Vamshi Jandhyala

A benchmark is not a control

✠

June 2026

A benchmark ranks models. It does not certify a control.

A model tops a leaderboard. The number is real, the evaluation was careful, the result reproduces. The question that decides whether any of it matters to a regulated business is the one the leaderboard cannot answer: does a high score license you to let an agent act on a client's portfolio, a loan file, a suitability assessment, and stand behind the result when someone with statutory power asks how it was produced?

A benchmark ranks models. It does not certify a control. The two get conflated constantly, because the score arrives wrapped in the language of correctness: accuracy, pass rate, resolved. But a control in a regulated workflow is a different object from a high average on a public test set. A control has to hold on the specific decisions you make, work to a stated tolerance rather than an average, make its failures detectable and route them before they reach the client, be checkable after the fact by someone who was not in the room, and survive the world changing underneath it. The respected benchmarks were not built to measure any of that, and reading them as though they were is how a procurement decision goes wrong while every chart points up and to the right.

What follows is a field guide to the gap. Not a takedown of benchmarks, which are doing useful work, but an account of the six structural reasons a score and an acceptable output come apart, each shown in the benchmarks people actually cite. Once the structure is visible, you can read any leaderboard the way a risk function should: for what it licenses and what it does not.

What the benchmark calls correct

Start with the foundation, because it is shakier than the scores suggest. A benchmark score is a comparison against a reference answer, and the reference answer is made by people, who are fallible, working at speed on material that is often genuinely hard. When the reference is wrong, the score measures agreement with a mistake.

This is not a marginal worry. On the two most-used text-to-SQL benchmarks, a 2026 audit found the gold reference answers wrong in a majority of the cases examined, mostly because annotators had misread the schema. Correcting them moved measured accuracy by up to a third and reshuffled the rankings.[1] FrontierMath, whose launch headline was that frontier models solved under two percent, later shipped a

second version that "addressed errors in 42% of problems."[2] Humanity's Last Exam, assembled from nearly a thousand expert contributors, had answers contradicted by peer-reviewed literature in about 29% of the chemistry and biology questions one lab checked, and its own team later put the rate near 18% on a subset.[3] Even GPQA estimates that only about three quarters of its questions have uncontroversially correct answers.[4]

The lesson is not that these are bad benchmarks. They are among the best, which is the point: if the people who build the reference answers get them wrong this often, a single confident score is resting on a foundation that the builders themselves cannot fully stand behind. In a regulated workflow you do not get to treat "the benchmark says correct" as ground truth. The ground truth is what a competent professional, and eventually an examiner, will accept, and that is a higher and different bar than agreement with a crowdsourced key.

Passing the test is not finishing the job

The most important gap has a clean name, borrowed from software: a change can pass every test and still be one no professional would merge. SWE-bench, the benchmark that asks a model to resolve a real GitHub issue, scores success by applying the model's patch and running the project's unit tests; if they pass, the issue is counted resolved.[5] That is execution success. It is not the same as a maintainer accepting the change.

The benchmark's own history is the proof. When OpenAI and the original authors built SWE-bench Verified, a human-screened subset, they did it because annotators found the test suites "often overly specific, and in some cases even unrelated to the issue," and found issue descriptions underspecified to the point of ambiguity. Over two thirds of sampled instances were filtered out for problems unrelated to model capability. Cleaning the set roughly doubled the same model's measured score, from 16% to 33.2% for GPT-4o, by removing tasks that were being graded unfairly in both directions.[6] The gap between "the tests went green" and "this resolves the issue a person actually raised" was large enough to move the headline number by half.

The repair fixed which tasks were graded fairly. It could not change what the verdict is made of, which is a test suite, and a test suite can be told what to say. That is a separate failure taken up below.

This is the gap that matters most in finance, where almost nothing of consequence is verifiable by a unit test. A trade allocation can reconcile to the penny and still breach the client's mandate. A generated suitability rationale can be fluent, on-template, and wrong about the client. When the only available check is an automatic one, a high score certifies that outputs pass the check, not that a professional would accept them, and in regulated work those are different bars with different owners. Whether an output is acceptable to the person whose name goes against the decision is a separate measurement, and it is the one a regulated firm actually needs. I have argued the operational version of this elsewhere, that an answer that runs is not an answer you can trust. The benchmark version is that a task scored resolved is not a task professionally finished.

When the grader is a language model

Some benchmarks do not check answers against a key at all. They ask another language model to judge. This is now common, because it scales: Humanity's Last Exam uses a model to verify answer correctness, MT-Bench scores responses with a model grader on a ten-point scale, and a large share of the agent-evaluation literature leans on an LLM judge somewhere in the loop.[7] The convenience is real. So is the problem it introduces: the grader inherits all the failure modes of the thing it is grading, and the score inherits them from the grader.

The measurements are not subtle. Reordering the two candidate answers alone can reverse the ranking, enough for a weaker model to "beat" a stronger one on 66 of 80 questions purely by appearing first.[8] MT-Bench's own authors went looking for self-enhancement, a judge's tendency to prefer answers it generated itself. They measured GPT-4 favouring its own answers by ten points of win rate and Claude-v1 by twenty-five, then declined to conclude the effect was real, because the data was thin and the judges also favoured models other than themselves.[9] A 2026 preprint puts a number on the raw instability: a judge flipping its own verdict on repeated trials about 14% of the time, with 28% of questions exceeding a 20% flip rate, and roughly eleven repeated trials needed before majority voting recovers a reference verdict.[10]

For a regulated workflow this makes a judged score insufficient as a standalone control. A control has to be reproducible within a known tolerance: run it twice on the same input and you need to know how often the verdict changes, or it is not evidence an auditor can use. A judge that disagrees with itself one time in seven, and can be swung by answer order or verbosity, is not a control unless it is run many times with randomised positions and its uncertainty reported. Most leaderboards do none of that; they report a single judged number. The practical reading is simple. When a benchmark's grader is itself a language model, treat the score as an opinion poll with a biased panel of one, not as a measurement, until you see the multi-trial reliability behind it.

Defeating the grader instead of the task

The gaps so far are specification gaps: the grader faithfully checks a definition of success that is wrong, incomplete, or unstable. This one is different in kind. The agent does not satisfy the grader, it manipulates the mechanism that produces the verdict. A grader is a program, and a program can be defeated rather than persuaded.

When Berkeley researchers pointed an automated exploit agent at eight of the most-cited agent benchmarks in April 2026, it reached 100% on SWE-bench Verified, 100% on SWE-bench Pro, 100% on Terminal-Bench, close to 100% on WebArena and GAIA, and 73% on OSWorld. It solved nothing. In most runs it did not call a language model at all.[11]

The mechanisms were mundane. A pytest hook that forces every test to report as passing. A `file://` URL that serves the grader's answer key off local disk. An `eval()` call on text the agent controls. In one case a scoring function that never checked correctness. None of this establishes that published model scores were obtained this way. It says something about what the number is: a measurement of what a grader will accept, which is only ever as good as the grader. A verdict that can be produced without doing the work is not evidence that the work was done, and a firm that adopts

a benchmark as an assurance has quietly adopted the grader's bugs as well.

A number that decays

A benchmark score is a photograph, and the scene keeps moving. Two forces erode it: contamination, where the test questions leak into training data, and saturation, where models climb so high the benchmark stops telling models apart.

Contamination is measurable and large. LiveCodeBench, built specifically to date its problems and watch for this, found models scoring "considerably worse on problems released since" their training cutoff, with a "stark drop" at the boundary just before one model's release date, the signature of memorised earlier problems.[12] On the AIME competition maths often quoted in model cards, a controlled study pushed the 2024 score from 43% to 62% through contamination alone, while contamination detectors performed "near random guess."[13] The widely used practice of reporting last year's competition is partly an attempt to stay ahead of this, and it is a treadmill.

Saturation is the quieter problem. GPQA's own domain experts score 65%, and frontier systems passed that long ago, with a verified 87% recorded in mid-2025. Once a benchmark sits near its ceiling it can no longer rank the models you care about.[14] MMLU was superseded in practice for exactly this reason by a harder successor, MMLU-Pro, which raised the difficulty and expanded each question from four options to ten.[15] And the benchmarks built on human preference have a structural integrity problem of their own: an analysis of the most popular preference arena found that a small number of large providers received around a fifth of all the comparison data each, that one firm tested 27 private model variants before a release and published only the best, and that access to extra arena data alone could lift a model's measured win rate by up to 112%.[16] A number that can be farmed by whoever submits the most variants is a leaderboard position, not a capability measurement.

Usually is not a control

The last gap is the one a single average hides most completely. A benchmark reports a mean: out of a thousand tasks, how many succeeded. A regulated control asks something a mean cannot answer: on this task, how often does it fail, and is that rate inside the tolerance I have set?

The agent benchmark that takes this seriously is τ-bench, which evaluates an agent holding a multi-turn conversation under a written policy, in retail and airline domains, and introduces a metric it calls pass^k: not the chance that at least one of k tries succeeds, but the chance that all k independent tries succeed, averaged across tasks. The result is sobering. A model with better than 60% average success on the retail tasks sees its pass^8 fall below 25%: run the same task eight times and the odds that it works on all eight collapse.[17] Its successor extended the idea to a harder support domain and watched one model fall from 74% on retail and 56% on airline to 34% on telecom.[18] METR, measuring how long a task an agent can complete autonomously, reports its central estimates with error bars of roughly a factor of two in each direction, and is explicit that a time horizon is "not the length of time AIs can work independently."[19]

This is the gap that should worry a regulated firm most, because the failure is invisible in the headline. An agent that succeeds 95% of the time looks excellent and, run across a thousand client decisions a day, fails fifty of them, each one a decision

someone has to own. The professional-acceptance question and the reliability question compound, because you do not get to choose which run reaches the client. What you do get to choose is the tolerance, whether the bad runs are detectable when they happen, and what stops them before the client sees them. None of that is available from a mean. A mean tells you the agent usually works. A control needs to know what happens on the runs where it does not, how often those are, and whether you can see them coming. Almost no leaderboard reports it.

How to read a leaderboard if you run agents on regulated data

The structure above turns into a short discipline. None of it requires distrusting benchmarks, only reading them for what they license.

Treat a single score as a screening signal, not a performance guarantee. It is the model under near-ideal, possibly contaminated, conditions on tasks someone else chose, and your workflow may expose failure modes the benchmark never tested. Ask what "correct" meant: a key the builders admit is often wrong, an execution check that says nothing about acceptance, or another model's opinion. Discount any judged-by-LLM number that does not come with multi-trial reliability, because a verdict that varies run to run needs calibration and repeat measurement before it can carry weight. Read saturation as a warning that the benchmark has stopped discriminating among the models you are choosing between, and contamination as a reason last year's number may not survive this year's data. And demand the reliability view, the all-k-succeed shape, not the average, because the average is the number that looks best and tells a regulated firm least.

Most importantly, notice what no public benchmark measures: whether an agent's output is acceptable on your objects, your policies, your definition of done, to the person who has to answer for it. That is not a gap in the benchmarks. It is the thing benchmarks structurally cannot supply, because acceptable is defined locally and the whole point of a public benchmark is to be global. The only evaluation that governs your workflow is one built on your own acceptance criterion and run as a standing control, which is the argument for keeping a private eval as the thing you actually trust and treating the leaderboard as weather, not ground. The benchmark tells you which models are worth testing. It never tells you which one you can be accountable for. That last step does not come from a leaderboard, and a firm that forgets the difference will buy a high score and inherit a liability.

Sources

[1]Tengjun Jin, Yoojin Choi, Yuxuan Zhu and Daniel Kang (University of Illinois), "Text-to-SQL Benchmarks are Broken: An In-Depth Analysis of Annotation Errors," CIDR 2026, extended as "Pervasive Annotation Errors Break Text-to-SQL Benchmarks and Leaderboards" (arXiv:2601.08778). Expert analysis puts annotation error rates at "52.8% and 62.8%" for BIRD Mini-Dev and Spider 2.0-Snow. Re-evaluating all 16 open-source agents from the BIRD leaderboard on a corrected subset, "performance changes range from −7% to 31% (in relative terms) and rank changes range from −9 to +9 positions." As of June 2026.

[2]FrontierMath, Epoch AI (arXiv:2411.04872). At launch (November 2024) the benchmark reported that state-of-the-art models solved "under 2% of problems." Epoch's tiers page states that "on 2026-06-12, we released v2 which addressed errors in 42% of problems." Separately,

on who funds the scorer: OpenAI commissioned and owns the problems and can see the statements and solutions apart from a 50-problem holdout set. Epoch was contractually barred from disclosing this until the o3 announcement in December 2024, and more than sixty contributing mathematicians said they had not been told (Epoch's account). As of June 2026.

[3]Humanity's Last Exam, Center for AI Safety and Scale AI (arXiv:2501.14249). FutureHouse reported that "29 ± 3.7% (95% CI) of the text-only chemistry and biology questions had answers with directly conflicting evidence in peer reviewed literature" and noted that reviewers were not required to verify the full accuracy of a question's rationale if doing so would take more than five minutes (FutureHouse research announcement, 23 July 2025). The HLE team's own follow-up put the problem rate on a Bio/Chem subset around 18%, with at least one of three reviewers disagreeing 25% of the time. The benchmark grades with a model judge (GPT-4o in the paper). As of June 2026.

[4]GPQA, Rein et al. (arXiv:2311.12022). The paper estimates "the proportion of questions that have uncontroversially correct answers at 74% on GPQA Extended," and its validator-error table groups about 28% of disagreements under variants of "the question is bad." Domain-expert accuracy is about 65%. The Diamond subset of 198 questions is the one usually quoted. As of June 2026.

[5]SWE-bench, Jimenez et al. (arXiv:2310.06770), ICLR 2024. The task: "Given a codebase along with a description of an issue to be resolved, a language model is tasked with editing the codebase to address the issue." Scoring is purely execution-based: apply the patch, run the fail-to-pass and pass-to-pass tests, and count the issue resolved if all pass. No model judge is involved. At launch the best model resolved 1.96% of issues. As of June 2026.

[6]SWE-bench Verified, OpenAI with the original authors, August 2024 (announcement). The annotation campaign found unit tests "often overly specific, and in some cases are even unrelated to the issue," issue descriptions "underspecified, leading to ambiguity," and filtered out 68.3% of sampled instances; 93 developers screened the set down to 500 human-validated tasks. OpenAI reported that "GPT-4o's performance on the best-performing scaffold reaches 33.2% on SWE-bench Verified, more than doubling its score of 16% on the original SWE-bench," which it read as validating "our initial suspicion that the original SWE-bench dataset underestimates agent abilities." As of June 2026.

[7]MT-Bench uses a model grader: "An LLM judge is presented with a question and two answers, and tasked to determine which one is better," and a single-answer mode that assigns a score "on a scale of 10" (arXiv:2306.05685). Humanity's Last Exam states "We use GPT-4o as a judge to verify answer correctness." Many agent benchmarks deliberately avoid this in favour of execution or state comparison; the distinction is worth checking per benchmark. As of June 2026.

[8]Wang et al., "Large Language Models are not Fair Evaluators," ACL 2024 (arXiv:2305.17926): "the quality ranking of candidate responses can be easily hacked by simply altering their order of appearance in the context," to the point that "Vicuna-13B could beat ChatGPT on 66 over 80 tested queries with ChatGPT as an evaluator." See also Ye et al., "Justice or Prejudice? Quantifying Biases in LLM-as-a-Judge," ICLR 2025 (arXiv:2410.02736). As of June 2026.

[9]MT-Bench / "Judging LLM-as-a-Judge," Zheng et al. (arXiv:2306.05685), NeurIPS 2023. The authors examine position bias, verbosity bias, and self-enhancement bias, "the effect that LLM judges may favor the answers generated by themselves." They find that "GPT-4 favors itself with a 10% higher win rate; Claude-v1 favors itself with a 25% higher win rate," but stop short of the conclusion: "due to limited data and small differences, our study cannot determine whether the models exhibit a self-enhancement bias." That restraint is worth noting, since the effect is often cited as settled. The same paper reports GPT-4 matching human preferences at "over 80% agreement, the same level of agreement between humans," which is why model judges are used at all. As of June 2026.

[10]Abel Yagubyan, "The Coin Flip Judge? Reliability and Bias in LLM-as-a-Judge Evaluation" (arXiv:2606.13685). Pairwise preferences "flip on average 13.6% of the time across repeated trials, with 28% of questions exceeding a 20% flip rate," with a 72% first-position majority and

only 76% cross-judge agreement; the paper estimates about eleven repeated trials are needed for majority voting to match reference verdicts with 95% probability. A single preprint; figures from the abstract. As of June 2026.

[11]Hao Wang, Qiuyang Mang, Alvin Cheung, Koushik Sen and Dawn Song (UC Berkeley RDI), "How We Broke Top AI Agent Benchmarks: And What Comes Next," April 2026 (rdi.berkeley.edu). An automated exploit agent reached 100% on SWE-bench Verified (500 tasks), SWE-bench Pro (731), Terminal-Bench (89) and FieldWorkArena (890), about 100% on WebArena (812), about 98% on GAIA (165) and 73% on OSWorld (369), solving no tasks and, in most runs, making no LLM call. The proposed remedy is to run a zero-capability adversarial agent against any harness before publishing: a score above zero means the evaluation has a bug. The post does not state which maintainers have since patched. As of June 2026.

[12]LiveCodeBench, Jain et al. (arXiv:2403.07974), ICLR 2025. The benchmark time-stamps problems and scores by execution (pass@1, no model judge). It finds models "perform considerably worse on problems released since" their cutoffs and "a stark drop in the performance of DS-Ins-33B model after Aug. 2023 (right before its release date), which suggests that the earlier problems might indeed be contaminated." Questions are refreshed monthly. As of June 2026.

[13]AIME is the American Invitational Mathematics Examination, a human competition (15 integer-answer questions per paper) repurposed as a reasoning eval, typically scored by exact match. Han Wang, Haoyu Li, Brian Ko and Huan Zhang, "On The Fragility of Benchmark Contamination Detection in Reasoning Models" (arXiv:2510.02386), report the AIME 2024 score rising from 43.33 to 61.67 under supervised-fine-tuning contamination, while "almost all the detection approaches perform near random guesses (i.e., AUROC $\approx$ 50%)." The small sample (about 30 problems for a combined year) also makes scores high-variance. As of June 2026.

[14]Epoch AI's GPQA Diamond tracking records a top score of 87% (Grok 4, July 2025). The 65% baseline is the GPQA paper's own measured accuracy for experts holding or pursuing PhDs in the relevant domain; Epoch and OpenAI have used a re-benchmarked figure near 70%, which is why both circulate. Humanity's Last Exam, built after MMLU saturated, remained unsaturated at the time of writing. No HLE figure is quoted here: the published tops move monthly and are largely vendor-submitted. As of June 2026.

[15]MMLU-Pro, Wang et al. (TIGER-Lab) (arXiv:2406.01574), NeurIPS 2024 Datasets and Benchmarks, introduced as "an enhanced dataset designed to extend the mostly knowledge-driven MMLU benchmark by integrating more challenging, reasoning-focused questions and expanding the choice set from four to ten options." It exists because the original MMLU had saturated near its ceiling. As of June 2026.

[16]Singh et al., "The Leaderboard Illusion" (arXiv:2504.20879), on Chatbot Arena / LMArena. "Google and OpenAI have received an estimated 19.2% and 20.4% of all data on the arena, respectively," while "a combined 83 open-weight models have only received an estimated 29.7% of the total data." The paper documents "27 private LLM variants tested by Meta in the lead-up to the Llama-4 release" and finds that "even limited additional data can result in relative performance gains of up to 112% on the arena distribution." LMArena's own work also found response style (length, formatting) materially affects rankings, prompting a style-control adjustment. As of June 2026.

[17]τ-bench, Yao et al. (Sierra) (arXiv:2406.12045). The agent converses with an LLM-simulated user under a written policy and is scored deterministically by comparing the final database state to a unique ground-truth outcome (no model judge). It introduces pass^k, "the chance that all k i.i.d. task trials are successful, averaged across tasks." A model with better than 60% average success on retail sees pass^8 fall below 25%. As of June 2026.

[18]τ^2-bench, Barres et al. (Sierra) (arXiv:2506.07982), extends τ-bench to a dual-control telecom support domain where the user can also act; "gpt-4.1 pass^1 drops from 74%/56% for retail and airline respectively to 34% for telecom." The framework offers several success criteria, one of which is a natural-language assertion that would require a model judge, but the paper states that "in telecom, only assertion functions are used to evaluate task success." As of June

2026.

[19]METR, "Measuring AI Ability to Complete Long Tasks" (arXiv:2503.14499) and its time-horizon limitations note. The 50%-task-completion time horizon grew from about 50 minutes (Claude 3.7 Sonnet) toward roughly 4 hours 49 minutes (Claude Opus 4.5, with a 95% CI from under 2 hours to over 20), on a doubling of about 7 months. METR cautions that "error bars have historically been a factor of ~2 in each direction" and that the horizon "is not the length of time AIs can work independently." As of June 2026.