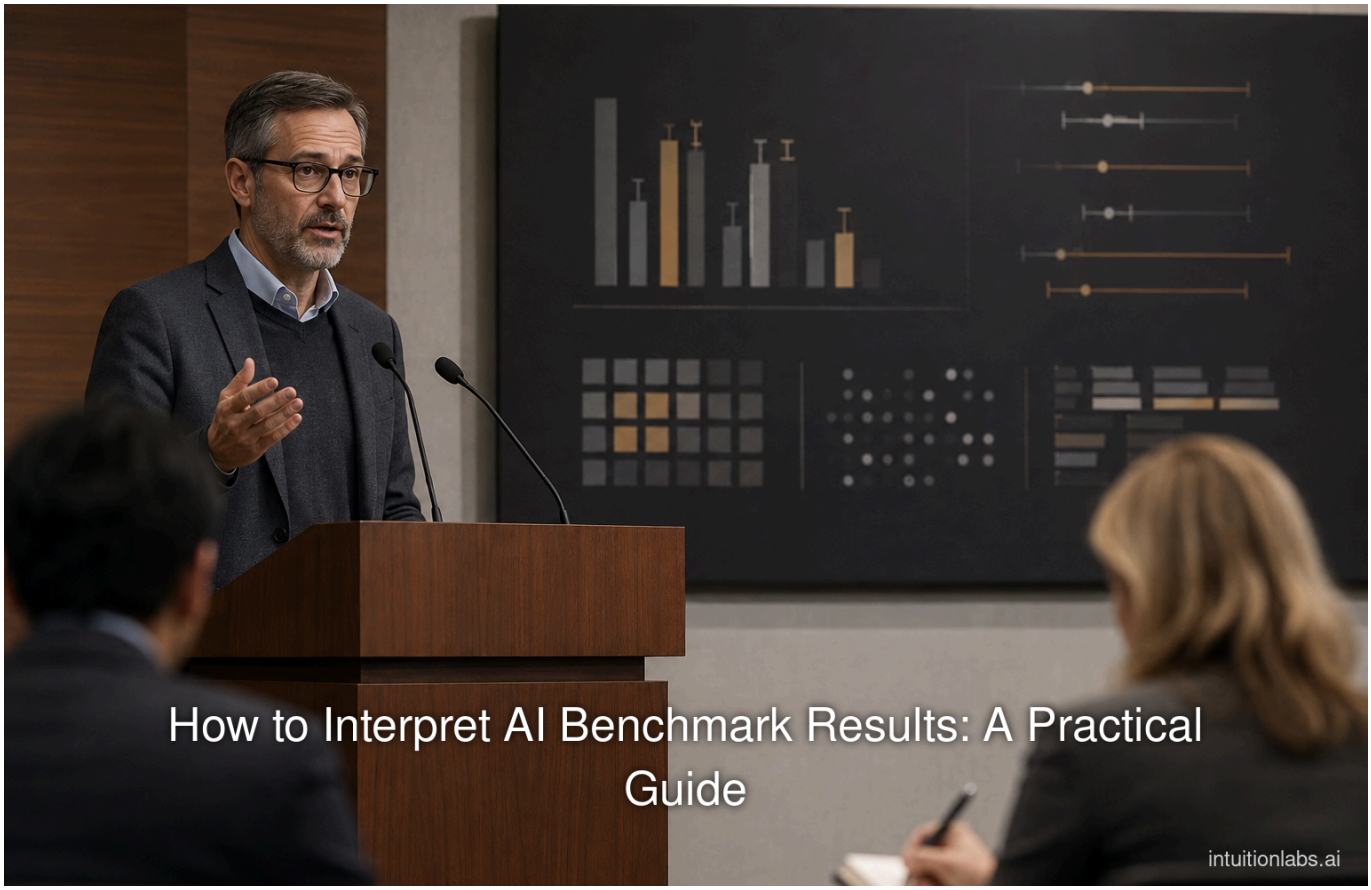
How to Interpret AI Benchmark Results: A Practical Guide

IntuitionLabs.ai · Adrien Laurent · 2026-09-05 · ai benchmarks · llm evaluation · benchmark contamination · data contamination · llm leaderboards · statistical significance · evaluation harness · ai model performance



Executive Summary

Interpreting an AI benchmark result correctly requires separating four things a single headline score conflates: whether test data leaked into training data beforehand, which evaluation harness and prompt format produced the number, how much compute or sampling budget the model was given, and how much statistical noise is present given the benchmark's size. This report, current as of September 2026, documents each confound using primary sources from OpenAI, Anthropic, Google DeepMind, Meta, and the academic and standards-body literature on large language model (LLM) evaluation, and assembles the findings into a reproducible, six-point comparison checklist.

On statistical treatment, the widely cited HumanEval coding benchmark contains only 164 examples, small enough that published analyses show single-run comparisons and multi-run confidence intervals for the same models can lead to very different conclusions about which one leads, as detailed below.

Cost now travels with accuracy on the most demanding benchmarks: on ARC-AGI-2, a top verified commercial model scored 37.6% at $2.20 per task versus 54% at $30 per task for a more expensive refinement-loop configuration (arcprize.org), and MLCommons announced results for its industry-standard MLPerf Inference v6.0 benchmark suite (mlcommons.org), providing an example of a standardized comparison process. IntuitionLabs, a life-sciences and AI consultancy, applies this same diligence when advising regulated organizations on foundation-model selection (intuitionlabs.ai).

Introduction and Background

An AI lab announces a new model with a headline benchmark score. A rival's model beats it by a few points on the same benchmark weeks later. For an enterprise buyer deciding which foundation model to build on, or a journalist reporting the news, that gap raises a harder question than it first appears to: is it real?

Interpreting an AI benchmark result correctly requires distinguishing four things a single headline number conflates: the model's underlying capability, the evaluation harness and prompt format used to measure it, the statistical noise inherent in any finite test, and the possibility that the test questions leaked into training data before the test was run. Ignoring any one can turn a reporting artifact into a false capability claim, in either direction.

Benchmark evaluations involve choices in data, tooling, and reporting that should be disclosed when results are compared.

This report, current as of September 2026, walks through the four confounds in turn: how test-data contamination is detected and quantified in primary AI-lab and academic sources; how evaluation harness, prompt format, and sampling budget change scores independent of capability; how statistical significance and run-to-run variance apply to benchmark comparisons; and how the major public leaderboards are built, and where their own maintainers say the limits are. It closes with a reproducible checklist a reader can apply to any benchmark comparison. **IntuitionLabs**, a life-sciences and AI consultancy that advises pharmaceutical and life-science organizations on AI adoption, tracks this literature because clients evaluating foundation models for regulated, scientific use cases need to know which benchmark claims will hold up in production (intuitionlabs.ai); this article extends the consultancy's earlier survey of 2025 AI research trends (intuitionlabs.ai) by focusing on the method of interpretation rather than the results themselves.

What Interpreting a Benchmark Score Actually Requires

A reported benchmark score is the output of a chain of choices, not a direct readout of capability. At minimum, four layers sit between "the model" and the number a headline reports:

- **The benchmark's construction**: what questions it contains, how they were sourced, and how much of their content already existed on the public web before any model was trained on it.
- **The evaluation harness**: the software that feeds prompts to a model, extracts an answer, and scores it, whether that is EleutherAI's lm-evaluation-harness, Stanford's HELM (Holistic Evaluation of Language Models), a benchmark's own reference implementation, or a lab's unpublished internal tooling.
- **The sampling and compute budget**: whether the model answered once per question (single-pass) or many times with the best or most common answer selected (best-of-n or self-consistency voting), and how much test-time compute or tool access it was allowed.

- **The statistical treatment**: whether the reported number is a single run's raw accuracy or a mean with a confidence interval across multiple runs, and how large the underlying test set is relative to the size of the difference being claimed.

Two models with identical capability can show different scores on any of these axes, and two models with different capability can show the same score if the axes are not controlled. Formal efforts to standardize the chain exist: Stanford's HELM project measures each of 16 core evaluation scenarios against seven separate metrics, including robustness and calibration, not just accuracy, precisely because a single number understates how these choices affect comparability (crfm.stanford.edu).

Vendors and independent evaluators do not always disclose all four layers for a given number. A statement that "Model X scores 88% on GPQA" (Graduate-Level Google-Proof Q&A, a widely used science-reasoning benchmark) without specifying whether tools were allowed, how many samples were taken, or which harness produced the number, is not directly comparable to a rival's "84%" measured differently. The remaining sections work through each layer before assembling the findings into a checklist.

Data Contamination: How Test Data Leaks Into Training Data

Benchmark data contamination is an evaluation-validity problem: if a task, answer, or close variant was available during training, or is available to a browsing system during evaluation, a score can measure memorization or retrieval rather than the capability the benchmark is intended to test (OpenAI). That risk does not make every public benchmark unusable, but it means a score should not be treated as evidence of generalization until the evaluation’s exposure controls are disclosed.

A useful disclosure identifies the benchmark version and release date, the model’s relevant training-data cutoff where it is known, whether benchmark tasks and answers were screened against training material, the matching method and threshold used, and how flagged items affected the reported score. For systems with web access, the evaluation should also state whether browsing or other retrieval tools were available, because public answers can contaminate an otherwise held-out test at evaluation time.

Readers should distinguish a documented control from a blanket assurance. A report can make its contamination check auditable by naming the corpus or access boundary it checked, preserving the benchmark version, and reporting results for any retained or excluded items. Where those details are unavailable, the appropriate conclusion is not that contamination occurred, but that the reader cannot determine how much the score reflects previously available material.

Evaluation Harness, Prompt Format, and Sampling Budget Effects

Even with zero contamination, two evaluations of the same model checkpoint can produce very different scores purely because of how the test was administered. Three mechanisms account for most of the documented variation: prompt formatting, harness implementation, and sampling budget.

Prompt format. A 2024 study measuring format sensitivity found meaning-preserving changes to few-shot prompt formatting, such as separator characters or answer-choice labeling, produced accuracy differences of up to 76 percentage points on the same task with LLaMA-2-13B, and that a format favoring one model does not reliably

favor another, since format performance only weakly correlates between models (Table 2). A related study found GPT-NeoX-20B scored 72.4% on ARC-Easy under a cloze-style prompt but only 26.5% under an MMLU-style prompt, on identical questions (Table 2).

Harness implementation. Evaluation reports should identify the harness and scoring format used to produce a result.

Even MMLU (Massive Multitask Language Understanding) is not immune: whether accuracy is averaged per-question ("micro") or per-subject across its 57 categories ("macro") is a choice that on its own shifts the reported score by several points (arxiv.org). Newer agentic benchmarks show the same dynamic: IEEE Spectrum reported on 2026 benchmarks built by Carnegie Mellon and Fujitsu to test whether AI agents are safe enough for unsupervised business operations, finding leading multimodal models "sometimes hallucinated and struggled with counting objects precisely and measuring specific distances" (spectrum.ieee.org), with such benchmarks needing steady replacement as scores approach "the point of minimal progress" (spectrum.ieee.org).

Sampling budget and tool access. Reported scores also depend on how many attempts a model gets. On SWE-bench Lite, one study found solve rate rose from 15.9% with a single attempt to 56% with 250 sampled attempts and the best kept, surpassing the previously reported single-attempt state of the art of 43% (Table 2). On math word problems, sampling multiple reasoning paths and taking a majority vote ("self-consistency") instead of a single greedy answer improved GSM8K accuracy by 17.9 percentage points on its own (Table 2). Neither change reflects a better model, only more compute spent per question. Before standardized suites existed this inconsistency was systemic: Stanford's HELM project found models were, on average, evaluated on only a small fraction of a common set of core scenarios, with some prominent models not sharing a single scenario in common with each other.

Table 2 below collects the quantified swings documented above by non-capability factor, to show their rough scale relative to typical reported score gaps between competing models.

Factor	Documented Effect	Model / Benchmark	Source
Prompt format	76-point swing from format changes alone	LLaMA-2-13B (arxiv.org)	
Prompt style (cloze vs. multiple-choice)	72.4% vs 26.5% on identical questions	GPT-NeoX-20B on ARC-Easy (arxiv.org)	
Sampling budget (best-of-n)	15.9% to 56% solve rate, 1 to 250 samples	SWE-bench Lite (arxiv.org)	
Self-consistency voting	+17.9 percentage points	GSM8K (arxiv.org)	
Cost / compute budget	37.6% at $2.20/task vs 54% at $30/task	Claude Opus 4.5 vs. a Gemini 3 Pro refinement system, ARC-AGI-2 (arcprize.org)	

These swings are frequently larger than the score gaps headlines describe as one model "beating" another, which is why a source's stated methodology, not only its stated score, determines whether a comparison is meaningful.

Statistical Significance and Run-to-Run Variance

Even a clean, well-harnessed benchmark run is a single draw from a noisy process. Large language model outputs are sampled, not deterministic, in most production configurations, and "large language models are heavily stochastic objects" whose single-run scores can misrepresent performance. One widely cited analysis illustrates this by plotting the same models' GPQA scores twice, once as single-run point estimates and once as a 95% confidence interval from ten repeated runs each: "the two plots lead to very different conclusions" about which model is ahead (arxiv.org).

Test-set size compounds the problem. HumanEval, one of the most widely cited coding benchmarks, contains only 164 examples, small enough that a one- or two-point difference between models can plausibly be resampling noise rather than a real capability gap. This is not unique to HumanEval: a 2020 study of experimental design in natural language processing (NLP) found a machine-translation test set of 2,000 sentences, a fairly typical size, has only about 75% statistical power to detect a genuine 1-point difference on the BLEU translation-quality metric, and defined an underpowered experiment as one with "less than 80% power" (arxiv.org), meaning even a 2,000-example test sits only just above the field's own adequacy threshold at that effect size. Below that threshold, both false negatives and, when a result reaches significance by chance, inflated or sign-reversed effect estimates become more likely.

The field has been aware of this gap for some time. A 2018 methodology paper presented at the Association for Computational Linguistics' annual meeting found "statistical significance testing is often ignored or misused" in published NLP comparisons (aclanthology.org), and later methodological work argued significance testing alone is insufficient, proposing researchers also estimate effect size and conduct power analysis to gauge the risk of missing a real difference because the test was underpowered.

Recommended practice, per a widely cited 2024 paper on reproducible language model evaluation, is to report variance and run multiple seeds rather than a single score: the authors advise researchers to "perform statistical analyses, and report on sources of variance and error," noting that "reporting results run over more than one random seed can dramatically boost the validity and utility of results" (arxiv.org). To lower the barrier to doing this, lm-evaluation-harness reports bootstrapped standard error metrics by default, so adding a confidence interval to a published result requires copying an additional number rather than rerunning the full evaluation design.

For a reader evaluating a benchmark claim, the practical questions are: how many examples make up the benchmark relative to the size of the claimed score gap; was the result reported as a single run or as a mean with a confidence interval across multiple runs; and if only a single run is available, is the claimed gap large relative to the documented run-to-run swings catalogued in Table 2 above.

Leaderboard Design and Known Limitations

Public leaderboards aggregate results across many models and use different methodologies.

Benchmark-specific rankings carry their own construction issues. GPQA was deliberately built to resist casual lookup: its creators report "highly skilled non-expert validators only reach 34% accuracy, despite spending on average over 30 minutes with unrestricted access to the web" on GPQA questions, versus roughly 65% for matched domain PhD holders (arxiv.org), a gap meant to show the benchmark measures expertise rather than search ability.

None of this means the leaderboards are not useful; each disclosure was published by the leaderboard's own maintainers or by researchers improving it, and each led to a documented correction. The practical lesson is that a leaderboard position is only as reliable as its version and date, since the underlying methodology has changed, sometimes substantially, for every major public leaderboard discussed here.

A Reproducible Benchmark-Comparison Checklist

The preceding sections point to a consistent set of questions that separate a defensible benchmark comparison from an unverifiable one. The checklist below condenses them into a sequence a reader can apply to any two benchmark scores being compared, whether in a vendor announcement, a leaderboard snapshot, or a news article.

Table 3 below organizes these checks into six dimensions, the question to ask for each, and the signal that indicates a comparison cannot yet be trusted.

Dimension	Key Question	Red Flag
Contamination disclosure	Did the source state a detection method and a per-benchmark contamination rate?	No contamination statement at all, or a vague claim that checks "were performed"
Harness and version	Was the same evaluation harness and version used for every model being compared?	Scores pulled from different labs' self-reported numbers, each using its own undisclosed harness
Prompt format and scoring	Was prompt format, few-shot count, and answer-extraction method held constant and disclosed?	Cloze-style and multiple-choice results are combined, or micro- and macro-averaged MMLU scores mixed, without noting the difference
Sampling and tool budget	Were all models given the same number of attempts, tools, and compute budget per question?	One model's best-of-n or agentic score is compared against another's single-pass score
Statistical treatment	Is the score a single run or a mean with a confidence interval, and how large is the test set relative to the claimed gap?	A one- or two-point difference is reported on a benchmark of a few hundred examples with no confidence interval
Benchmark quality and freshness	Is the benchmark version current, and has it been independently audited for ground-truth errors or saturation?	The benchmark is near-saturated (most leading models within a few points of the ceiling) or has a documented, uncorrected error rate

No public leaderboard or vendor announcement currently discloses all six dimensions for every reported score as a matter of course. MLCommons' MLPerf Inference suite is a useful standardized reference point: MLCommons says it measures system performance "in an architecture-neutral, representative, and reproducible manner" (mlcommons.org). Readers comparing less standardized sources should expect to fill in several checklist rows themselves, or discount the comparison accordingly.

Data Analysis and Evidence

The gap between reported and verified capability is visible in the numbers major labs themselves publish. OpenAI's **GPT-5** launch announcement in August 2025 reported 74.9% on SWE-bench Verified and, for its "pro" configuration, a new state of the art on GPQA Diamond of 88.4% without external tools (openai.com) (openai.com). Its system card specifies that "all SWE-bench evaluation runs use a fixed subset of n=477 verified tasks," not the full 500-task set, a detail that matters for anyone trying to reproduce the number (cdn.openai.com).

Cost and compute budget now appear alongside accuracy as a standard reporting dimension for the field's most demanding benchmarks. In its 2025 results analysis, the ARC Prize Foundation reported 1,455 teams submitted 15,154 entries to its ARC-AGI-2 reasoning competition, alongside "90 papers submitted, up from 47 last year" (arcprize.org). Within that competition, a Kaggle entrant reached 24% accuracy on the private ARC-AGI-2 test set at a reported cost of $0.20 per task (arcprize.org), while the top verified commercial model, Claude Opus 4.5 in its extended "Thinking" mode, scored 37.6% at $2.20 per task, and a separate refinement-loop system built on Gemini 3 Pro reached 54% at $30 per task (arcprize.org), a roughly fourteen-fold cost difference for a seventeen-point accuracy gain. None of these figures is comparable to the others without its dollar figure attached.

Anthropic's November 2025 release notes for Claude Opus 4.5 apply the same principle to efficiency: the company reports Opus 4.5 "matches Sonnet 4.5's best score on SWE-bench Verified, but uses 76% fewer output tokens," and at its highest effort setting exceeds Sonnet 4.5's score by 4.3 points while using 48% fewer tokens (anthropic.com), a case where accuracy alone would understate the efficiency difference.

Independent measurement organizations reach similar conclusions with different methodologies. Google DeepMind's own model documentation states its Gemini 3.1 Pro model, in an extended-reasoning configuration, scores 94.3% on GPQA Diamond without tools (deepmind.google), while its smaller Gemini 3.8 Flash model reaches 54.9% on the HLE-Verified benchmark (deepmind.google). Epoch AI argues "most benchmarks saturate too quickly to study long-run AI trends" (epoch.ai). METR, an AI evaluation nonprofit, instead measures a "time horizon", the length of task an agent can complete with 50% reliability, finding this has grown with "a doubling time of around 7 months" across the industry, with Claude 3.7 Sonnet having "a time horizon of approximately one hour" by that measure (metr.org) (metr.org). Axios reported in July 2026 that the same dynamic applies outside general-purpose benchmarks, citing Stanford's AI Index that cybersecurity capability "evaluations intended to be challenging for years are saturated in months" (axios.com).

Implications and Future Directions

Governance frameworks have started to formalize this. NIST's AI Risk Management Framework, released January 26, 2023 after a multi-year consultation process (nist.gov), was extended in July 2024 with a Generative AI profile addressing generative-system risks (nist.gov), treating evaluation as an ongoing governance activity rather than a one-time checkbox, consistent with the version-by-version corrections documented throughout this report.

For organizations in regulated or scientific domains, where model outputs may inform clinical, regulatory, or safety-relevant decisions, benchmark literacy has direct operational consequences. A model selected on a single, uncontextualized leaderboard score, without checking whether it reflects the harness, format, and sampling budget intended for production use, risks a mismatch between reported and actual performance in deployment.

IntuitionLabs advises life-science and pharmaceutical organizations navigating this kind of AI adoption decision, and treats the four confounds discussed here, contamination, harness effects, statistical noise, and benchmark quality, as standard diligence items when evaluating a foundation model for a regulated use case (intuitionlabs.ai).

Several sources cited here point toward the same structural fix: benchmarks refreshed on a schedule (as MLCommons does with MLPerf rounds and ARC Prize does annually), evaluation harnesses that report confidence intervals by default (as lm-evaluation-harness already does), and disclosure norms separating a model's raw score from the compute or tool budget spent obtaining it (as ARC Prize's cost-per-task reporting and Anthropic's token-efficiency disclosures both do). None of these fixes is universal yet, which means the checklist in Table 3 will likely remain necessary for readers doing comparisons manually for some time to come.

Frequently Asked Questions (FAQs)

Why do AI benchmark scores vary so much between reports of the same model? Variation comes from at least four sources documented throughout this report: harness and version differences, prompt format and answer-extraction differences, sampling budget or tool access differences, and ordinary statistical noise from a finite test set (see Tables 2 and 3). A reported difference should be checked against all four before it is treated as a capability difference.

How is data contamination detected in large language models? The two dominant method families are n-gram overlap, checking whether a benchmark question shares a long token sequence with the training corpus, and learned or paraphrase-aware detectors, which catch reworded contamination n-gram methods miss. No method is complete; a disclosed methodology should be read alongside its reported contamination rate.

How reliable are public LLM leaderboards? Reliability depends on the leaderboard and its version. A score should be read together with its leaderboard's version and date.

What is statistical significance in the context of an AI benchmark, and why does it matter? It is a formal test of whether an observed score difference between two systems is unlikely to have arisen from chance, given test-set size and score variance. Many published comparisons report only a single run's point estimate without this test, which is why researchers recommend multi-run confidence intervals instead.

Can a benchmark comparison ever be fully trusted without rerunning it? Only to the extent the source discloses enough of the six checklist dimensions in Table 3; MLCommons says its MLPerf Inference suite measures system performance "in an architecture-neutral, representative, and reproducible manner" (mlcommons.org), but even MLPerf results should be read alongside their exact submission round and hardware configuration.

Conclusion

A benchmark score is not a fact about a model; it is a fact about a model measured under specific, and frequently underdisclosed, conditions. The evaluation considerations discussed here show why contamination, harness choice, prompt format, sampling budget, and statistical noise should be considered when comparing benchmark scores.

None of this means benchmark scores are meaningless. It means they carry an implicit unit of measurement, the harness, format, sampling budget, and test-set size behind them, that must travel with the number for a comparison to be valid, much as a price is meaningless without a currency and date. The checklist in Table 3 offers a practical way to check whether that unit of measurement has been disclosed for any two scores being compared.

This report prioritizes evaluation context over a single leaderboard snapshot.
