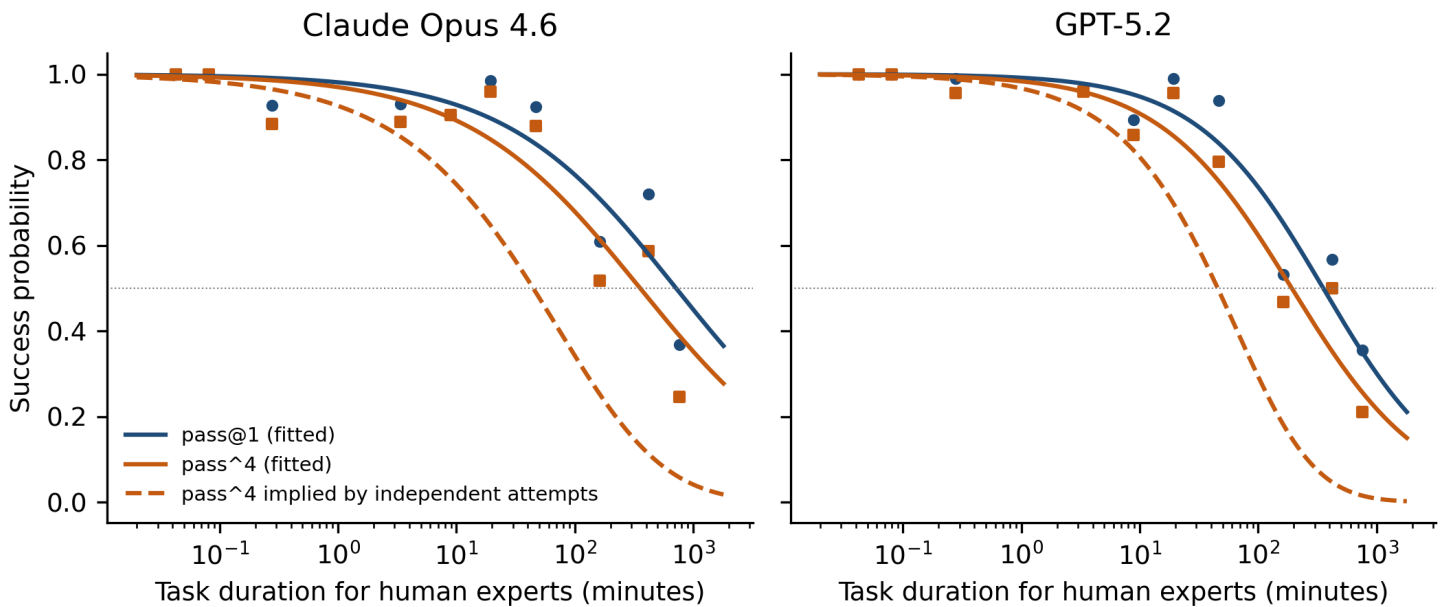


Figure 2

Fitted success curves for the two longest-horizon models in benchmark version 1.1. Solid lines: fractional-logit fits for average success ($\text{pass}@1$) and for success on all four attempts (pass^4). Dashed

line: pass^4 implied if attempts were independent, $(\text{pass}@1)^4$. Points: unweighted means within deciles of task duration.

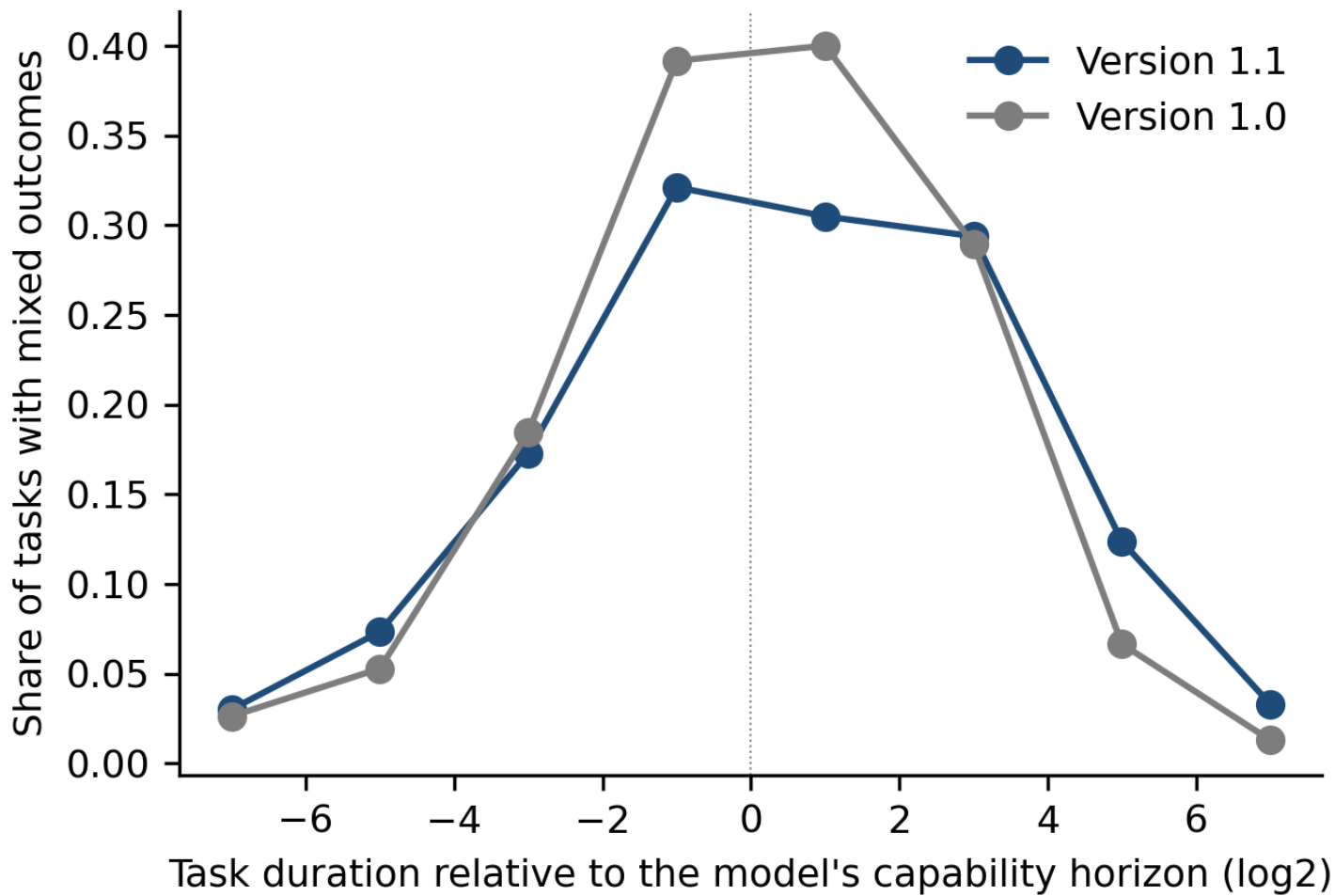


Figure 3

Share of model-task cells with mixed outcomes (some but not all attempts successful) by task duration relative to each model's capability horizon, $z = \log_2(m / h(0.5))$. Inconsistency peaks just below the capability horizon.

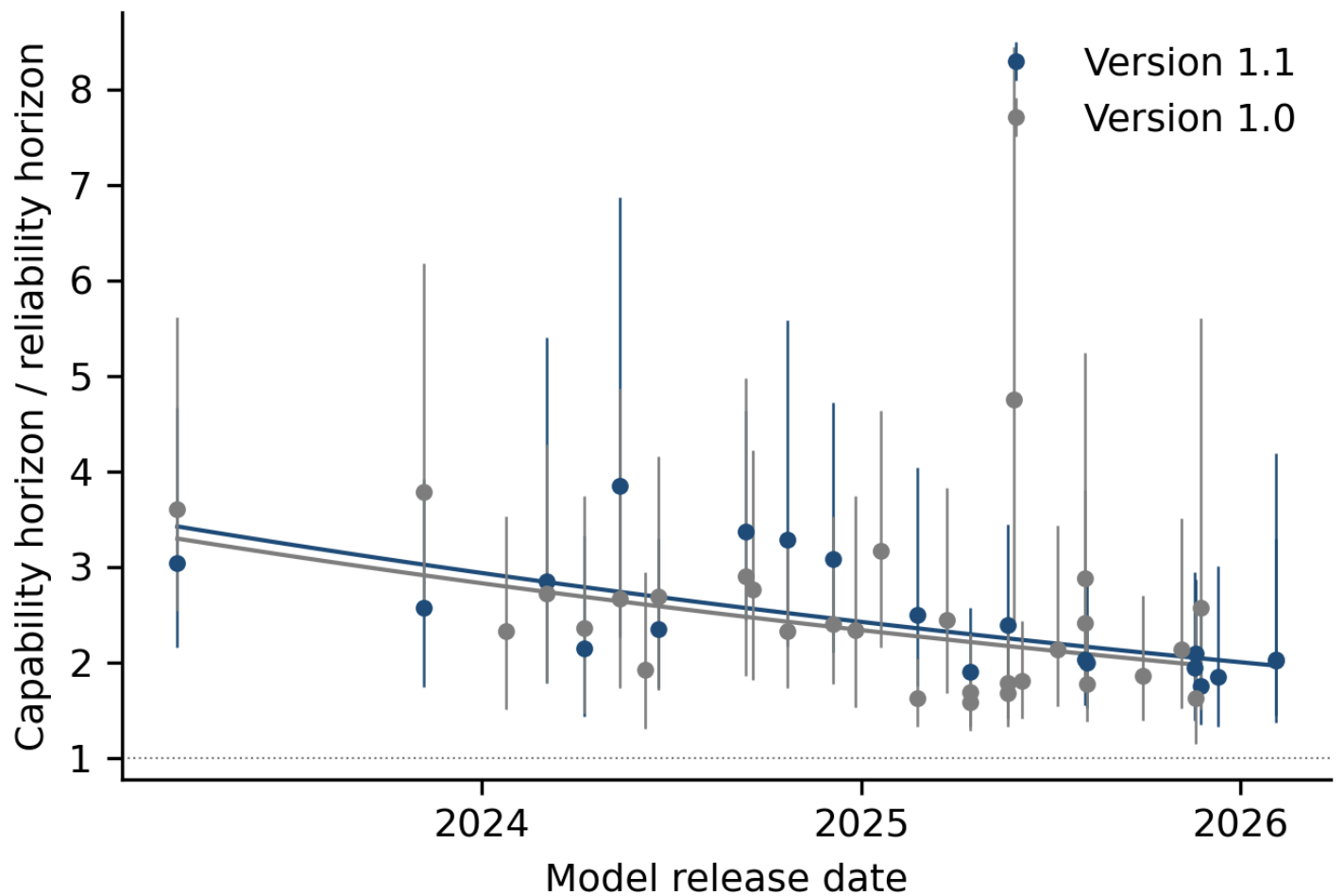


Figure 4

Ratio of the capability horizon to the reliability horizon by model release date, with 95% task-family cluster-bootstrap intervals. Lines are least-squares fits of the log ratio on release date; neither slope is statistically distinguishable from zero.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [Appendix.docx](#)