On Evaluating Methods vs. Evaluating Models

Olawale Salaudeen[1†] **Florian Dorner**[2,3] **Peter Hase**[4†]
[1]MIT [2] Max Planck Institute for Intelligent Systems, Tübingen
[3] ETH Zürich [4] Stanford University
Correspondence to olawale@mit.edu

Abstract

We distinguish between two often-conflated uses of datasets: evaluating methods, which compare algorithms under controlled conditions, and evaluating models, which assess the capabilities of a fixed model. Method evaluation emphasizes relative rankings and is robust to model-independent distortions, while model evaluation requires valid absolute scores and can be biased by flaws like contamination or task mismatch. Using simple formulations and synthetic experiments, we demonstrate how this conflation can reverse rankings and misrepresent model capabilities, leading to misleading leaderboards and flawed conclusions about a model's true capabilities. We conclude with recommendations for designing and interpreting state-of-the-art evaluations, grounded in the critical distinction between methods and models.

1 Introduction

A central ambiguity runs through machine learning evaluation. When a system is tested on a dataset, are we *evaluating a method or a model*? This question lies at the core of how benchmarks are used and interpreted. Consider ImageNet [1]. Evaluating the ResNet architecture [2] on ImageNet primarily demonstrates the effectiveness of the *method* (architecture), not necessarily the specific trained model, which is limited in scope. ResNets remain useful across a variety of learning tasks and domains [3]. Evaluating a specific ResNet model trained on ImageNet instead tests the fixed image classification model itself, whose use for new domains, e.g., for medical imaging diagnosis via transfer learning, is unknown from just the ImageNet scores [4]. The first case evaluates an algorithmic idea, while the second evaluates a learned model. Conflating the two interpretations of evaluation leads to confusion about what evaluation results actually mean.

Public benchmarks have driven progress in AI [5–8]. They can be used as *instruments for comparing methods* or as *measures of model capability*. However, failing to separate these uses has led to different interpretations of the utility of benchmarks. Those focused on technical advances are bullish, viewing them as engines of reproducible progress [6, 7, 5], while those focused on predicting deployment performance are bearish, seeing them as poor indicators of real-world capability due to the mismatch between what benchmarks measure and real-world deployment needs [9], reliability [10], and domain fit [11]. We argue that large-language-model (LLM) evaluation particularly suffers from this lack of clarity about what exactly such benchmarks are meant to measure.

We distinguish between evaluating methods and evaluating models (Figure 1). In *method evaluation*, the primary goal is to compare algorithms by their relative rankings, where the target is the method that produced the evaluated model. The model and dataset together form a measurement instrument of the method's efficacy. Success means outperforming alternatives under controlled conditions. When scores shift under benign distortions, like uniform noise, the ordering is preserved,

[†]Work done while AI Institute Fellow in Residence at the Schmidt Sciences.

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: Evaluating the Evolving LLM Lifecycle: Benchmarks, Emergent Abilities, and Scaling.

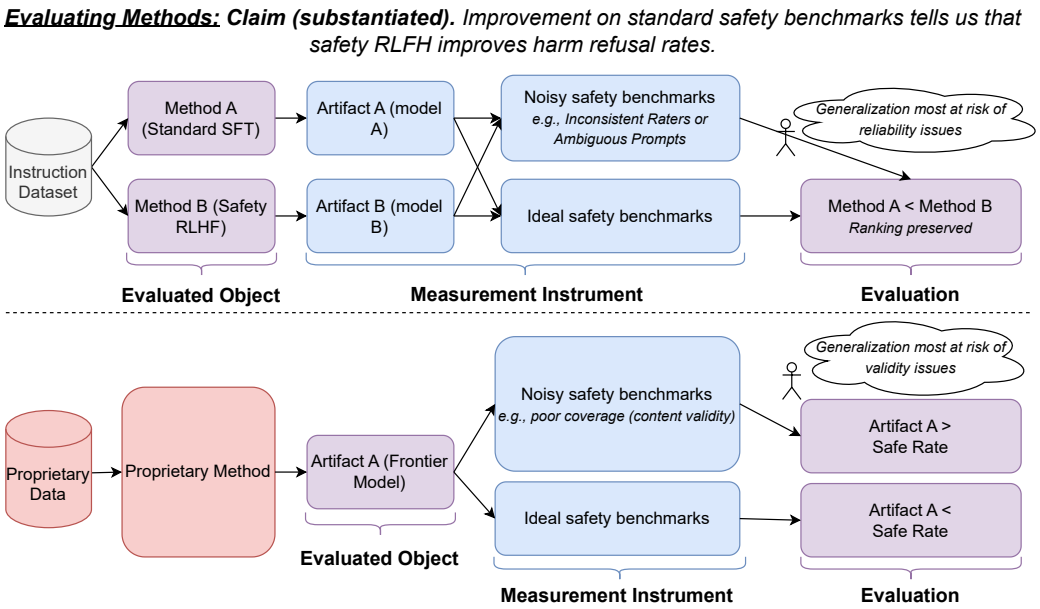
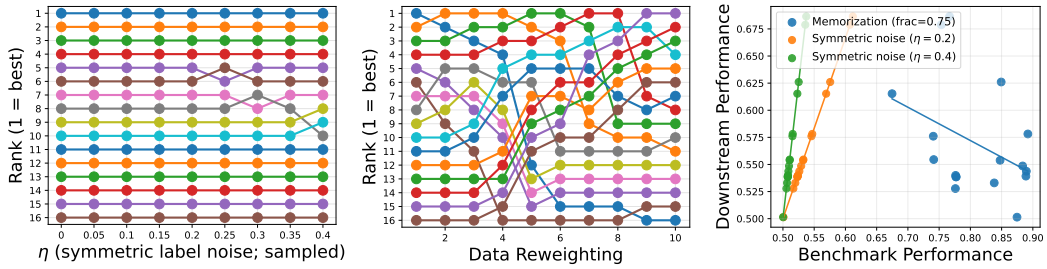


Figure 1: Top: Method evaluation relies on relative evidence. Even when the measurement instrument is noisy (e.g., inconsistent raters), the ranking between methods (Safety RLHF > Standard SFT) remains robust and reliable. Bottom: Model evaluation relies on capability evidence (absolute scores). Here, the same measurement flaws (e.g., lack of content validity or static benchmarks) break the link between the score and the construct, leading to invalid safety claims where high scores do not guarantee real-world safety.

much like a race where the winner remains the same even if the stopwatch is slightly off. In *model evaluation*, the target is the fixed artifact itself, i.e., the model and system it is embedded in. The absolute score matters because it informs decisions such as safety audits or deployment approvals. Continuing the analogy, to qualify for the Olympics, it does not matter if you beat the runner next to you; what matters is whether your absolute time meets the strict standard required to compete. We are no longer asking "who is faster on this track," but "exactly how fast is this runner, so we can compare to a standard," e.g., to certify that a model's safety score meets the minimum 'standard' to deploy.

Measurement vs. Evaluation. The same measurement process can yield identical scores, but what we aim to evaluate, and whether the measurement is valid for that purpose, matters most. Measurement records an outcome such as accuracy or win rate. A measurement instrument is the standardized procedure (data, tasks, and scoring rules) used to produce a measurement. Evaluation interprets that outcome relative to a goal. For methods, interpretation is straightforward because rankings are defined in relation to other algorithms under the same conditions. For models, interpretation is harder since the score must correspond to the intended construct, the underlying concept or property that a score is intended to measure, and domain of use. As a result, measurements can be precise yet misleading when they fail to capture the property of interest.

Model Rankings. Rankings of fixed models on a shared benchmark can serve a practical role [12, 13]. They help with triage, support comparison among alternatives, and provide a coarse marketplace signal when training pipelines are opaque or irreproducible [14]. However, the scores that produce these rankings are not necessarily valid measures of the broader capabilities or risks often attached to them [15]. For example, GPQA [16] scores are sometimes read as evidence of graduate-level reasoning ability, but may only reflect relative success on the benchmark's specific question set. A model with a high rank may only show relative performance under the stated task mix and scoring rule, but may not establish readiness for a particular downstream deployment context or task. Without valid scores, other tasks associated with the same assumed capabilities may also not have the same rankings [17]. Even the best model, with scores higher than human experts, may fail on tasks that one



(a) (b) (c)

Figure 2: We evaluate a set of algorithms, e.g., SVMs, nearest-neighbors, and random forests, on a binary classification problem to build intuition. Figure 2a shows that method comparisons are robust under uniform label noise (rankings preserved in expectation). Figure 2b shows that changing task mix or using model-dependent protocols can alter method rankings. Figure 2c shows that absolute scores used to evaluate models are misaligned under noise and contamination. Details in Appendix A.

would reasonably expect from a system with graduate-level reasoning, such as correctly counting the number of 'r', in 'strawberry'. Model rankings aid triage and selection, but they depend on the task mix and scoring; a model ranking higher might not be meaningful if the scores are invalid evaluations for the intended use case.

Dual roles of datasets. The same dataset can serve both as a testbed for method comparison and as a measurement instrument of model properties; the generalizability of claims about the latter is often more narrow. The LLM era has made these dual roles more explicit. Evaluation now focuses less on training procedures and more on the properties of released artifacts [18, 19]. As training pipelines become opaque, benchmarks are increasingly interpreted as signals of commercial status rather than controlled scientific evidence. Yet, leaderboards persist and blur the line between evaluating methods and evaluating models [20, 12, 13].

This distinction is not always clear-cut, but it allows AI developers to separate their evaluation pipelines. *Method rankings* (relative evidence) are best suited for internal research and iterative development, where controlled comparisons guide engineering choices. *Model evaluation* (capability evidence) is necessary for external validation and deployment, where decisions depend on absolute validity rather than relative rank. Confusing these purposes leads to overstated capability and underestimated risk.

Even the best general-purpose model on state-of-the-art benchmarks may be inadequate for many real-world applications [21]. Evaluations that inform deployment must establish *validity*, meaning that a score measures the intended construct [15]. Establishing validity often requires designs beyond static general-purpose tests, such as uplift studies [22], randomized controlled trials [23], or longitudinal field evaluations [24]. We conjecture that general-purpose benchmarks remain essential for scientific progress but are insufficient for assessing use-case–specific readiness in practice.

Contributions.

- We formalize the distinction between evaluating methods and evaluating models by making the target of claim explicit.
- We derive intuitive results that characterize when rankings are stable or unstable and when capability claims drawn from those rankings are reliable or misleading, supported by simple synthetic experiments.
- We offer practical guidance for LLM evaluation, including uncertainty reporting and sensitivity checks, to improve the interpretation and use of benchmark results.

2 Analysis

A key insight is that *evaluating methods* can be robust to model-independent distortions in a dataset or metric, such as uniform noise [25, 26]. When any strictly monotonic function remaps scores,

orderings among methods often persist. This preserves methodological progress, i.e., ranking of methods, but it does not preserve claims about the model artifact's capabilities based on scores.

2.1 Evaluating methods.

Here, the **claim is about the relative efficacy of methods.** When considering methods, the notion of a score of efficacy is irregular in practice. In this section, when we evaluate a model, we are evaluating the method that generated this model. We first explain why simple metric changes need not alter method ranks.

Proposition 1 (Ranking stability under monotone remaps)**.** *Let $s(h)$ be the score of model h generated by a method of interest, and let g be strictly increasing. Define $\tilde{s}(h) = g(s(h))$. For any h_A, h_B,*

$$s(h_A) > s(h_B) \iff \tilde{s}(h_A) > \tilde{s}(h_B).$$

For examples, such transformations include switching perplexity to $\log$ perplexity in language modeling or rescaling human ratings [27]. These transformations preserve ranks even though numerical gaps change. For method comparison, this property underscores reliability, since the goal is often to make claims about relative efficacy. For evaluating models, such changes to scores can obscure the signal one needs for decision making, e.g., when scaling changes the units of the score and blurs interpretation (Figure 2). Another example of ranking preserving distortion is model-independent label noise, such as flips at rate η across K classes, where accuracy can contract toward a constant that depends on η while preserving order.

Proposition 2 (Score contraction under symmetric label noise)**.** *Let $K \geq 2$ and $\eta \in [0, \frac{K-1}{K})$ be the symmetric label-flip rate. For fixed predictions of h, where $Acc(\cdot)$ gives expected accuracy*

$$\mathrm{Acc}_\eta(h) = \left(1 - \eta - \tfrac{\eta}{K-1}\right)\mathrm{Acc}(h) + \tfrac{\eta}{K-1}.$$

Concrete sources of such noise include inconsistent crowd-sourced data or flaky unit tests in code benchmarks that randomly pass or fail [28]. In these cases, observations often resemble Figure 2a: rankings are stable while absolute scores shrink. In expectation, rankings remain perfectly conserved.

The stability demonstrated under model-independent perturbations, such as symmetric noise and metric remaps, reinforces the scientific utility of method evaluation through public benchmarks. It confirms that observed rank improvements are primarily attributable to the intrinsic methodological advance rather than external measurement artifacts.

However, not all perturbations are benign; particularly, model-dependent perturbations can affect method rankings. For instance, selecting items in a benchmark to emphasize specific types of examples or tailoring the mixture of a benchmark suite to each model artifact can alter the rankings. For instance, rankings can be reversed when examples are adversarially selected [29].

Proposition 3 (Rank instability under data reweighting)**.** *Let $s_D(h) = \mathbb{E}_{(x,y)\sim D}[L(h(x), y)]$. There exist D, a reweighting D', and models h_A, h_B such that $s_D(h_A) < s_D(h_B)$ but $s_{D'}(h_A) > s_{D'}(h_B)$.*

Importantly, this also adversely for evaluating models. Real-world examples of data reweightings are common. Reweighting can arise when a benchmark shifts emphasis, for instance, from math to coding, from low-resource to high-resource languages, or from longer to shorter contexts [30]. Other model-dependent factors include hand-tuned prompts, jailbreaks [31], red-team attacks [32], and interactive tasks. Such dependence often produces ranking instability, as shown in Figure 2b.

2.2 Evaluating models.

Suppose we now fix a model M that we would like to use elsewhere and ask whether its performance is adequate for an intended use. Absolute scores become central to quantify a property via a score and make **claims about specific capabilities from the score** [33]. Validity concerns dominate here because any systematic mismatch between the test and the concept of interest biases the claim [15, 34].

Proposition 4 (Bias in model evaluation)**.** *Let the evaluation distribution be a mixture of valid data D_{clean} and flawed data D_{flawed} with fraction ϵ. If $U(M) = \mathbb{E}\left[\mathrm{Perf}(M, D_{clean})\right]$ and $U_{flawed}(M) = \mathbb{E}\left[\mathrm{Perf}(M, D_{flawed})\right]$, where Perf is a performance score, then*

$$\widehat{U}(M) = (1-\epsilon)U(M) + \epsilon U_{flawed}(M) = U(M) + \epsilon\big(U_{flawed}(M) - U(M)\big).$$

Having invalid data can bias interpretations of scores. Examples include test contamination, sample or task, from pre-training corpora or finetuning [35], near-duplicate examples across training and testing [36], exact-match grading that rewards template copying [37], and reference errors in automatic judges [38]. Figure 2c shows that both noise and contamination can spuriously favor certain models.

Reliability vs. Validity. Method benchmarking aligns more with *reliability*, asking whether evaluations yield stable relative orderings under benign perturbations. Model evaluation aligns more with *validity*, asking whether scores capture the intended concept and predict downstream utility. Reliability is necessary but not sufficient for validity; a stable leaderboard shows reliability, not the validity of the score. Rank stability primarily diagnoses reliability, whereas bias, contamination, and concept misspecification threaten validity [15, 39]. Both are required.

A defensible capability claim must specify the concept and target domain, justify the task mix and scoring rule, report uncertainty, and include checks for concept-irrelevant variance, such as noise, contamination, and mixture tests. Reliability helps but does not guarantee validity.

Methods on fixed models. Evaluating methods like prompting, decoding, or steering on a fixed model is common. Here, the target of claims remains the method, and rankings are important to identify the best method. However, ranking reliability hinges on which base is used since model-specific quirks can masquerade as method gains. Replicating rankings across diverse models and reporting the variation is essential.

This distinction clarifies the role of prompting. When the goal is method evaluation (e.g., 'Does Chain-of-Thought improve reasoning?'), the prompt strategy is the claim target, and relative gains across models are the signal. When the goal is Model Evaluation (e.g., 'Is GPT-4 safe?'), prompt sensitivity acts as measurement error. Consequently, model benchmarks that allow per-model prompt engineering (e.g., 'system prompt optimization') inadvertently switch the evaluation mode from assessing the model's artifacts to evaluating the developer's prompting method, introducing the rank instability predicted by Proposition 3.

Our stance is simple: **compare methods to guide development and evaluate models to make guide real-world utility decision-making.**

2.3 Experiments.

We present a simplifying intuition of phenomena already observed in LLM evaluations rather than reproducing them in Section 3. We ground our discussion in a small, reproducible synthetic testbed, Figure 2. We consider a simple classification task; we train a set of classifiers and evaluate them on a set of tasks. Under noise applied to evaluation datasets, scores contract toward a constant (Figure 2c) while method order stays fixed, as predicted (Figure 2a)—notably, the rankings in this result are not perfect as a consequence of real-world finite sample effects. In expectation, the rankings are maintained exactly. Reweighting the test-task mix, data contamination, i.e., test set included in training, and model-dependent changes leads to ranking instability (Figure 2b).

Although for illustrative purposes, these parameters map directly to current evaluation challenges. Symmetric label noise (η) proxies for inconsistent crowdsourced annotations or flaky code execution environments, where results stochastically change independent of model capability. Our results show that while such measurement noise causes score contraction—systematically understating absolute capabilities, it acts as a benign distortion that largely preserves method rankings. In contrast, data reweighting shifts composition of leaderboards (e.g., prioritizing coding over chat); as predicted by Proposition 3, this guarantees rank instability whenever models exhibit non-identical capability profiles.

3 Related Work

Evaluations in machine learning display a wide spectrum of reliability. In computer vision, new ImageNet test sets reduce overall accuracy yet preserve model rankings [40], implying that measurement noise and minor distributional drift do not always overturn comparative orderings. Follow-up studies [41, 25] further confirmed that even under counterfactual contrasting datasets on ImageNet, test sets yield highly correlated results with the original ImageNet tests. This suggests that method-level

progress, improvements shared across architectures in this case, are generalizable even when absolute scores from model artifacts change. This pattern is consistent with theoretical and empirical analyses showing that rankings can be robust to model-independent perturbations, such as symmetric label noise or strictly monotone remappings of the metric [26].

Early work showing 'accuracy on the line', where across related distributions, model performance tends to fall along approximately linear trends [42, 43], suggests that this rank stability might hold for models too, even though the slope of that relationship depends on the training data and the kinds of shifts encountered [44]. However, follow-up work shows that this stability of model (rather than method) rankings only holds when the two tests are very similar; otherwise, the correlation can be near-zero or negative [11]. Furthermore, when dataset construction alters the weighting of latent subgroups or the mechanisms underlying the tasks, model ranking stability deteriorates [29], revealing how benchmark composition can yield misleading conclusions, i.e., aggregation can give misleading claims about relative capabilities.

These empirical findings motivate a distinction between the robustness of comparisons when considering methods and the validity of scores when considering models. Rank stability indicates reliability in assessing methodological progress, what might be called "relative claim." In contrast, validity, the extent to which a score measures the intended capability for a fixed model, is a precondition for making capability claims about a fixed system. The persistence of rank correlations across ImageNet-style tests therefore says more about reliable method benchmarking than about the capability generalizability of individual models.

Now consider the case of *model rankings* in large-language-model (LLM) evaluations. Multitask benchmark design faces an inherent trade-off between diversity and stability: broader task mixtures capture a wider range of capabilities but make aggregate rankings noisier and more sensitive to weighting choices [17]. Additionally, contamination and targeted exposure to benchmark tasks confound comparisons and can spuriously inflate scores [35]. Efforts to standardize data and equalize fine-tuning, e.g., train-before-test, *harmonize* rankings across benchmarks [45], yet this synchronization may primarily reflect shared training distributions and often changes absolute scores, so it does not by itself establish capability construct validity. Consequently, *model rankings* face validity challenges. Without rigorous standardization, they cannot be read as method rankings. Even when model rankings are stable, this indicates measurement *reliability*, not necessarily construct *validity*; internally consistent scores can still miss intended properties such as reasoning, safety, or fairness [33, 15, 34].

Taken together, this body of work points toward an emerging "science of evaluation" [8, 46]. Studies in vision illustrate when and why rankings remain stable under controlled perturbations; work on LLMs exposes how dataset contamination, aggregation choices, and social interpretation shape what evaluations measure. Our contribution complements these trends by making the target of claim explicit: clarifying when an evaluation supports comparative claims about methods versus capability claims about models. By formalizing this distinction, we connect empirical patterns of rank stability and validity breakdowns to a coherent measurement framework.

4 Discussion

Our framework clarifies the tension in large-scale evaluation platforms such as HELM [12] and Chatbot Arena [20]. These platforms provide rankings that, externally, have limited information as method rankings due to a lack of experimental control and transparency in the methods, nor is there sufficient validation for what concepts their scores capture. We argue that this evaluation mode is inherently ambiguous.

Consider Chatbot Arena. While rankings may align with model performance on user query distributions, the validity of these scores for measuring specific concepts requires further validation. Specifically, Elo ratings illustrate the risk of interpreting method-style rankings as model evaluations. As relative measures of win-probability, they fulfill the 'comparative evidence' criteria of method evaluation and are robust to global shifts in judge strictness (Proposition 1). However, because they are relative, they do not inherently isolate intrinsic capability from the composition of the opponent pool. A model's Elo describes its relation to specific peers, not necessarily its mastery of a domain; thus, treating Elo as a proxy for absolute capability relies on the unverified assumption that the relative ordering perfectly maps to the intended real-world construct.

This confusion between evaluating methods and models complicates benchmark interpretation, especially when assessing real-world utility. Comparative rankings from method benchmarking should not be taken as evidence of real-world safety and reliability—assertions that require more substantial evidence through proper model evaluation. This claim gap explains the discrepancy between benchmark performance and real-world outcomes. Mistaking high benchmark performance for real-world readiness can lead to unsafe system deployment, overestimated scientific progress, and inflated hype cycles. Distinguishing between "method validation" and "capability assessment" is therefore critical for safe deployment, accurate progress tracking, and responsible communication.

One downstream implication of our discussion is that estimating **uncertainty** must take different forms when evaluating methods vs. models. If the goal is to rank order methods, researchers should: (1) quantify uncertainty over individual method scores, then (2) quantify uncertainty over possible rank orderings [47]. This generalizes uncertainty estimation of model "win-rates" in head-to-head comparisons used to claim superiority in chat settings [48]. The output of such an analysis, for a single model, is the probability that the model occupies each possible rank, from first to last of the models being evaluated. In contrast, evaluating models for absolute capabilities requires uncertainty quantification only over target domain performance, since performance in the target domain is what we care about (not the benchmark score). However, quantifying domain shift effects on confidence intervals remains an open problem [49, 50]. Consequently, translating the confidence intervals on the benchmark score (which are not even regularly reported on leaderboards) to confidence intervals in the target domain is a difficult and rarely undertaken task.

Beyond Benchmarks for Model Evaluation. Benchmarks alone provide limited evidence for real-world utility. A high score on a shared dataset may reflect benchmark-specific artifacts rather than capability for deployment. Model evaluation requires designs beyond static tests, such as uplift studies [22], prospective studies, or longitudinal field evaluations [24]. These approaches directly measure whether a model's traits and behavior in either a carefully controlled environment or in the intended use case environment. While more costly, such evaluations are essential to establish valid measurements that support the intended claim, since dataset performance by itself rarely justifies readiness for high-stakes use.

Limitation. This conceptual separation is sometimes not so black and white in practice. However, we argue that keeping this distinction when evaluating or interpreting evaluations is critical for reliable evaluations. Method evaluation may also consider relative differences—e.g., how much better is this algorithm than another, and whether this relative comparison generalizes [25], but we leave this aspect for future work.

5 Conclusion

The confusion between evaluating methods and evaluating models is not merely semantic but represents a fundamental ambiguity in the target of claim. As we have shown, method rankings are often robust to measurement noise, whereas model capability claims can collapse under the slightest noise or distribution change. By formalizing this distinction, we move beyond treating benchmarks as universal yardsticks and toward a mature science of evaluation where relative improvements in algorithms are clearly separated from absolute claims about the safety and utility of model artifacts.

References

[1] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[3] Zhuoren Zhang. Resnet-based model for autonomous vehicles trajectory prediction. In *2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE)*, pages 565–568. IEEE, 2021.

[4] Yuning Huang, Jingchen Zou, Lanxi Meng, Xin Yue, Qing Zhao, Jianqiang Li, Changwei Song, Gabriel Jimenez, Shaowu Li, and Guanghui Fu. Comparative analysis of imagenet pre-trained deep learning models and dinov2 in medical imaging classification. In *2024 IEEE 48th Annual Computers, Software, and Applications Conference (COMPSAC)*, pages 297–305. IEEE, 2024.

[5] Moritz Hardt and Benjamin Recht. *Patterns, predictions, and actions: Foundations of machine learning*. Princeton University Press, 2022.

[6] David Donoho. Data science at the singularity. *arXiv preprint arXiv:2310.00865*, 2023.

[7] Benjamin Recht. The mechanics of frictionless reproducibility. *Harvard Data Science Review*, 6(1), 2024.

[8] Moritz Hardt. The emerging science of machine learning benchmarks. *Manuscript. https://mlbenchmarks. org*, 2025.

[9] Richard Ren, Steven Basart, Adam Khoja, Alice Gatti, Long Phan, Xuwang Yin, Mantas Mazeika, Alexander Pan, Gabriel Mukobi, Ryan Kim, et al. Safetywashing: Do ai safety benchmarks actually measure safety progress? *Advances in Neural Information Processing Systems*, 37:68559–68594, 2024.

[10] Joshua Vendrow, Edward Vendrow, Sara Beery, and Aleksander Madry. Do large language model benchmarks test reliability? *arXiv preprint arXiv:2502.03461*, 2025.

[11] Olawale Salaudeen, Nicole Chiou, Shiny Weng, and Sanmi Koyejo. Are domain generalization benchmarks with accuracy on the line misspecified? *arXiv preprint arXiv:2504.00186*, 2025.

[12] Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*, 2022.

[13] Open llm leaderboard. `https://huggingface.co/open-llm-leaderboard`. Accessed 2025-08-29.

[14] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[15] Olawale Salaudeen, Anka Reuel, Ahmed Ahmed, Suhana Bedi, Zachary Robertson, Sudharsan Sundar, Ben Domingue, Angelina Wang, and Sanmi Koyejo. Measurement to meaning: A validity-centered framework for ai evaluation. *arXiv preprint arXiv:2505.10573*, 2025.

[16] David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. Gpqa: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*, 2024.

[17] Guanhua Zhang and Moritz Hardt. Inherent trade-offs between diversity and stability in multi-task benchmarks. *arXiv preprint arXiv:2405.01719*, 2024.

[18] OpenAI. Gpt-4 technical report. *arXiv:2303.08774*, 2023. URL `https://arxiv.org/abs/2303.08774`.

[19] Rishi Bommasani, Kevin Klyman, Shayne Longpre, Sayash Kapoor, Nestor Maslej, Betty Xiong, Daniel Zhang, and Percy Liang. The foundation model transparency index. *arXiv preprint arXiv:2310.12941*, 2023.

[20] Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Banghua Zhu, Hao Zhang, Michael Jordan, Joseph E Gonzalez, et al. Chatbot arena: An open platform for evaluating llms by human preference. In *Forty-first International Conference on Machine Learning*, 2024.

[21] Abinitha Gourabathina, Yuexing Hao, Walter Gerych, and Marzyeh Ghassemi. The medperturb dataset: What non-content perturbations reveal about human and clinical llm decision making. *arXiv preprint arXiv:2506.17163*, 2025.

[22] Meredith Somers and MIT Sloan. How generative ai can boost highly skilled workers' productivity. *Ideas That Matter*, 2023.

[23] Ethan Goh, Robert Gallo, Jason Hom, Eric Strong, Yingjie Weng, Hannah Kerman, Joséphine A Cool, Zahir Kanjee, Andrew S Parsons, Neera Ahuja, et al. Large language model influence on diagnostic reasoning: a randomized clinical trial. *JAMA network open*, 7(10):e2440969–e2440969, 2024.

[24] Tao Long, Sitong Wang, Émilie Fabre, Tony Wang, Anup Sathya, Jason Wu, Savvas Petridis, Dingzeyu Li, Tuhin Chakrabarty, Yue Jiang, et al. Facilitating longitudinal interaction studies of ai systems. *arXiv preprint arXiv:2508.10252*, 2025.

[25] Olawale Salaudeen and Moritz Hardt. Imagenot: A contrast with imagenet preserves model rankings. *arXiv preprint arXiv:2404.02112*, 2024.

[26] Florian E Dorner and Moritz Hardt. Don't label twice: Quantity beats quality when comparing binary classifiers on a budget. *arXiv preprint arXiv:2402.02249*, 2024.

[27] Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. Rankme: Reliable human ratings for natural language generation. *arXiv preprint arXiv:1803.05928*, 2018.

[28] Edoardo Manino, Long Tran-Thanh, and Nicholas Jennings. On the efficiency of data collection for crowdsourced classification. 2018.

[29] Olawale Salaudeen, Haoran Zhang, Kumail Alhamoud, Sara Beery, and Marzyeh Ghassemi. Aggregation hides out-of-distribution generalization failures from spurious correlations. In *Advances in Neural Information Processing Systems (NeurIPS) 2025*, 2025.

[30] Mostafa Dehghani, Yi Tay, Alexey A Gritsenko, Zhe Zhao, Neil Houlsby, Fernando Diaz, Donald Metzler, and Oriol Vinyals. The benchmark lottery. *arXiv preprint arXiv:2107.07002*, 2021.

[31] Sibo Yi, Yule Liu, Zhen Sun, Tianshuo Cong, Xinlei He, Jiaxing Song, Ke Xu, and Qi Li. Jailbreak attacks and defenses against large language models: A survey. *arXiv preprint arXiv:2407.04295*, 2024.

[32] Ethan Perez, Saffron Huang, Francis Song, Trevor Cai, Roman Ring, John Aslanides, Amelia Glaese, Nat McAleese, and Geoffrey Irving. Red teaming language models with language models. *arXiv preprint arXiv:2202.03286*, 2022.

[33] Ahmed Alaa, Thomas Hartvigsen, Niloufar Golchini, Shiladitya Dutta, Frances Dean, Inioluwa Deborah Raji, and Travis Zack. Medical large language model benchmarks should prioritize construct validity. *arXiv preprint arXiv:2503.10694*, 2025.

[34] Hanna Wallach, Meera Desai, A Feder Cooper, Angelina Wang, Chad Atalla, Solon Barocas, Su Lin Blodgett, Alexandra Chouldechova, Emily Corvi, P Alex Dow, et al. Position: Evaluating generative ai systems is a social science measurement challenge. *arXiv preprint arXiv:2502.00561*, 2025.

[35] Ricardo Dominguez-Olmedo, Florian E Dorner, and Moritz Hardt. Training on the test task confounds evaluation and emergence. *arXiv preprint arXiv:2407.07890*, 2024.

[36] Kushal Tirumala, Daniel Simig, Armen Aghajanyan, and Ari Morcos. D4: Improving llm pretraining via document de-duplication and diversification. *Advances in Neural Information Processing Systems*, 36:53983–53995, 2023.

[37] Moritz Willig, Matej Zečević, Devendra Singh Dhami, and Kristian Kersting. Probing for correlations of causal facts: Large language models and causality. 2023.

[38] Florian E Dorner, Vivian Y Nastl, and Moritz Hardt. Limits to scalable evaluation at the frontier: Llm as judge won't beat twice the data. *arXiv preprint arXiv:2410.13341*, 2024.

[39] Tom Sühr, Florian E Dorner, Olawale Salaudeen, Augustin Kelava, and Samira Samadi. Stop evaluating ai with human tests, develop principled, ai-specific tests instead. *arXiv preprint arXiv:2507.23009*, 2025.

[40] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International conference on machine learning*, pages 5389–5400. PMLR, 2019.

[41] Simon Kornblith, Jonathon Shlens, and Quoc V Le. Do better imagenet models transfer better? In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2661–2671, 2019.

[42] John P Miller, Rohan Taori, Aditi Raghunathan, Shiori Sagawa, Pang Wei Koh, Vaishaal Shankar, Percy Liang, Yair Carmon, and Ludwig Schmidt. Accuracy on the line: on the strong correlation between out-of-distribution and in-distribution generalization. In *International conference on machine learning*, pages 7721–7735. PMLR, 2021.

[43] Christina Baek, Yiding Jiang, Aditi Raghunathan, and J Zico Kolter. Agreement-on-the-line: Predicting the performance of neural networks under distribution shift. *Advances in Neural Information Processing Systems*, 35:19274–19289, 2022.

[44] Zhouxing Shi, Nicholas Carlini, Ananth Balashankar, Ludwig Schmidt, Cho-Jui Hsieh, Alex Beutel, and Yao Qin. Effective robustness against natural distribution shifts for models with different training data. *Advances in Neural Information Processing Systems*, 36:73543–73558, 2023.

[45] Guanhua Zhang, Ricardo Dominguez-Olmedo, and Moritz Hardt. Train-before-test harmonizes language model rankings. *arXiv preprint arXiv:2507.05195*, 2025.

[46] Laura Weidinger, Inioluwa Deborah Raji, Hanna Wallach, Margaret Mitchell, Angelina Wang, Olawale Salaudeen, Rishi Bommasani, Deep Ganguli, Sanmi Koyejo, and William Isaac. Toward an evaluation science for generative ai systems. *arXiv preprint arXiv:2503.05336*, 2025.

[47] Adrien Foucart, Arthur Elskens, and Christine Decaestecker. Ranking the scores of algorithms with confidence. In *ESANN 2025*, 2025.

[48] Roland Daynauth, Christopher Clarke, Krisztian Flautner, Lingjia Tang, and Jason Mars. Ranking unraveled: Recipes for llm rankings in head-to-head ai combat. *arXiv preprint arXiv:2411.14483*, 2024.

[49] Yunye Gong, Xiao Lin, Yi Yao, Thomas G Dietterich, Ajay Divakaran, and Melinda Gervasio. Confidence calibration for domain generalization under covariate shift. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8958–8967, 2021.

[50] Zeju Li, Konstantinos Kamnitsas, Mobarakol Islam, Chen Chen, and Ben Glocker. Estimating model performance under domain shifts with class-specific confidence scores. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 693–703. Springer, 2022.

A Experimental setup and additional results

We use a small synthetic testbed to illustrate how method comparisons and model assessments can diverge. All random seeds and hyperparameters are fixed for reproducibility.

Data. Binary classification with $d{=}40$ features and two subpopulations $z \in \{0, 1\}$. For $z{=}0$ (linear), the first two coordinates are drawn from a correlated Gaussian with means $\pm(\mu, \mu)$ and covariance $\begin{pmatrix} \sigma^2 & \rho\sigma^2 \\ \rho\sigma^2 & \sigma^2 \end{pmatrix}$; the remaining 38 coordinates are $\mathcal{N}(0, 1)$. Labels follow the blob assignment with independent flips at rate 0.12. For $z{=}1$ (XOR), we place four Gaussian blobs at $(\pm\mu, \pm\mu)$ with XOR labels, add a small shift to the first two coordinates, and flip labels at rate 0.22. We inject a label–correlated nuisance in feature $j{=}3$ that *helps* $z{=}0$ and *hurts* $z{=}1$. To control where method rankings cross under mixture reweighting, we add a linear *leak* in feature $j{=}2$ for $z{=}1$ only; its magnitude is tuned by bisection so that a linear model and a shallow tree cross near the right edge of the mixture ($p^\star \approx 0.95$). We use a fixed $40/60$ train/test split.

Evaluation-time shift. On the *test* set we invert the spurious correlation in feature $j{=}3$, add heavy-tailed $\mathrm{t}_{\nu=3}$ noise (scale 0.9) to the first six coordinates, and apply feature dropout (each entry zeroed with probability 0.10). These shifts make the task meaningfully nontrivial for margin-based methods.

Models. We compare off-the-shelf classifiers with fixed hyperparameters: logistic regression ($C \in \{0.2, 0.6, 2.0\}$), linear SVM, RBF SVM ("scale"), RBF SVM with $\gamma \in \{0.2, 2.0\}$, decision trees (depth 1–3), random forests (50 trees / depth 6; 200 trees / depth 10), gradient boosting, and kNN ($k \in \{1, 5, 15\}$). We use `StandardScaler` inside pipelines when appropriate. For panel (c) we additionally include 1-NN as a canonical memorizer.

Scoring and ranking. Primary metric is accuracy. At each condition we compute a lexicographic rank: (i) higher accuracy is better; (ii) ties are broken by lower *clean* negative log-likelihood; (iii) remaining ties by model name. Predictions and probabilities on the (clean) test inputs are cached once and then reused across conditions.

Figure 1a: symmetric label noise. At evaluation time we *only* flip test labels independently with rate $\eta \in \{0, 0.05, \ldots, 0.40\}$ while holding predictions fixed. For each η we average over 25 independent flip draws per model. As predicted by the fixed-predictor analysis, scores contract linearly in expectation (for $K{=}2$ classes: $\mathrm{Acc}_\eta(h) = (1-2\eta)\,\mathrm{Acc}_0(h) + \eta$) and the expected ordering is preserved for $\eta < 1/2$. The bump plot reports ranks vs. η.

Figure 1b: mixture reweighting. Let a_0 and a_1 be a model's accuracies on $z{=}0$ and $z{=}1$. For mixture proportion p, the weighted score is $W(p) = (1-p)a_0 + pa_1$; we sweep p on a uniform grid and rank models at each p (same tie-breaker). With the tuned leak, a linear method and a depth-2 tree cross near $p^\star \approx 0.95$, yielding a visible rank flip.

Figure 1c: contamination vs. symmetric noise. We contrast two model-assessment perturbations using a scatter of *downstream* performance (clean y on test inputs; vertical axis) vs. *benchmark* performance under a perturbation (horizontal axis). (i) *Contamination*: replace a fraction $\phi{=}0.75$ of test pairs (x, y) with randomly drawn training pairs (x', y'). (ii) *Symmetric noise*: flip test labels at $\eta \in \{0.2, 0.4\}$ while keeping inputs and predictions fixed. We fit a separate least-squares line for each perturbation series. Contamination disproportionately benefits memorization (e.g., 1-NN) and can inflate apparent benchmark performance without reflecting true downstream utility, whereas symmetric noise contracts scores in a model-independent way and largely preserves relative order.

B Proofs

Conventions. WLOG, we treat s as a *score* (larger is better). If you use a loss, replace s by $-f$. Proposition 2 assumes evaluation-time label flips with predictions held fixed.

B.1 Proposition 1 Ordinal stability under monotone remaps

Proof. If g is strictly increasing, then for any real a, b we have $a > b$ iff $g(a) > g(b)$. Taking $a = s(h_A)$ and $b = s(h_B)$ gives

$$s(h_A) > s(h_B) \quad \Longleftrightarrow \quad g(s(h_A)) > g(s(h_B))\,,$$

so the ranking is preserved under $\tilde{s}(h) = g(s(h))$. □

Proposition 2 (Accuracy under symmetric label noise, in expectation). Let $(X, Y) \sim \mathcal{D}$ be a K-class problem, $K \geq 2$, and let $h : \mathcal{X} \to \{1, \ldots, K\}$ be a fixed predictor (no adaptation to noise). For $\eta \in [0, \frac{K-1}{K})$, define the symmetric-noise channel N_η that, independently of (X, Y) and h, replaces Y by $\widetilde{Y} \sim \mathsf{N}_\eta(Y)$ where

$$\widetilde{Y} = \begin{cases} Y & \text{w.p. } 1-\eta, \\ \text{a label drawn uniformly from } \{1, \ldots, K\} \setminus \{Y\} & \text{w.p. } \eta. \end{cases}$$

Define the accuracy under noise by

$$\mathrm{Acc}_\eta(h) \;:=\; \mathbb{E}_{(X,Y)\sim\mathcal{D}}\, \mathbb{E}_{\widetilde{Y}\sim\mathsf{N}_\eta(Y)} \big[\, \mathbf{1}\{h(X) = \widetilde{Y}\} \,\big].$$

Then

$$\mathrm{Acc}_\eta(h) \;=\; \Big(1-\eta-\tfrac{\eta}{K-1}\Big)\,\mathrm{Acc}_0(h) \;+\; \tfrac{\eta}{K-1},$$

where $\mathrm{Acc}_0(h) := \mathbb{E}_{(X,Y)\sim\mathcal{D}}[\mathbf{1}\{h(X) = Y\}]$. In particular, for $K = 2$, $\mathrm{Acc}_\eta(h) = (1 - 2\eta)\,\mathrm{Acc}_0(h) + \eta$. Hence for $0 \le \eta < \frac{K-1}{K}$ the map $a \mapsto \Big(1-\eta-\frac{\eta}{K-1}\Big)a + \frac{\eta}{K-1}$ is strictly increasing, so the *expected* ranking of predictors by accuracy is preserved; at $\eta = \frac{K-1}{K}$ it collapses to the constant $1/K$.

Proof. Let $A = \{h(X) = Y\}$. Conditioning on whether the noise flips the label,

$$\begin{aligned}\mathrm{Acc}_\eta(h) &= \mathbb{E}\Big[(1-\eta)\,\mathbf{1}\{h(X) = Y\} \;+\; \eta\,\mathbf{1}\{h(X) = \widetilde{Y}\}\Big]\\ &= (1-\eta)\,\mathbb{P}(A) \;+\; \eta\,\mathbb{E}\Big[\mathbf{1}\{h(X) = \widetilde{Y}\}\Big].\end{aligned}$$

If A holds, then $h(X) = Y$ and under a flip $\widetilde{Y} \neq Y$, so $\mathbb{P}(h(X) = \widetilde{Y} \mid A, \text{flip}) = 0$. If A^c holds, then $h(X) \neq Y$ and, under a flip, $\widetilde{Y}$ is uniform over the $K-1$ labels not equal to Y, so $\mathbb{P}(h(X) = \widetilde{Y} \mid A^c, \text{flip}) = \frac{1}{K-1}$. Thus

$$\mathrm{Acc}_\eta(h) = (1-\eta)\,\mathbb{P}(A) + \eta\,\Big(0\cdot\mathbb{P}(A) + \tfrac{1}{K-1}\mathbb{P}(A^c)\Big) = (1-\eta-\tfrac{\eta}{K-1})\,\mathrm{Acc}_0(h) + \tfrac{\eta}{K-1}.$$

B.2 Proposition 3 Rank reversals under data reweighting

Proof. Let $D_p = (1-p)D_0 + pD_1$ and let $s_D(h) = \mathbb{E}_{(x,y)\sim D}[\phi(h(x), y)]$ be a bounded score. Choose h_A, h_B such that $s_{D_0}(h_A) > s_{D_0}(h_B)$ but $s_{D_1}(h_A) < s_{D_1}(h_B)$. Then

$$s_{D_p}(h) = (1-p)\,s_{D_0}(h) + p\,s_{D_1}(h),$$

so the difference $\Delta(p) = s_{D_p}(h_A) - s_{D_p}(h_B) = (1-p)\Delta_0 + p\Delta_1$ with $\Delta_0 > 0$ and $\Delta_1 < 0$ crosses zero at a unique $p^\star \in (0,1)$. Thus, the order flips between any $p_1 < p^\star$ and $p_2 > p^\star$. □

B.3 Proposition 4 Bias in model assessment

Proof. Let (X, Y) be drawn from $(1-\epsilon)D_{\text{clean}} + \epsilon D_{\text{flawed}}$. By total expectation,

$$\widehat{U}(M) = (1-\epsilon)\,\mathbb{E}_{D_{\text{clean}}}[\mathrm{Perf}(M)] + \epsilon\,\mathbb{E}_{D_{\text{flawed}}}[\mathrm{Perf}(M)] = (1-\epsilon)U(M) + \epsilon U_{\text{flawed}}(M).$$

Equivalently, $\widehat{U}(M) = U(M) + \epsilon\big(U_{\text{flawed}}(M) - U(M)\big)$, which exhibits the bias term. □

Notes. The noise result is for evaluation-time flips with fixed predictions; if training or predictions adapt to the noise, ranks may change. The reweighting flip requires only mixture linearity and bounded scores. The model-dependent construction keeps scores bounded and uses the same (x, y), yet dependence on the evaluated model alone suffices to break ordinal stability.