

DualEval: Joint Model-Item Calibration for Unified LLM Evaluation

Aaron J. Li^{1*}, Hao Huang¹, Youngmin Park¹, Yitong Ma¹,
Wei-Lin Chiang², Li Chen², Cho-Jui Hsieh^{2,3}, Bin Yu¹, Ion Stoica^{1,2}

¹University of California, Berkeley

²Arena Intelligence Inc

³University of California, Los Angeles

Abstract

Current LLM evaluation relies on two complementary but often disconnected signals: static benchmarks with objective correctness labels and arena-style preference data that better reflect open-ended user interactions. We introduce DUALEVAL, a latent model-item calibration framework that represents models and evaluation items in a shared space, jointly estimating model ability together with item difficulty and sharpness. We apply DUALEVAL across four domains: coding, math, miscellaneous domain-knowledge tasks, and generic everyday user queries. Our evaluation uses 18 frontier LLMs, static benchmark labels, and reward-model scores validated against held-out human preferences for open-ended model responses. Empirically, our framework produces reliable and balanced model rankings, while pre-calibrated item profiles support efficient evaluation of previously unseen models and residual-based screening for anomalous responses. Overall, DUALEVAL unifies static and arena-style evaluation through joint model-item calibration, producing model rankings and item-level diagnostics that support more sample-efficient, interpretable, and auditable evaluation pipelines.

1 Introduction

LLM evaluation increasingly draws on two complementary signals: static benchmark correctness and human preferences in open-ended arenas. Static benchmarks provide ground-truth labels and standardized grading protocols (Hendrycks et al., 2020; Chen et al., 2021; Austin et al., 2021), but can become saturated or contaminated as models improve (Ott et al., 2022; Magar and Schwartz, 2022). Arena-style evaluations, such as Chatbot Arena (Chiang et al., 2024), better reflect open-ended user interactions through Bradley-Terry-style rankings

(Bradley and Terry, 1952), but open-ended preference judgments can involve annotator disagreement, subjectivity, and systematic evaluator biases, making them difficult to interpret without additional structure (Van Der Lee et al., 2019; Basile et al., 2021; Zheng et al., 2023). Recent work has begun to connect benchmark and preference-based evaluation (Li et al., 2024; Ni et al., 2024), but the two settings still provide weakly integrated views of model capability.

Furthermore, both settings are largely model-centric: they aggregate item-level outcomes into overall model scores, treating all items as equally informative. This misses a key insight: ranking signal depends on model-item interaction. Items that are nearly always solved or nearly always failed provide limited separation among the current model pool; the most informative items are those for which small ability differences among evaluated models translate into detectable differences in correctness or preference outcomes. Modeling this item structure can make evaluation more stable, interpretable, and cost-efficient, while also identifying which items are saturated, discriminative, or future-facing.

Toward this end, we propose DUALEVAL¹, a unified evaluation framework that jointly estimates model abilities and item properties from both static benchmark correctness and arena-style preference signals. Inspired by Item Response Theory (IRT) (Ackerman et al., 2003; Lord and Novick, 2008), DUALEVAL places model abilities and item difficulties on a shared latent scale. For static benchmarks, we use a two-parameter logistic IRT model over binary correctness labels. For arena-style data, we use reward-model scores to construct soft pairwise preference targets and fit them through the same latent ability-difficulty structure. This joint formulation allows correctness and preference su-

*Corresponding author: aaronjli@berkeley.edu

¹Code: <https://github.com/aaron-jx-li/DualEval>

pervision to update shared model and item parameters, yielding not only model rankings but also item-level diagnostic information.

Our framework has two key benefits. First, it bridges static and arena evaluation: correctness labels and preference signals can update the same latent model parameters, allowing each source to compensate for the other’s weaknesses. Static benchmarks provide clean, controlled supervision, while arena prompts provide open-ended signals closer to realistic user interactions. Second, DUALEVAL turns evaluation data itself into an object of analysis. The learned item parameters characterize each question’s difficulty and sharpness, identify saturated or low-signal items, and support low-cost subsets that preserve the full-data ranking.

We apply DUALEVAL to four domains: coding, math, miscellaneous domain-knowledge tasks, and generic everyday user queries. The first three domains combine static benchmark labels with reward-model scores for open-ended arena responses, while the generic domain uses arena data only. Our experiments cover 18 frontier LLMs from multiple providers, with primary arena supervision from our proprietary scalar reward model, developed internally at LM Arena and validated against held-out human preferences, and robustness checks using the public Skywork-Reward-V2-Qwen3-8B model. Across domains, DUALEVAL reconstructs both evaluation signals, achieving 88–92% static-label accuracy in the three static-anchored domains and 68–81% decisive-pair agreement on arena comparisons. Compared with static-only and arena-only baselines, the joint model yields more balanced rankings across evaluation sources and remains robust under reward-model substitution. Its learned item profiles further support diagnostic applications: selecting the top 10% of items by Fisher information recovers the full-data ranking with Spearman $\rho \geq 0.95$ across all static-anchored domains. More importantly, in leave-one-model-out evaluation, a frozen item bank calibrated without the target model places an unseen model within 2.11 ranks on average using only 10% of the items, outperforming simple static-arena fusion even when the latter uses the full item set. DualEval is also substantially less sensitive than fusion baselines to ability-correlated missingness. Finally, residual-based anomaly screening remains effective under out-of-pool calibration and under less-targeted, response-level contamination.

Overall, we reframe LLM evaluation as joint model-item calibration, and our contributions can be summarized as follows:

- We propose DUALEVAL, a novel evaluation paradigm that unifies static benchmark correctness and open-ended preference signals on a shared latent model-item scale, enabling mutually informed frontier model evaluation and item diagnostics.
- We evaluate DUALEVAL on 18 frontier LLMs across coding, math, miscellaneous domain knowledge, and generic everyday user queries, showing that it produces balanced rankings across signal sources, remains robust to reward-model choice, and is less sensitive than simple fusion baselines to structured missingness.
- We demonstrate two diagnostic applications: Fisher-information-based benchmark compression, which supports sample-efficient placement of previously unseen models using a pre-calibrated item bank, and residual-based anomaly detection, which flags suspicious model–item outcomes under out-of-pool calibration and less-targeted, response-level contamination.

2 Related Work

Static and open-ended LLM evaluation. Static benchmarks cover NLI and reasoning (Bowman et al., 2015; Williams et al., 2018; Wang et al., 2018), QA (Rajpurkar et al., 2016; Yang et al., 2018; Kwiatkowski et al., 2019; Liang et al., 2022), and math (Hendrycks et al., 2020; Cobbe et al., 2021; Hendrycks et al., 2021), but vary substantially in difficulty and discriminability (Zhou et al., 2026; Castleman et al., 2025; Wang et al., 2026). Open-ended preference evaluations such as Chatbot Arena (Chiang et al., 2024) and MT-Bench (Zheng et al., 2023) capture user-facing quality through aggregate Bradley-Terry rankings without explicit item modeling. Several efforts integrate the two settings (Li et al., 2024; Ni et al., 2024).

IRT and efficient evaluation. Item Response Theory (Embretson and Reise, 2013) has long been used in educational testing and was applied early to NLP for instance-level analysis (Lalor et al., 2019; Martínez-Plumed et al., 2019). Recent work uses IRT-style modeling for benchmark diagnostics

(Zhou et al., 2026; Castleman et al., 2025), label-free evaluation (Robertson, 2025), data-efficient evaluation (Liao et al., 2025; Wang et al., 2025), and adaptive testing (Li et al., 2025). Closest to our compression application, tinyBenchmarks (Polo et al., 2024) and metabench (Kipnis et al., 2025) use IRT to build small high-information subsets, and Fluid Benchmarking (Hofmann et al., 2025) adapts item selection to the model under test; Rodriguez et al. (2021) and Vania et al. (2021) use IRT to compare test sets and quantify item informativeness. Prior work therefore establishes the value of IRT for item selection, test-set comparison, and efficient evaluation, including evaluation of held-out models. DUALEVAL differs by coupling static correctness with reward-distilled open-ended preferences in a shared latent model-item space, and by using the resulting item bank for both out-of-pool model placement and residual-based diagnostics.

3 Method

In our DUALEVAL framework, each model i is assigned an ability parameter $\theta_i \in \mathbb{R}$, and each task q is assigned a difficulty parameter $b_q \in \mathbb{R}$ and a sharpness parameter $a_q > 0$ (the discrimination parameter in standard IRT terminology), parameterized as $a_q = \exp(k_q)$ with $k_q \in \mathbb{R}$. Intuitively, $\theta_i - b_q$ measures how capable model i is relative to task q , while a_q is a sharpness parameter that controls how rapidly the predicted success probability changes as model ability moves relative to item difficulty.

Static IRT. For static benchmarks with binary correctness labels, we use a standard two-parameter logistic IRT model:

$$p_{i,q} = P(y_{i,q} = 1) = \sigma(a_q(\theta_i - b_q)),$$

where $\sigma(\cdot)$ is the sigmoid function. Let $\mathcal{S}$ denote the set of observed static labels. We define the static loss as:

$$\mathcal{L}_{\text{static}} = \mathbb{E}_{\mathcal{S}} \text{BCE}(p_{i,q}, y_{i,q}).$$

Reward-Distilled Pairwise IRT. For open-ended evaluations, a reward model assigns each model response a scalar score $r_{i,q}$. Our framework accepts any scalar response-quality scorer; the joint formulation provides robustness to RM noise when static correctness labels are available (Appendix §F). We first standardize rewards globally by $z_{i,q} = \frac{r_{i,q} - \bar{r}}{\sigma_r}$, where $\bar{r}$ and σ_r are the mean

and standard deviation over all reward observations. For each question q and model pair (i, j) , we define a soft pairwise preference target:

$$p_{ijq}^* = \sigma(z_{i,q} - z_{j,q}),$$

which converts reward gaps into calibrated preference strengths: small gaps yield targets near 0.5, while large gaps yield more confident preferences. We retain the pairwise form because it is the natural interface to genuine human preference data: replacing the soft target p_{ijq}^* with a binary win/loss label leaves every other term in the objective unchanged. The tie and both-bad filters mirror the cases a human annotator would mark as a tie or as “both bad”. Reward-model scores are the supervision available at our scale, but the formulation is not specific to them.

Given the shared IRT parameters, we define the latent success probability of model i on question q as:

$$p_{i,q} = \sigma(a_q(\theta_i - b_q)).$$

The predicted preference probability is then:

$$\hat{P}(i \succ j | q) = \sigma(\gamma(p_{i,q} - p_{j,q})),$$

where $\gamma > 0$ is a learned arena-temperature parameter.

Tie and Both-Bad Filtering. Not all reward-derived pairs provide useful relative preference signal. We identify ties using the empirical reward-gap distribution: a pair is marked as a tie if $|z_{i,q} - z_{j,q}| < \delta_{\text{tie}}$, where δ_{tie} is chosen as the configured percentile of pairwise absolute reward gaps. Ties are excluded from the arena loss.

We also identify “both-bad” pairs, where both models receive low standardized rewards with $\max(z_{i,q}, z_{j,q}) < \tau_{\text{bb}}$, where τ_{bb} is chosen as the configured percentile of pairwise maximum rewards. Let $\mathcal{A}_{\text{dec}}$ denote the decisive arena pairs, defined as non-tie and non-both-bad pairs, and let $\mathcal{A}_{\text{bb}}$ denote the both-bad, non-tie pairs. We define the full arena loss as:

$$\begin{aligned} \mathcal{L}_{\text{arena}} = & \lambda_{\text{pref}} \mathbb{E}_{\mathcal{A}_{\text{dec}}} \text{BCE}(\hat{P}(i \succ j | q), p_{ijq}^*) \\ & - \lambda_{\text{bb}} \mathbb{E}_{\mathcal{A}_{\text{bb}}} [\log(1 - p_{i,q}) + \log(1 - p_{j,q})]. \end{aligned}$$

The first term fits relative preference evidence from decisive pairs. The second term treats both-bad pairs as absolute failure evidence, pushing both models toward low success probability on that question and avoiding conflicting gradients between “both models failed” and “one model was slightly better”.

Joint Objective. The static and arena components share the same latent parameters (θ, b, k) . The full optimization objective is:

$$\mathcal{L} = \lambda_{\text{static}}\mathcal{L}_{\text{static}} + \mathcal{L}_{\text{arena}} + \lambda_{\text{reg}} (\|\theta\|_2^2 + \|b\|_2^2 + \|k\|_2^2 + (\log \tilde{\gamma})^2).$$

Let $\gamma_0 = 4$ denote the default arena-temperature scale. We regularize the relative scale $\tilde{\gamma} = \gamma/\gamma_0$ toward one.

For identifiability, after each optimization step we center the model abilities to have zero mean and absorb the same shift into the item difficulties. The relative arena temperature $\tilde{\gamma}$ is regularized separately toward one. All compared fits use the same regularization configuration. Final model rankings are obtained by sorting the learned abilities θ_i .

4 Experiments

In this section, we first validate DUALEVAL’s joint fit and ranking behavior (§4.1), then study two practical applications enabled by the learned item profiles: sample-efficient benchmark compression (§4.2) and residual-based anomaly detection (§4.3). Throughout, we distinguish *in-pool* evaluation, in which all 18 models participate in calibration, from *out-of-pool* evaluation, in which the item bank is calibrated *without* the model that is subsequently placed or screened – the realistic situation a leaderboard faces when a new submission arrives.

Experimental Setup. We evaluate DUALEVAL on four settings: coding, math, miscellaneous domain-knowledge tasks, and generic everyday user queries. The first three combine challenging static benchmarks with reward-scored open-ended responses; the generic setting uses arena-style data only, serving as an arena-only stress test without static anchors. We select evaluation-relevant arena prompts using LLM judges, with filtering details in Appendix §H. Static benchmarks provide binary correctness labels, while open-ended responses are scored by our proprietary scalar RM, a Qwen-3-32B backbone with a reward head trained on over five million human-preference pairs under a Bradley–Terry objective. To test robustness to reward-model choice, we repeat the main static-anchored analyses with the public Skywork-Reward-V2-Qwen3-8B RM, which agrees less well with held-out human preferences; both RMs are validated against human preferences in Appendix §C. For each model-question

pair, we generate one response under a consistent inference setup (Appendix §H). Table 1 lists the benchmark composition; the 18 evaluated models span eight providers and are listed in Appendix §I. For agentic coding tasks that require a fixed execution harness, we use the same harness for all models: Terminus2 for TerminalBench (Merrill et al., 2026) and mini-SWE-agent v2 for SWE-Bench (Jimenez et al., 2024).

Protocols and Baselines. Two calibration protocols recur below. **In-pool** estimates item parameters from all 18 models and then ranks or screens those same models, measuring structure within a fixed pool. **Out-of-pool** holds out each model in turn, calibrates item parameters – and, where relevant, Fisher scores and the clean reference – on the remaining 17, and places or screens the held-out model against that frozen bank, so no responses from the held-out model are used to calibrate the bank; results are averaged over the leave-one-model-out folds unless noted. The two applications differ in how the held-out model’s ability is obtained: in §4.2 it is estimated from selected cells in both modalities, whereas in §4.3 it is estimated only from the opposite, unscreened modality, so that contaminated cells cannot influence placement. For the structured-missingness analysis and the out-of-pool placement experiment, we additionally use two parameter-free fusion baselines that observe the same cells as DUALEVAL but model no item structure. **Mean-z fusion** averages a model’s z -scored static accuracy and z -scored mean arena reward, while **rank-avg fusion** averages its two per-signal ranks. These methods provide ranking baselines but cannot themselves identify informative items or construct model-item residuals because they learn no item parameters. All experiments use the proprietary RM unless stated otherwise.

4.1 DUALEVAL Results

Figures 1, 3, 4, and 5 show the learned model abilities and item-difficulty distributions across the four settings, with math, miscellaneous, and generic visualizations provided in Appendix §A.

Static and Arena Fit. Under the cell-level 80/20 holdout protocol, averaged over three random seeds, DUALEVAL achieves static-label accuracies of 90.3%, 91.8%, and 88.4% on coding, math, and miscellaneous, respectively. On held-out arena comparisons, it achieves decisive-pair agreement of 77.8%, 68.2%, 78.1%, and 80.9% on coding, math,

Evaluation data composition — four domains, each question answered by all 18 evaluated models.

Domain	Static benchmarks (#Q each)	Arena #Q	Total
Coding	LiveCodeBench v6 (120), MBPP-Plus (80), SWE-Bench-lite (60), TerminalBench-2.0 (40)	300	600
Math	AIME 2025 (30), AIME 2026 (30), HLE-Math (60), Olympiad-Math (80)	200	400
Misc	HLE: Bio.+Med. (52), Eng. (25), Hum.+Soc.+Sci. (90), Other (83); SimpleQA (50)	300	600
Generic	—	500	500

Table 1: Evaluation data composition. Static benchmarks give binary correctness labels; arena prompts are filtered from LMArena by LLM judges, and the generic domain uses arena prompts only. All 18 models answer every question, one response per (model, question) pair. Models: Appendix §I. Sources and sampling: Appendix §H.

miscellaneous, and generic, respectively. The arena agreement is comparable to the proprietary reward model’s agreement with held-out human preferences, suggesting that the joint latent structure captures a substantial share of the preference signal rather than merely fitting noise. Reliability diagrams in Appendix §E show approximate calibration for both static success probabilities and arena preference probabilities, with ECE 0.04–0.07 on static cells and 0.01–0.03 on arena soft pairwise targets, suggesting no systematic biases of over- or under-confidence across domains.

Difficulty Structure. The learned item profiles are strongly heterogeneous, with many items far from the main ability cluster – saturated, or harder than the evaluated pool. The two modalities also differ: static items show near-zero correlation between difficulty b_q and log-sharpness $\log a_q$, whereas arena items show a positive association ($\rho \approx 0.6$; Appendix §J). Difficulty and sharpness therefore carry different information across sources, motivating a Fisher-information score (§4.2) that combines both rather than selecting on either alone.

Static–Arena Integration. We next test whether joint calibration integrates the two signals without collapsing to either. Per static-anchored setting we compare against a static-only 2PL IRT model, an arena-only Bradley–Terry model on reward-derived pairs, and an arena-only average-reward ranking (Appendix §G), averaging over three random question splits and scoring each method by held-out arena decisive-pair accuracy and by Spearman correlation with the full-data static-only IRT and Arena BT references.

Table 2 shows the expected source-specialization tradeoff: the static-only model aligns best with the static reference but transfers weakly to the arena one, arena-only baselines do the reverse, and DUALEVAL preserves nearly the static-only model’s static-reference agreement while improv-

ing arena-reference agreement over it – an integration mechanism rather than a single-source optimizer. The references are drawn from this same table, so the $\rho = 1.000$ entries are self-comparisons, not performance.

Static anchoring also stabilizes the fit when arena supervision is noisier. Under the public Skywork RM, Arena BT’s agreement with the public sub-leaderboards falls to $\rho = 0.51/0.54/0.25$ for coding/math/misc, whereas public-RM DUALEVAL still matches the static reference at $\rho = 0.976/0.984/0.963$ and stays closer to the proprietary-RM DUALEVAL ranking than arena-only training does (cross-RM $\rho = 0.963/0.992/0.835$; Appendix §F). The arena-only generic domain is the no-anchor counterfactual: without static labels the two RMs diverge sharply ($\rho = 0.25$). Pair accuracy is scored against proprietary-RM targets for every row, so the lower DUALEVAL (pub.) value reflects disagreement between reward models rather than a failure of the joint fit.

Robustness to Structured Missingness. Many practical leaderboards are incomplete, and their missingness can be systematic: newer strong models are often not run on older saturated benchmarks. Averaging over observed cells inherits this bias, since a strong model that skipped easy items looks worse than it is, whereas DUALEVAL models the observed cells jointly with item difficulty.

To avoid scoring each method against its own preferred ranking, we generate semi-synthetic data from the shared latent item-response layer of a DUALEVAL fit pooled over all four domains (18×2100 cells, dense before masking), so the true ordering θ^* is known, and mask 40% or 60% of cells under a missing-completely-at-random (MCAR) mechanism or under an ability-correlated missing-not-at-random (MNAR) mechanism whose drop propensity is proportional to

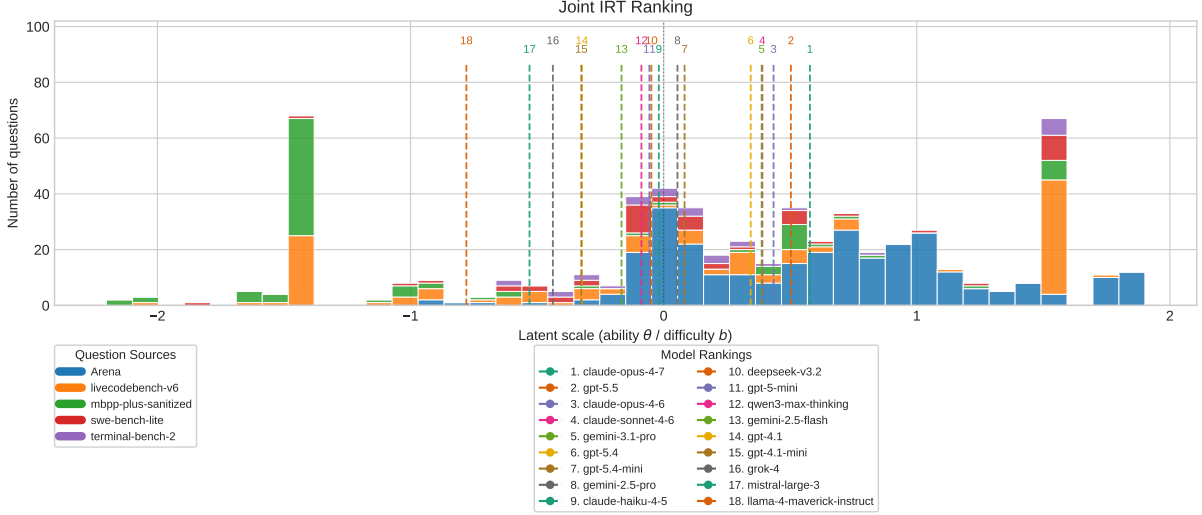


Figure 1: Rankings of model abilities and model-centric question difficulties on the **coding** domain, using our proprietary reward model.

Method	Coding				Math				Misc			
	Pair Acc.	Static ρ	Arena ρ	Public ρ	Pair Acc.	Static ρ	Arena ρ	Public ρ	Pair Acc.	Static ρ	Arena ρ	Public ρ
Static 2PL	0.731	1.000	0.773	0.763	0.641	1.000	0.571	0.682	0.701	1.000	0.695	0.701
Arena BT (prop.)	0.809	0.773	1.000	0.851	0.730	0.571	1.000	0.860	0.806	0.695	1.000	0.874
Arena BT (pub.)	0.680	0.255	0.416	0.511	0.667	0.457	0.624	0.536	0.779	0.220	0.350	0.247
DUALEVAL (prop.)	0.778	0.959	0.878	0.845	0.682	0.990	0.734	0.703	0.781	0.893	0.940	0.833
DUALEVAL (pub.)	0.591	0.976	0.805	0.789	0.600	0.984	0.625	0.692	0.612	0.963	0.724	0.657

Table 2: Method comparison across domains and reward models, averaged over three random question splits (test fraction 0.2). Pair Acc.: held-out arena decisive-pair accuracy, scored against proprietary-RM soft targets for every row. Static/Arena ρ : Spearman correlation with the full-data static-only IRT and Arena BT rankings; Public ρ : with the matched public LMArena sub-leaderboard (Text-Coding/Text-Math/Text-Expert), over the same 18 models. prop./pub.: proprietary RM and Skywork-Reward-V2-Qwen3-8B. Average-reward baseline in Appendix §G.

ability_rank(model) $\times$ age(source) (three seeds; Appendix §O). Under MCAR all methods recover θ^* well ($\rho \approx 0.95$ – 0.98), so the MNAR gap is not simply an artifact of using an item-response generator: fusion is competitive when missingness is ignorable and loses 0.10–0.16 in rank agreement only once it correlates with ability, while DUALEVAL’s penalty is approximately 0.03 or smaller.

Ablations. We additionally ablate both-bad anchoring and tie filtering and evaluate their effects. Removing the both-bad loss reduces both-bad AUC from 0.903 to 0.622 and also lowers held-out arena pair accuracy from 0.778 to 0.771, showing that it contributes to both absolute failure identification and relative preference prediction. Omitting tie filtering produces smaller but consistent degradations in both metrics. The full configuration therefore performs best overall and is used throughout; detailed results are provided in Appendix §M, Table 10.

4.2 Item Informativeness

A central diagnostic question for any benchmark is which items carry the bulk of the ranking signal between models, and which are redundant. The joint formulation produces item-level parameters that let us answer this question directly: we score each item by its Fisher information about model abilities under the fitted DUALEVAL objective, then quantify how much of the full-data ranking can be recovered from the most informative items alone.

Quantifying Informativeness via Fisher Score.

We score each item by the Fisher information it carries about model abilities under the fitted DUALEVAL objective. For a static item this is the standard 2PL information $a_q^2 p_{i,q}(1 - p_{i,q})$, averaged over models; for an arena item it is the trace of the Fisher information with respect to the two abilities involved in a decisive pair, averaged over that item’s decisive pairs. Each term is weighted by its loss coefficient ($\lambda_{\text{static}}, \lambda_{\text{pref}}$), so static and arena items are scored in the same units – expected infor-

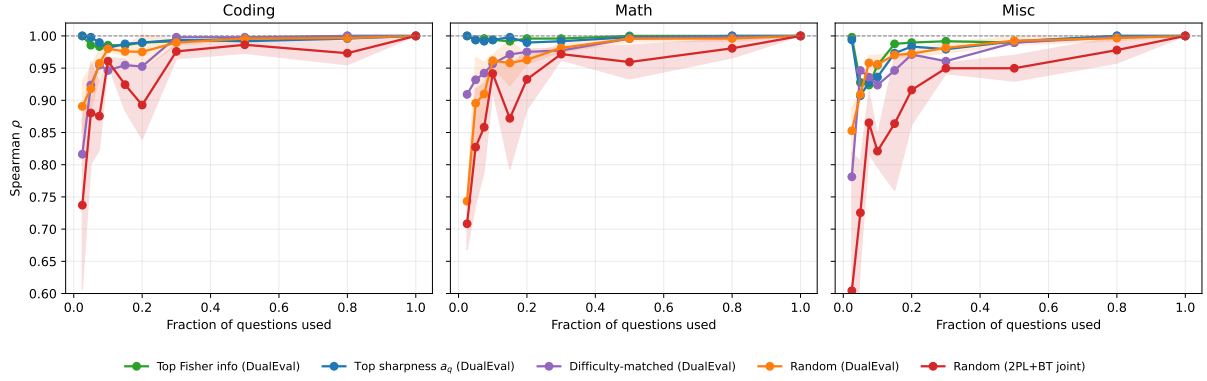


Figure 2: In-pool subset-recovery agreement (Spearman ρ vs. the full-data DUALEVAL ranking) against the fraction of items retained, for coding, math, and miscellaneous. Selection strategies: Top Fisher info (DUALEVAL), the sharpness-only and difficulty-matched ablations, and two random baselines.

mation about the model-ability vector – and averaging over pairs prevents arena items from gaining an artificial advantage from their $O(M^2)$ comparisons. Full expressions are given in Appendix §K.

In-Pool Subset Recovery. This experiment is in-pool by construction: item parameters and Fisher scores come from the same 18 models whose ranking is then recovered, so it measures redundancy within a fixed pool rather than transfer. For each domain, we select a fraction of items, refit ability parameters on the subset (item parameters held fixed at full-data values), and measure Spearman correlation between the resulting ranking and the full-data DUALEVAL ranking. We compare five selection strategies, two of which ablate a component of S_q : **Top Fisher info** (descending S_q); **Top sharpness** a_q , dropping the $p(1 - p)$ frontier weight; **Difficulty-matched**, smallest $|b_q - \text{median}(\theta)|$, i.e. frontier-matched difficulty alone; and uniform **Random** subsets refit with DUALEVAL and with a single-source 2PL+BT baseline that has no arena item parameters.

Ranking Signal Is Concentrated in a Small Subset. Current benchmarks are substantially redundant (Figure 2): in all three static-anchored domains the top 10% of items by Fisher information recover the full-data ranking at Spearman $\rho \geq 0.95$, and the top 30% at $\rho \geq 0.99$. The full score tracks the sharpness-only ablation closely but is the most consistent at small fractions, so sharpness explains much of the signal while local non-saturation still helps. Difficulty matching alone is less reliable: proximity to the ability range does not by itself confer sensitivity to model differences. The 2PL+BT baseline trails throughout, indicating that jointly

fit item parameters carry information missing from disjoint-source rankings.

Out-of-Pool Transfer to Unseen Models. We now remove the in-pool dependence. Holding out each model in turn, we calibrate both the item bank and the Fisher scores on the remaining 17, then place the held-out model by estimating its ability with item parameters frozen, from its outcomes on the selected subset alone; ability uses both modalities and selection is stratified within each, so the subset keeps the static/arena mix (Appendix §L). We report mean $|\text{transferred rank} - \text{full-data in-pool rank}|$ over 54 model-domain folds (18 held-out models $\times$ 3 static-anchored domains). We use rank error rather than rank correlation because correlation is dominated by the 17 in-pool models, which barely move, and stays high even when the newcomer is badly misplaced; random placement among 18 positions gives ≈ 6.0 . Alongside the two fusion baselines we report **refit-from-scratch**, which re-estimates everything on the subset *with* the held-out model included – an optimistic in-pool reference, not an out-of-pool method.

According to Table 3, a bank that has never seen a model still places it within 2.11 ranks from 10% of the items, that is 30/20/30 graded static responses for coding/math/misc plus the matching arena items (Table 3). DUALEVAL at 10% already beats mean-z fusion given the *entire* item set (2.67): a pre-calibrated bank makes a small selected subset go a long way, which is exactly the leaderboard-refresh regime. Error is not monotone in subset size (1.56 at 50%, 1.83 at 100%), which may reflect finite-sample variation, low-information observations, or model mismatch; the fusion baselines are

Method	Fisher-selected items			
	10%	25%	50%	100%
DUALEVAL transfer	2.11	1.83	1.56	1.83
Mean-z fusion	3.50	4.00	3.72	2.67
Rank-avg fusion	2.56	2.28	2.94	3.00
Refit-from-scratch	1.67	0.72	0.33	0.00

Table 3: Out-of-pool rank recovery from a frozen bank: mean $|\text{transferred} - \text{in-pool rank}|$ over 54 model-domain folds (18 held-out models $\times$ 3 domains), lower better. Bold: best out-of-pool; refit-from-scratch is an in-pool reference. Appendix §L.

non-monotone throughout. Refit-from-scratch at 100% reproduces the in-pool ranking exactly, validating the procedure. Because informativeness is relative to the evaluated pool, high-signal subsets should be recalibrated periodically as abilities shift.

4.3 Residual-Based Anomaly Detection

A second diagnostic question is whether the joint fit can identify suspicious (model, item) pairs whose observed outcomes are inconsistent with what DUALEVAL expects given the model’s ability and the item’s difficulty. Such residual anomalies are candidates for contamination, leakage, or data-quality issues, and the joint model’s per-cell expectations make them directly computable.

Per-Cell Residual Scores. For a static (i, q) pair we define the standardized residual

$$r_{i,q}^{\text{static}} = \frac{y_{i,q} - p_{i,q}}{\sqrt{p_{i,q}(1 - p_{i,q}) + \varepsilon}},$$

where $y_{i,q} \in \{0, 1\}$ is the observed correctness label and $p_{i,q}$ is a DUALEVAL success probability. Large positive residuals indicate unexpected successes; large negative residuals, unexpected failures. For an arena (i, q) pair, we average the target-prediction gap across decisive opponents of model i on item q :

$$r_{i,q}^{\text{arena}} = \frac{1}{|\mathcal{P}_q^{(i)}|} \sum_{j \in \mathcal{P}_q^{(i)}} (\sigma(z_{i,q} - z_{j,q}) - \mu_{ijq}),$$

where $z_{i,q}$ is the standardized reward, $\mu_{ijq} = \sigma(\gamma(p_{i,q} - p_{j,q}))$ is the DUALEVAL-predicted pairwise probability, and $\mathcal{P}_q^{(i)}$ is the set of decisive opponents of model i on item q .

In-Pool Targeted Injection. We first evaluate residual screening under controlled injection with

all 18 models in the calibration pool: static labels are flipped on failed cells in low-probability success regions, and arena rewards in the bottom 30% are shifted by $+2.0$ standardized units before the pairwise targets are rebuilt, at rates $\{0.5, 1, 2, 5\}\%$ with five seeds. Comparing a *contaminated-fit* detector, refit on the corrupted data, against a *clean-reference* detector that scores corrupted outcomes under the original fit, clean-reference residuals reach AUROC ≥ 0.995 on static and 0.93–0.98 on arena contamination, while the contaminated-fit detector stays above random but weaker, since contamination is partly absorbed into the refitted parameters (Appendix §P). Because this injection is both targeted and written into the quantity the residual measures, we treat it as an upper bound and turn to two stricter protocols.

Out-of-Pool Screening of a New Submission.

The setting above still calibrates the screened model’s ability on its own clean data. We now refit the clean reference on the other 17 models, so the bank has never seen m ; the residual definitions are unchanged. To place m on the frozen scale we estimate θ_m from the modality that is *not* being screened – clean arena pairs when static cells are screened, clean static responses when arena cells are screened – so the screened modality never supplies the ability estimate and the detector cannot absorb the contamination it is meant to find. This cross-modality identification exists only under joint calibration, which is also why the arena column admits no static-only competitor: a static-only bank has neither arena item parameters to form a residual against, nor a second modality from which to identify an unseen model’s ability. The dashes in Table 4 are structural, not omissions. Baselines are a **static 2PL residual** under the same protocol and two parameter-free detectors: **difficulty-only**, scoring a static success by the item’s empirical failure rate over the other models, and **raw reward**.

Detection holds out of pool: AUROC 0.920–0.935 on static and 0.995–1.000 on arena contamination. The screened modality contributes neither to bank calibration nor to the held-out model’s ability estimate; the latter is estimated from clean responses in the opposite modality. The static margin over a static-only IRT residual (0.905–0.925) is modest and difficulty-only trails by 6–9 points; the substantive gain is on the arena side, where the joint bank is the only one of these detectors that is defined at all.

Method	Static	Arena
DUALEVAL residual	0.920–0.935	0.995–1.000
Static 2PL residual	0.905–0.925	—
Difficulty-only	0.840–0.870	—
Raw reward	—	0.930–0.935

Table 4: Out-of-pool screening, leave-one-model-out. Mean over all 18 folds; range spans injection rates $\{0.5, 1, 2, 5\}\%$ with domains pooled. Dashes mark detectors undefined for that modality.

Out-of-Pool, Response-Level Contamination.

Two properties make the in-pool injection easy: cells are drawn from the most suspicious region, and contamination is written directly into the quantity the residual measures. We relax both on math and miscellaneous, screening with the out-of-pool reference above. Cells are now sampled uniformly from failed cells, and contamination enters through an actual response: for a static cell we substitute a correct peer response verbatim and re-grade it with the unchanged judge, so the label flips only if the judge accepts; for an arena cell we substitute a top-quartile donor’s response and reuse that response’s reward, which is identical to re-scoring the unchanged prompt–response pair, giving a heterogeneous shift (mean $\Delta z \approx 0.7$ – 1.3 , occasionally non-positive) instead of a fixed offset. Items that no model solved have no donor and are excluded (Appendix §P).

Because the outcome is produced by the pipeline rather than written by us, the positive label is not true by construction: the judge rejects 5–9% of substituted correct responses, which remain $y = 0$ – a statistic with no analogue in label flipping. Coding is excluded because its labels come from execution rather than the judge, so a substituted passing solution would pass by definition and the protocol would reduce to label flipping. At 5% substitution DUALEVAL reaches AUROC 0.944/0.969 on static math/misc and 0.916/0.881 on arena math/misc (Table 5): clearly strongest on static math, and numerically comparable to the strongest baseline elsewhere. Calibrated residuals therefore remain useful when contamination is neither targeted nor injected in the space the detector measures.

5 Discussion

DUALEVAL treats evaluation items as calibrated measurement instruments rather than interchangeable test cases. Once items carry difficulty, sharp-

Setting	Method	Math	Misc
Static	DUALEVAL	0.944	0.969
	Acc-based	0.803	0.960
	Item-LOO	0.845	0.959
Arena	DUALEVAL	0.916	0.881
	Arena-2PL	0.904	0.879
	BT	0.912	0.868

Table 5: Mean model–item cell-level AUROC under 5% uniform response substitution, averaged over five random seeds. Screening uses the out-of-pool reference protocol of Table 4. Appendix §P provides the substitution procedure and baseline definitions.

ness, informativeness, and residual profiles, benchmark maintenance becomes a model-item problem: saturated items can be identified, high-signal subsets preserved, and suspicious outcomes audited. Importantly, these item profiles can transfer beyond the calibration pool: a pre-calibrated bank can efficiently place unseen models using only a small high-information subset, while joint item modeling also reduces sensitivity to structured missingness.

This perspective also points to richer extensions. A scalar ability parameter is useful for interpretability and ranking, but many evaluation domains contain multiple skills; multidimensional IRT provides a natural path toward modeling specialization across reasoning, coding, factuality, and interaction quality. More broadly, calibrated item profiles could support adaptive evaluation protocols that choose questions based on the current uncertainty about a model, while retaining item-level evidence for why a ranking changed.

6 Conclusion

We introduced DUALEVAL, a framework that jointly models static benchmark correctness and arena-style preference signals on a shared latent scale. Across four domains, DUALEVAL produces balanced and interpretable model rankings while estimating item difficulty, sharpness, and informativeness. The learned item profiles further support sample-efficient placement of unseen models and residual-based anomaly screening under out-of-pool and less-targeted contamination settings. Together, these results point toward evaluation pipelines that jointly rank models and calibrate benchmark items, making leaderboards more efficient, informative, and grounded in item-level evidence.

Limitations

Our study has several limitations. First, the primary arena supervision relies on a proprietary, non-public reward model, which limits direct reproducibility even though we describe its architecture, training objective, and training-data scale. We address this by validating the RM against held-out human preferences (Appendix §C) and replicating the main analyses with the publicly available Skywork-Reward-V2-Qwen3-8B model (Appendix §F), but broader validation across multiple public RMs and human preference annotations remains important.

Second, each model-question pair is evaluated with a single generated response. This matches many leaderboard settings but does not capture within-model sampling variability, sensitivity to decoding parameters, or multi-attempt performance. Extending DUALEVAL to repeated responses would allow item sharpness and residual anomalies to separate model capability from response-level stochasticity.

Third, the current formulation uses a scalar ability parameter within each domain. This keeps rankings interpretable, but future evaluation systems should move toward multidimensional model profiles that capture specialization across skills, task formats, and interaction settings. Such extensions could make item selection adaptive not only to overall model strength, but also to the specific capabilities an evaluator wants to measure.

References

- Terry A Ackerman, Mark J Gierl, and Cindy M Walker. 2003. Using multidimensional item response theory to evaluate educational and psychological tests. *Educational Measurement: Issues and Practice*, 22(3):37–51.
- Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and 1 others. 2021. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*.
- Valerio Basile, Michael Fell, Tommaso Fornaciari, Dirk Hovy, Silviu Paun, Barbara Plank, Massimo Poesio, and Alexandra Uma. 2021. We need to consider disagreement in evaluation. In *Proceedings of the 1st workshop on benchmarking: past, present and future*, pages 15–21.
- Samuel Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 632–642.
- Ralph Allan Bradley and Milton E Terry. 1952. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345.
- Jane Castleman, Nimra Nadeem, Tanvi Namjoshi, and Lydia T Liu. 2025. Rethinking math benchmarks for llms using irt. *Proceedings of Machine Learning Research*, 273:66–82.
- Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, and 1 others. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.
- Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Banghua Zhu, Hao Zhang, Michael Jordan, Joseph E Gonzalez, and 1 others. 2024. Chatbot arena: An open platform for evaluating llms by human preference. In *Forty-first International Conference on Machine Learning*.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.
- Susan E Embretson and Steven P Reise. 2013. *Item response theory for psychologists*. Psychology Press.
- Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, and 1 others. 2024. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3828–3850.
- Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.
- Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.
- Valentin Hofmann, David Heineman, Ian Magnusson, Kyle Lo, Jesse Dodge, Maarten Sap, Pang Wei Koh, Chun Wang, Hannaneh Hajishirzi, and Noah A Smith. 2025. Fluid language model benchmarking. *arXiv preprint arXiv:2509.11106*.

- Naman Jain, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. 2025. Livecodebench: Holistic and contamination free evaluation of large language models for code. In *International Conference on Learning Representations*, volume 2025, pages 58791–58831.
- Carlos E Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. 2024. Swe-bench: Can language models resolve real-world github issues? In *International Conference on Learning Representations*, volume 2024, pages 54107–54157.
- Alex Kipnis, Konstantinos Voudouris, Luca Schulze Buschoff, and Eric Schulz. 2025. metabench-a sparse benchmark of reasoning and knowledge in large language models. In *International Conference on Learning Representations*, volume 2025, pages 31734–31770.
- Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, and 1 others. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.
- John P Lalor, Hao Wu, and Hong Yu. 2019. Learning latent parameters without human response patterns: Item response theory with artificial crowds. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing. Conference on Empirical Methods in Natural Language Processing*, volume 2019, page 4240.
- Peiyu Li, Xiuxiu Tang, Si Chen, Ying Cheng, Ronald Metoyer, Ting Hua, and Nitesh V Chawla. 2025. Adaptive testing for llm evaluation: A psychometric alternative to static benchmarks. *arXiv preprint arXiv:2511.04689*.
- Tianle Li, Wei-Lin Chiang, Evan Frick, Lisa Dunlap, Tianhao Wu, Banghua Zhu, Joseph E Gonzalez, and Ion Stoica. 2024. From crowdsourced data to high-quality benchmarks: Arena-hard and benchbuilder pipeline. *arXiv preprint arXiv:2406.11939*.
- Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, and 1 others. 2022. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*.
- Lele Liao, Qile Zhang, Ruofan Wu, and Guanhua Fang. 2025. Toward a unified framework for data-efficient evaluation of large language models. *arXiv preprint arXiv:2510.04051*.
- Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. 2023. Is your code generated by chatgpt really correct? rigorous evaluation of large language models for code generation. *Advances in neural information processing systems*, 36:21558–21572.
- Frederic M Lord and Melvin R Novick. 2008. *Statistical theories of mental test scores*. IAP.
- Inbal Magar and Roy Schwartz. 2022. Data contamination: From memorization to exploitation. *arXiv preprint arXiv:2203.08242*.
- Fernando Martínez-Plumed, Ricardo BC Prudêncio, Adolfo Martínez-Usó, and José Hernández-Orallo. 2019. Item response theory in ai: Analysing machine learning classifiers at the instance level. *Artificial intelligence*, 271:18–42.
- Mathematical Association of America. 2026. American invitational mathematics examination (aime). MAA Invitational Competitions.
- Mike A Merrill, Alexander G Shaw, Nicholas Carlini, Boxuan Li, Harsh Raj, Ivan Bercovich, Lin Shi, Jeong Yeon Shin, Thomas Walshe, E Kelly Buchanan, and 1 others. 2026. Terminal-bench: Benchmarking agents on hard, realistic tasks in command line interfaces. *arXiv preprint arXiv:2601.11868*.
- Jinjie Ni, Fuzhao Xue, Xiang Yue, Yuntian Deng, Mahir Shah, Kabir Jain, Graham Neubig, and Yang You. 2024. Mixeval: Deriving wisdom of the crowd from llm benchmark mixtures. *Advances in Neural Information Processing Systems*, 37:98180–98212.
- Simon Ott, Adriano Barbosa-Silva, Kathrin Blagec, Jan Brauner, and Matthias Samwald. 2022. Mapping global dynamics of benchmark creation and saturation in artificial intelligence. *Nature Communications*, 13(1):6793.
- Long Phan, Alice Gatti, Ziwen Han, Nathaniel Li, Josephina Hu, Hugh Zhang, Chen Bo Calvin Zhang, Mohamed Shaaban, John Ling, Sean Shi, and 1 others. 2025. Humanity’s last exam. *arXiv preprint arXiv:2501.14249*.
- Felipe Maia Polo, Lucas Weber, Leshem Choshen, Yuekai Sun, Gongjun Xu, and Mikhail Yurochkin. 2024. tinybenchmarks: evaluating llms with fewer examples. *arXiv preprint arXiv:2402.14992*.
- Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.
- Zachary Robertson. 2025. Identity-link irt for label-free llm evaluation: Preserving additivity in tvd-mi scores. *arXiv preprint arXiv:2510.14966*.
- Pedro Rodriguez, Joe Barrow, Alexander Miserlis Hoyle, John P Lalor, Robin Jia, and Jordan Lee Boyd-Graber. 2021. Evaluation examples are not equally informative: How should that change nlp leaderboards? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4486–4503.

Chris Van Der Lee, Albert Gatt, Emiel Van Miltenburg, Sander Wubben, and Emiel Krahmer. 2019. Best practices for the human evaluation of automatically generated text. In *Proceedings of the 12th international conference on natural language generation*, pages 355–368.

Clara Vania, Phu Mon Htut, William Huang, Dhara Mungra, Richard Yuanzhe Pang, Jason Phang, Haokun Liu, Kyunghyun Cho, and Samuel Bowman. 2021. Comparing test sets with item response theory. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1141–1158.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP workshop BlackboxNLP: Analyzing and interpreting neural networks for NLP*, pages 353–355.

Shaobo Wang, Cong Wang, Wenjie Fu, Yue Min, Mingquan Feng, Isabel Guan, Xuming Hu, Conghui He, Cunxiang Wang, Kexin Yang, and 1 others. 2025. Rethinking llm evaluation: Can we evaluate llms with 200x less data? *arXiv preprint arXiv:2510.10457*.

Zhuo Wang, Wen Wu, Guoqing Wang, Guangze Ye, and Zhenxiao Cheng. 2026. Metaeval: Measuring the discrimination of benchmarks for efficient llm evaluation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 33773–33781.

Jason Wei, Nguyen Karina, Hyung Won Chung, Yunxin Joy Jiao, Spencer Papay, Amelia Glaese, John Schulman, and William Fedus. 2024. Measuring short-form factuality in large language models. *arXiv preprint arXiv:2411.04368*.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long papers)*, pages 1112–1122.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 2369–2380.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhonghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, and 1 others. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.

Hongli Zhou, Hui Huang, Ziqing Zhao, Lvyuan Han, Huicheng Wang, Kehai Chen, Muyun Yang, Wei Bao, Jian Dong, Bing Xu, and 1 others. 2026. Lost in benchmarks? rethinking large language model benchmarking with item response theory. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 35085–35093.

A Additional DUALEVAL Ranking Visualizations

We provide additional examples of DUALEVAL rankings of math and misc, and generic domains in Figures 3, 4, and 5.

B DUALEVAL Hyperparameters and Configurations

All DUALEVAL experiments in this paper use the same hyperparameters unless otherwise noted: Adam optimizer with learning rate 0.02, 2000 epochs, loss weights $\lambda_{\text{static}} = \lambda_{\text{pref}} = 1.0$ and $\lambda_{\text{bb}} = 0.2$, L2 regularization coefficient $\lambda_{\text{reg}} = 0.01$ applied to θ , b , k , and $\log \tilde{\gamma}$, and pair-treatment thresholds derived as the 15th percentile of $\max(z_i, z_j)$ for both-bad flagging and the 15th percentile of $|z_i - z_j|$ for tie filtering, applied per-domain. The arena temperature is parameterized as $\gamma = \gamma_0 \cdot \tilde{\gamma}$ with $\gamma_0 = 4$, and $\log \tilde{\gamma}$ is L2-regularized toward zero. After each optimizer step, model abilities are zero-mean centered with the same shift absorbed into question difficulties, preserving all differences $\theta_i - b_q$. All fits use single-precision floating point on CPU; a single domain’s joint fit completes in 10–25 seconds. Arena-only fits on the generic domain use the same configuration with the static loss disabled.

C Validation of Reward Model with Real Human Preference

This appendix validates the reward models against real human preferences on the arena prompts used by our evaluation pipeline. We quantify how well the proprietary and public reward models agree with decisive human preference labels across all four domains.

Reward-model alignment with human preference. For each domain we identify a subset of arena prompts for which human preference labels are available, and that have a decisive (non-tie) human winner. Sample sizes per domain are reported in Table 6. Both reward models are scored on the same pairs from this filtered subset. Table 6 reports

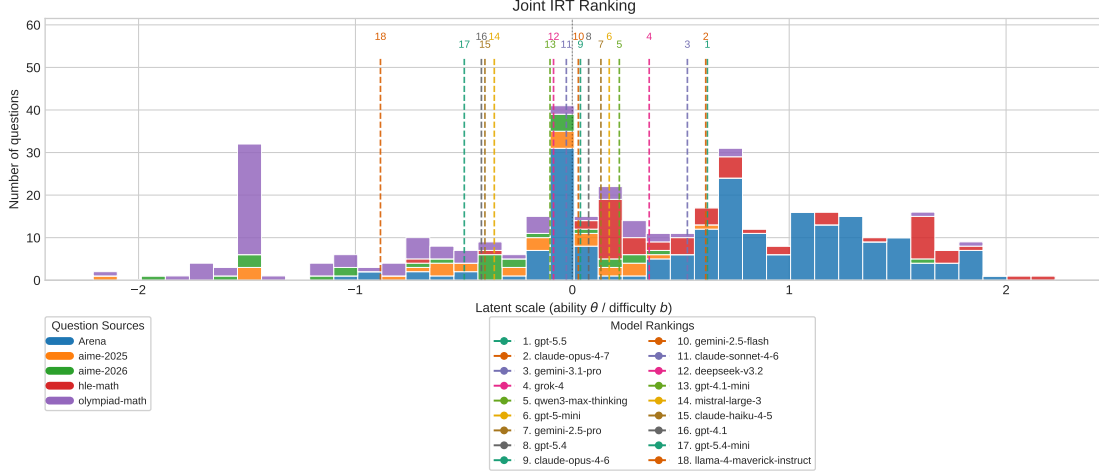


Figure 3: Learned model abilities and question difficulties on **math**.

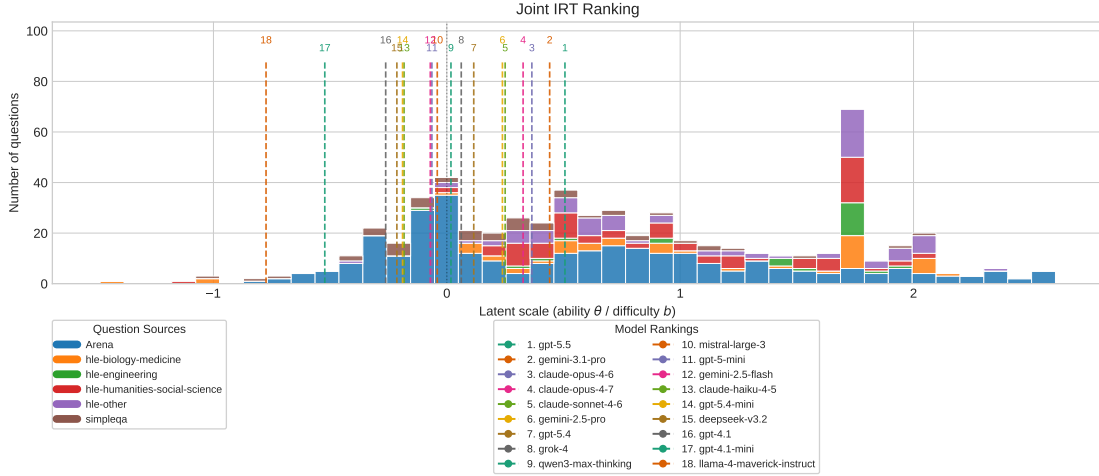


Figure 4: Learned model abilities and question difficulties on **miscellaneous**.

the per-domain pair count and the RM’s accuracy at predicting the human winner. The proprietary RM is substantially more human-aligned than the public RM in every domain.

Domain	N	Proprietary RM	Public RM
Coding	195	0.739	0.573
Math	120	0.875	0.708
Misc	200	0.715	0.655
Generic	376	0.742	0.596

Table 6: Pairwise accuracy of each reward model at predicting the human winner. Restricted to pairs with a decisive (non-tie) human-preference winner.

D Stability of DualEval Rankings

We assess the stability of DUALEVAL rankings using a question-level cluster bootstrap. For each domain we draw $B = 100$ bootstrap replicates, resampling questions with replacement stratified

by source (each benchmark source plus arena), so every replicate has the same per-source N as the original. Drawn copies of a question receive unique IDs so the IRT fit treats them as independent items—the standard cluster-bootstrap behavior. Reward z -scores are renormalized per replicate. For each replicate we refit DUALEVAL with the published hyperparameters and record per-model $\hat{\theta}_i$. We use the same procedure to bootstrap Static 2PL and Arena BT for direct comparison (Static 2PL is omitted for the arena-only generic domain).

Figure 7 shows per-model $\hat{\theta}_i$ with symmetric 95% CIs ($\hat{\theta}_i \pm 1.96, \hat{\sigma}_{\text{boot}}$) for all three methods in each domain. Top-3 and bottom-3 positions are stable to ± 1 rank across replicates in every domain. Middle-band positions span 4–6 ranks of 95% CI in the three static-anchored domains, indicating that not all adjacent leaderboard positions are statistically distinguishable at the current evaluation size. Generic—being arena-only with 500 prompts

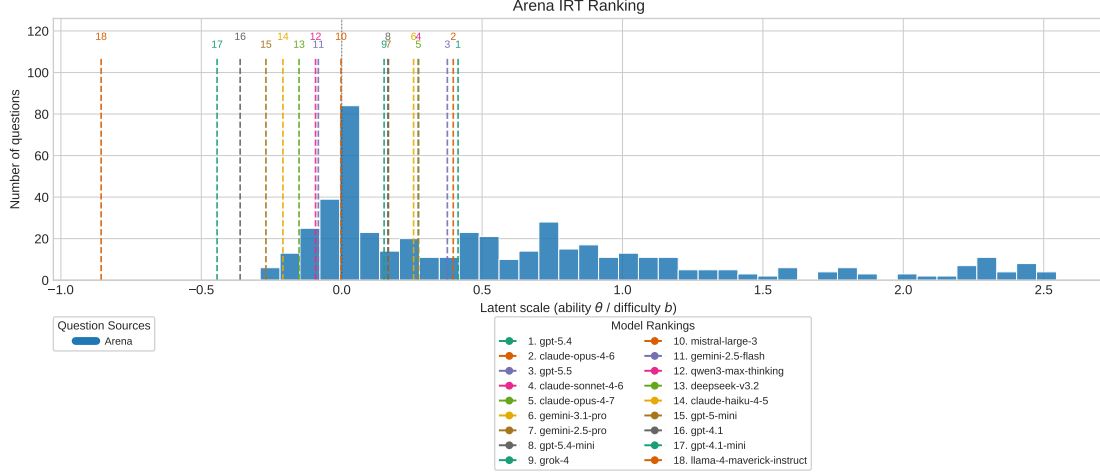


Figure 5: Learned model abilities and question difficulties on **generic**.

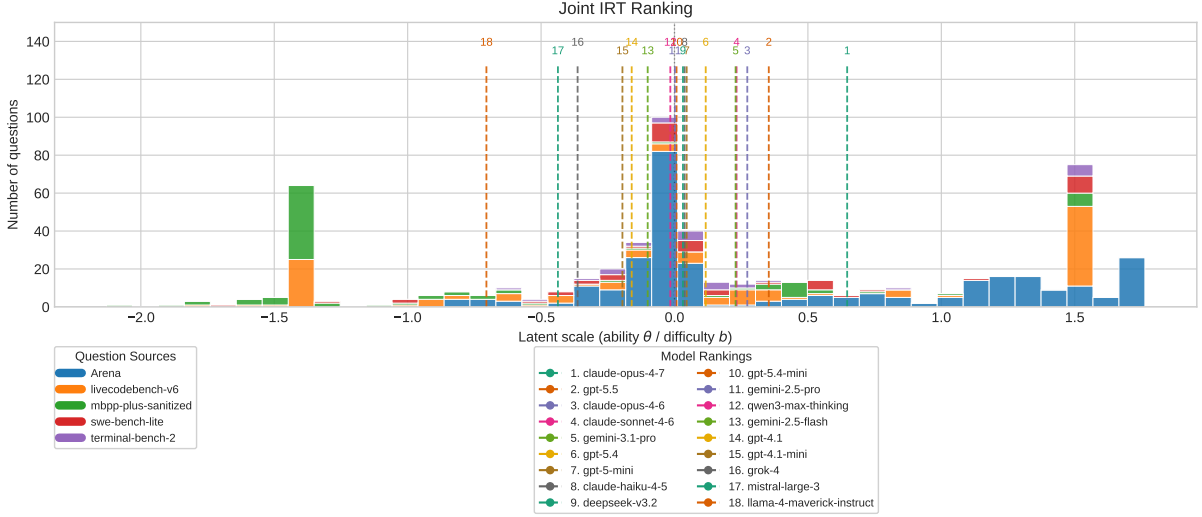


Figure 6: Rankings of model abilities and model-centric question difficulties on the **coding** domain, using public reward model Skywork-Reward-V2-Qwen3-8B.

contributing $\binom{18}{2}$ pairwise comparisons per item—produces the tightest CIs across the four domains.

Table 7 reports per-(domain, method) summary statistics. On equivalent data (generic, arena-only), DUALEVAL matches or slightly improves over Arena BT (median $\hat{\sigma}_\theta = 0.016$ vs 0.019), confirming that the IRT structure adds discriminative information without inflating uncertainty. Against Static 2PL on the static-anchored domains, the joint formulation reduces ranking variance by 30–60%. Against Arena BT on those domains, DUALEVAL carries a modest variance premium that reflects the additional cross-source information being integrated, in exchange for substantially better Static- ρ (Table 2).

E Held-out calibration

We evaluate calibration of the joint fit’s per-pair predictions on a cell-level 80/20 random holdout (averaged over three random seeds; same fit hyperparameters as §4). On held-out static cells, predicted success probabilities $p_{i,q}$ match observed binary outcomes with expected calibration error (ECE) of 0.04–0.07 across coding/math/misc domains. On held-out arena pairs, predicted preference probabilities μ_{ijq} match the soft pairwise targets $\sigma(z_{i,q} - z_{j,q})$ with ECE of 0.01–0.03, with reliability diagrams nearly diagonal across the predicted-probability range (Figure 8). For transparency we also report ECE against the binarised pairwise outcome $1[\sigma(z_{i,q} - z_{j,q}) \geq 0.5]$, which is higher (0.07–0.16) due to expected underconfidence when soft pairwise targets near 0.5 are

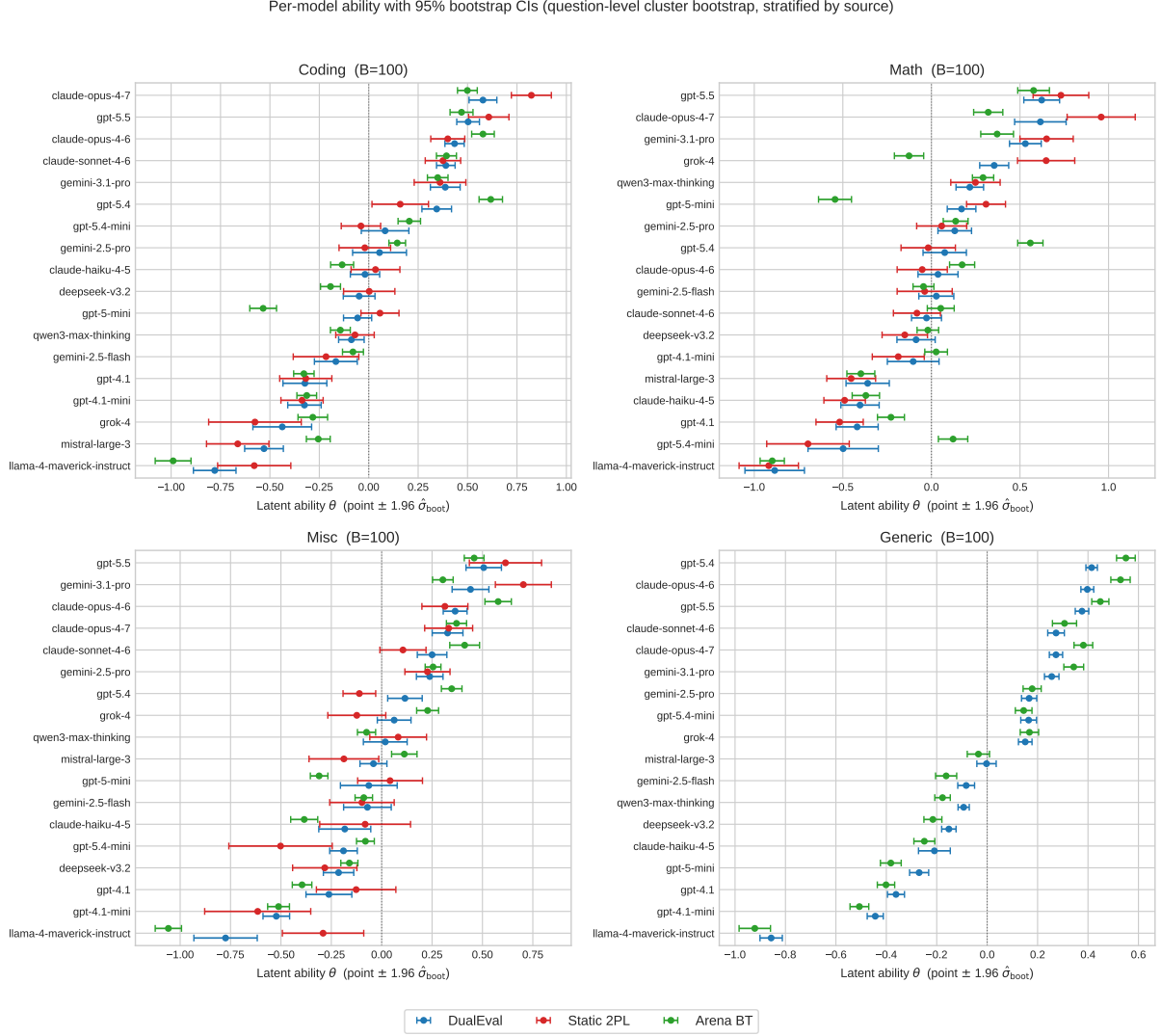


Figure 7: Per-model ability with 95% bootstrap CIs (question-level cluster bootstrap, $B = 100$, stratified by source).

discretised to hard win/loss labels; this binarisation does not reflect a miscalibration of the underlying joint fit. These calibration results justify treating $p_{i,q}$ and μ_{ijq} as meaningful probability estimates in the downstream Fisher information (§4.2) and residual diagnostics (§4.3).

F Cross-RM Ranking Agreement

Figure 9 plots DUALEVAL’s learned model abilities under the proprietary and public RMs against each other, one panel per static-anchored domain. Points fall close to the diagonal: Spearman correlations are 0.963 (coding), 0.992 (math), and 0.835 (misc), consistent with the Static $\rho \geq 0.96$ entries in Table 2 and indicating that static anchoring stabilizes the ranking against RM-induced noise. The arena-only generic domain provides the no-anchor counterfactual: with no static labels to pin the la-

tent scale, the agreement between the proprietary and public RMs drops to $\rho = 0.25$, empirically isolating static anchoring as the mechanism behind the joint framework’s cross-RM stability.

G Average-Reward Baseline

The arena-only average-reward ranking scores each model by its mean standardized reward, with no pairwise model at all. Table 8 reports it under the same protocol and columns as Table 2. It tracks Arena BT almost exactly: because the soft pairwise targets are a monotone function of reward differences, the Bradley–Terry fit recovers the mean-reward ordering, so the two agree on every rank-based column and differ only where tie and both-bad filtering removes pairs.

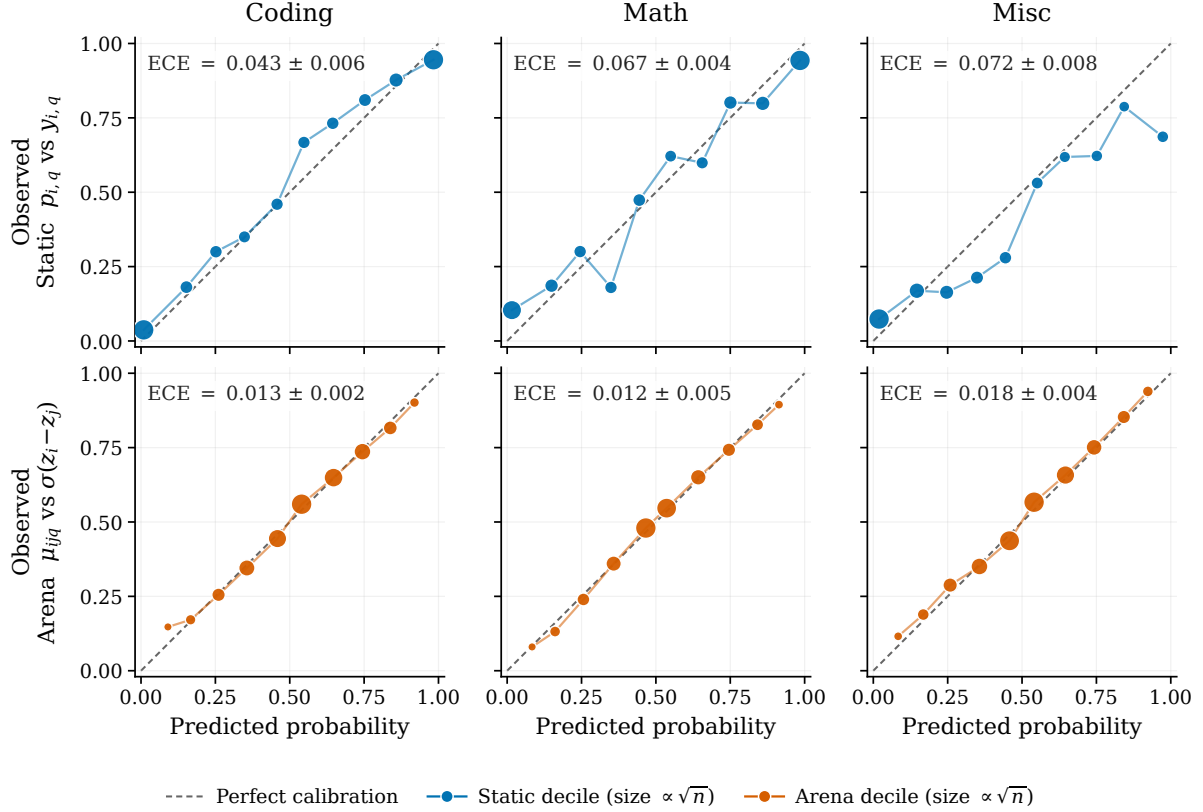


Figure 8: Held-out calibration of DualEval predictions. Rows: static cells ($p_{i,q}$ against binary correctness) and arena pairs (μ_{ijq} against soft pairwise targets $\sigma(z_i - z_j)$). Columns: coding, math, miscellaneous domains. Each point is a decile of the predicted-probability axis, with size $\propto \sqrt{n}$. Dashed line marks perfect calibration. ECE (mean $\pm$ SD over three seeds) annotated per panel.

H Evaluation Details

Benchmark sources. Our static benchmark pool draws from LiveCodeBench (Jain et al., 2025), MBPP-Plus from EvalPlus (Liu et al., 2023), SWE-Bench (Jimenez et al., 2024), TerminalBench (Merrill et al., 2026), AIME 2025/2026 (Mathematical Association of America, 2026), Humanity’s Last Exam (HLE) (Phan et al., 2025), OlympiadBench/Olympiad-Math (He et al., 2024), and SimpleQA (Wei et al., 2024). Arena prompts are drawn from publicly available LMArena (Chiang et al., 2024) logs and filtered by domain as described below.

Prompt sampling and filtering. Static items are drawn from each source benchmark by seeded stratified sampling, with strata defined by the natural axes of each source: difficulty tier for LiveCodeBench-v6, problem type for AIME and Olympiad-Math, HLE level for HLE-Math, and field for HLE-Misc (Bio./Med., Engineering, Humanities/Social Science, Other); SimpleQA is stratified by topic. MBPP-Plus, SWE-Bench-

lite, and TerminalBench-2.0 are taken in full up to the per-source quotas in Table 1, after dropping items shorter than 150 characters. Arena prompts are sampled from LMArena releases: arena-expert-5k for coding, math, and miscellaneous, and arena-human-preference-140k for generic. We first apply a coarse domain pre-filter using the dataset-provided occupational and category tags, then run an LLM judge (gpt-5.4-mini) with a domain-specific rubric that returns an inclusion label and a confidence score: coding requires the answer to depend on runnable code, math requires the core task to be mathematical reasoning or calculation, miscellaneous requires expert domain knowledge outside code/math, and generic requires a non-technical everyday task with no code or math content. Candidates with confidence below the per-domain threshold (0.5 for coding/math/misc, 0.8 for generic) are dropped, and the top- N surviving prompts by judge confidence are retained to match the per-domain arena pool sizes in Table 1 (coding 300, math 200, miscellaneous 300, generic 500).

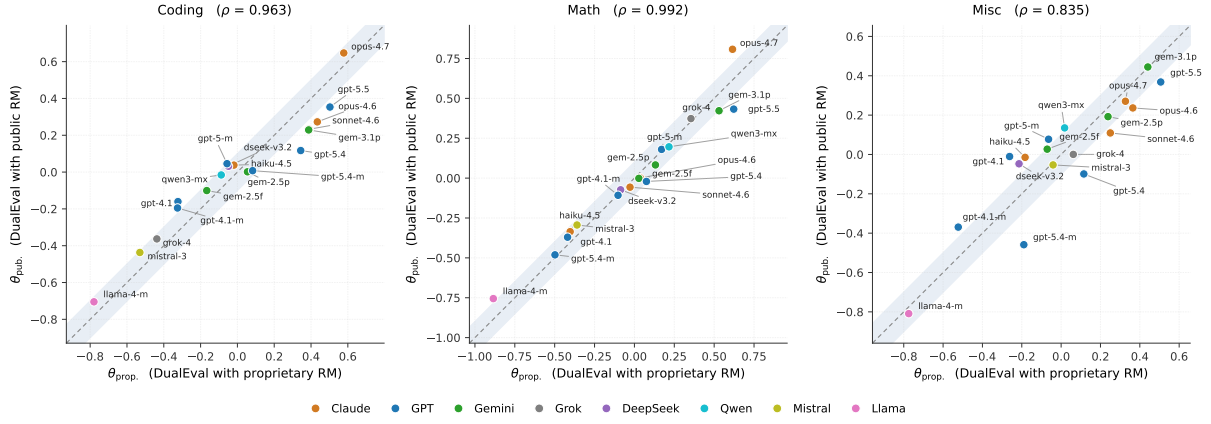


Figure 9: Per-domain scatter of DUALEVAL model abilities under the proprietary RM (x -axis) and the public RM (y -axis), three panels for coding/math/misc. Points are individual frontier LLMs. Static anchoring keeps the cross-RM rankings tightly aligned in static-anchored domains.

Domain	Method	Median $\hat{\sigma}_\theta$	Median rank-CI
Coding	Static 2PL	0.065	3.0
	Arena BT	0.028	1.0
	DualEval	0.040	2.0
Math	Static 2PL	0.074	3.0
	Arena BT	0.039	2.0
	DualEval	0.055	3.0
Misc	Static 2PL	0.082	4.5
	Arena BT	0.026	1.0
	DualEval	0.043	3.0
Generic	Arena BT	0.019	1.0
	DualEval	0.016	1.0

Table 7: Bootstrap uncertainty per (domain, method) under a question-level cluster bootstrap ($B=100$, stratified by source). $\hat{\sigma}_\theta$ is the bootstrap standard deviation of model ability; median rank-CI is the median 95% bootstrap rank interval width over the 18 models. Static 2PL is omitted for Generic (arena-only). On equivalent data (Generic), DualEval matches or improves over Arena BT; against Static 2PL the joint formulation reduces ranking variance by 30–60%.

Evaluation setup. For each (model, question) pair we generate a single response with default sampling settings for each provider’s API as of submission (temperature and top- p are model-default; no system prompt is prepended for arena prompts; a standard “provide a complete, correct solution” instruction is appended for coding/math/misc). Model identifiers are reported in Table 1. For agentic SWE-Bench tasks we route every model through mini-SWE-agent-v2 with a single attempt and the harness’s default rollout budget; for TerminalBench-2.0 we use Terminus2 under identical configuration for all models.

Static grading. Coding tasks are graded by their reference execution harness (test-suite pass for MBPP-Plus and LiveCodeBench; resolved-instance flag for SWE-Bench; harness pass for TerminalBench). Math and miscellaneous items use exact-answer matching after a normalization pass on the final-answer line; ambiguous cases are resolved by an LLM judge with chain-of-thought disabled.

Arena prompts. For coding, math, and miscellaneous we filter LMArena prompts to the target domain via LLM judges using a fixed rubric. The generic domain uses LM Arena’s category-mix without domain filtering. Per-domain arena pool sizes also appear in Table 1.

Reward scoring. Each model response is scored by a single forward pass of the relevant reward model (proprietary or Skywork-Reward-V2-Qwen3-8B for the replication). Rewards are global-standardised per (domain, RM) before pairwise targets are constructed.

I Evaluated Models

Table 9 lists the 18 frontier LLMs evaluated in every domain. All 18 answer every question, producing one response per (model, question) pair.

J Difficulty and Sharpness

Figure 10 plots learned item difficulty b_q against log-sharpness $\log a_q$ for every domain, with points colored by benchmark source. Static items show near-zero or weakly negative cor-

Method	Coding				Math				Misc			
	Pair Acc.	Static ρ	Arena ρ	Public ρ	Pair Acc.	Static ρ	Arena ρ	Public ρ	Pair Acc.	Static ρ	Arena ρ	Public ρ
Arena Avg Reward (prop.)	0.805	0.779	0.994	0.851	0.730	0.571	1.000	0.860	0.806	0.695	1.000	0.874
Arena Avg Reward (pub.)	0.673	0.197	0.311	0.432	0.661	0.445	0.639	0.503	0.777	0.228	0.354	0.222
Arena BT (prop.)	0.809	0.773	1.000	0.851	0.730	0.571	1.000	0.860	0.806	0.695	1.000	0.874
Arena BT (pub.)	0.680	0.255	0.416	0.511	0.667	0.457	0.624	0.536	0.779	0.220	0.350	0.247

Table 8: Arena-only average-reward baseline, with the corresponding Arena BT rows from Table 2 repeated for comparison. Columns and protocol are identical to Table 2.

Provider	Model
OpenAI	GPT-5.5, GPT-5.4, GPT-5.4-mini, GPT-5-mini, GPT-4.1, GPT-4.1-mini
Anthropic	Claude Opus 4.7, Claude Opus 4.6, Claude Sonnet 4.6, Claude Haiku 4.5
Google	Gemini 3.1 Pro, Gemini 2.5 Pro, Gemini 2.5 Flash
xAI	Grok-4
DeepSeek	DeepSeek-V3.2
Mistral	Mistral Large 3
Qwen	Qwen3-Max-Thinking
Meta	LLaMA-4 Maverick Instruct

Table 9: The 18 frontier LLMs evaluated, by provider.

relation in every static-anchored domain ($\rho = -0.00, +0.13, -0.36$ for coding, math, misc), confirming that difficulty and sharpness are largely independent properties of a static item: saturated items can be sharp, hard items can be flat, and Fisher information cannot be inferred from difficulty alone. Arena items, by contrast, show a robust positive coupling ($\rho \in [+0.58, +0.73]$ across all four domains), reflecting how continuous reward signals propagate both location and spread into the IRT parameters. The joint DUALEVAL formulation accommodates both regimes without imposing a single structural assumption on either, and the contrast itself confirms that the framework’s learned parameters reflect the structure of the underlying outcome modality.

K Fisher-Information Expressions

For a static item, $p_{i,q} = \sigma(a_q(\theta_i - b_q))$, where a_q, b_q , and θ_i are the learned item sharpness, difficulty, and model ability. The Fisher information for model i on item q is $\mathcal{I}_{i,q}^{\text{static}} = a_q^2 p_{i,q}(1 - p_{i,q})$, which we average over models and weight by the static loss coefficient:

$$S_q^{\text{static}} = \lambda_{\text{static}} \frac{1}{M} \sum_{i=1}^M a_q^2 p_{i,q}(1 - p_{i,q}).$$

For an arena item, DUALEVAL compares responses through $\mu_{ijq} = \sigma(\gamma(p_{i,q} - p_{j,q}))$. We

score each pair by the trace of the Fisher information with respect to the two involved ability parameters (θ_i, θ_j) , which measures local sensitivity of the predicted preference probability to either model’s ability:

$$\mathcal{I}_{ijq}^{\text{arena}} = \mu_{ijq}(1 - \mu_{ijq})\gamma^2 \cdot \left[(a_q p_{i,q}(1 - p_{i,q}))^2 + (a_q p_{j,q}(1 - p_{j,q}))^2 \right].$$

We average this quantity over decisive arena pairs for item q :

$$S_q^{\text{arena}} = \lambda_{\text{pref}} \frac{1}{|\mathcal{P}_q|} \sum_{(i,j) \in \mathcal{P}_q} \mathcal{I}_{ijq}^{\text{arena}},$$

where $\mathcal{P}_q$ denotes the set of decisive, i.e., non-tie and non-both-bad, model pairs for item q . Static and arena items are therefore scored in the same units: expected information about the model ability vector. Averaging over arena pairs prevents items from receiving an artificial advantage from $O(M^2)$ pairwise comparisons.

L Out-of-Pool Transfer Protocol

For each of the 18 models m and each domain in {coding, math, miscellaneous}, we fit DUALEVAL on the remaining 17 models, producing item parameters (a_q, b_q) and Fisher scores S_q that are independent of m . We then select the top fraction of items by S_q , stratified within each modality so that the static/arena composition of the subset matches that of the full domain, and estimate θ_m on the selected cells with all item parameters held fixed and both loss terms active ($\lambda_{\text{static}} = \lambda_{\text{pref}} = 1$). The transferred rank of m is its position when θ_m is inserted into the frozen 17-model ability ordering; the reference is m ’s rank in the full-data 18-model fit. At the 10% operating point this amounts to 30, 20, and 30 graded static responses for coding, math, and miscellaneous respectively, plus the matching 10% of arena items.

The fusion baselines are computed on exactly the same selected cells: **mean-z fusion** places m

Difficulty vs sharpness across domains. Arena items show consistent positive coupling ($\rho \approx 0.6, -0.7$) in every domain; static items show near-zero coupling.

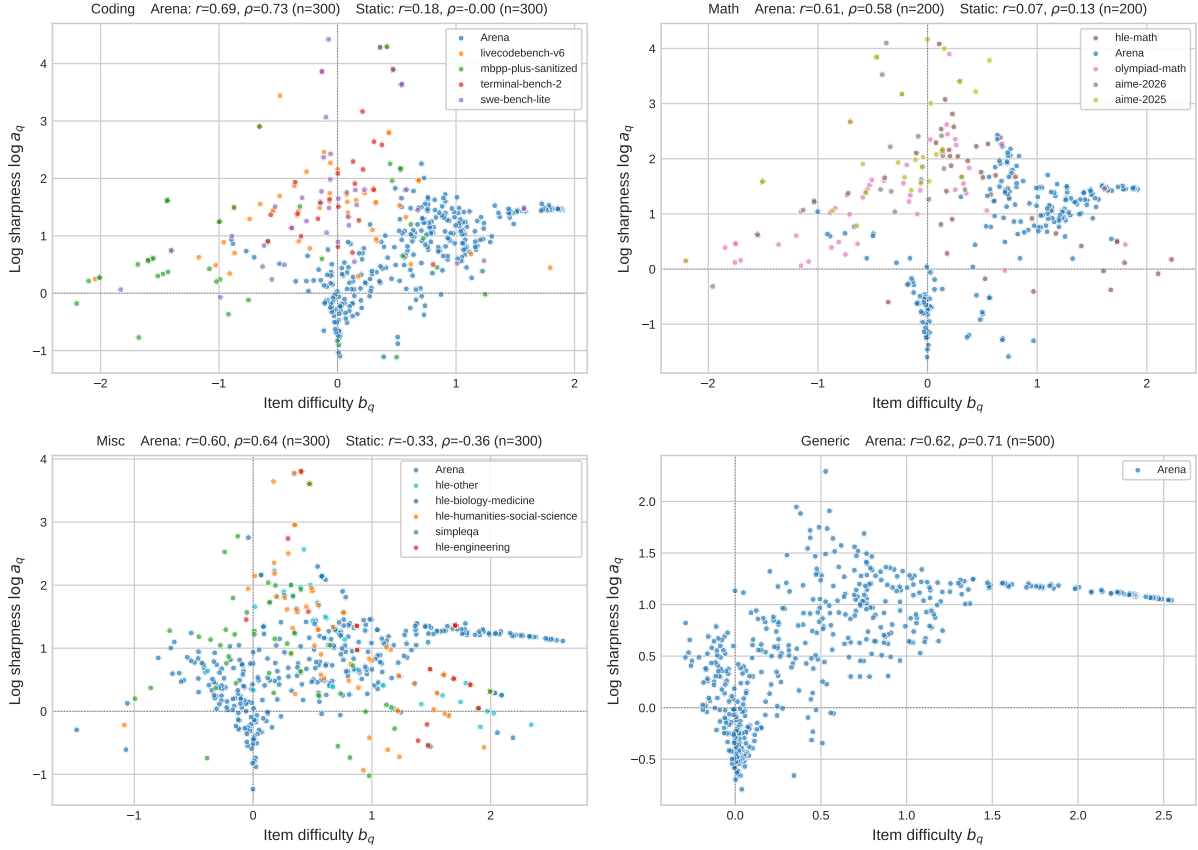


Figure 10: Learned item difficulty versus sharpness across domains. Static items show weak difficulty–sharpness coupling, while arena items exhibit a stronger positive association, highlighting modality-specific item structure captured by DUALEVAL.

by the mean of its z -scored static accuracy and z -scored mean arena reward, and **rank-avg fusion** by the mean of its two per-signal ranks, in both cases with the z -scores and ranks taken over the 18 models. **Refit-from-scratch** re-estimates all model and item parameters on the subset with m included; it is an in-pool reference and, at 100% of items, reproduces the full-data ranking exactly, which validates the harness.

M Pair-Treatment Ablation

We ablate the two pair-treatment components introduced in §3: both-bad (BB) anchoring and tie filtering. All variants are evaluated on the coding domain using the same 20% held-out pair set constructed with the full configuration thresholds, so metrics are directly comparable. The full model uses BB flagging with $\lambda_{bb} = 0.2$ and tie filtering. Arena pair accuracy is measured on held-out decisive pairs; BB-AUC measures whether

$1 - \min(p_i, p_j)$ ranks BB pairs above non-BB pairs.

Table 10 shows that BB anchoring is the dominant choice: removing the BB loss drops BB-AUC from 0.903 to 0.622 while leaving arena pair accuracy near 0.78, and removing BB flagging entirely gives essentially the same degradation. Tie filtering has a smaller, complementary effect — it slightly improves BB calibration when combined with the full model (0.903 vs 0.899) but removing it alone barely changes either metric. The full configuration is used in all main-text experiments.

N Public Leaderboard Rankings

Table 11 lists the public LMArena Text-leaderboard ranks for the 18 models we evaluate, taken from the matched public Arena Text sub-leaderboards used in Table 2 (Coding, Math, Text-Expert, Text-Overall). These are the external reference rankings against which Table 2’s *Public* ρ column is computed. Lower rank is better. Public ranks were re-

Variant	BB Loss	Tie Filter	Arena Pair Acc.	Both-Bad AUC
Full DUALEVAL	✓	✓	0.778	0.903
No both-bad loss	–	✓	0.771	0.622
No both-bad filtering	–	✓	0.774	0.627
No tie filter	✓	–	0.777	0.899
No both-bad / no tie	–	–	0.774	0.628

Table 10: Coding-domain ablation of pair treatment for arena comparisons. Arena pair accuracy is measured on held-out hard pairs. Both-bad AUC measures whether the fitted model truly identifies pairs where both responses receive low rewards.

trieved on the same date the paper’s arena prompts were sampled.

O Robustness to Structured Missingness

We compare DUALEVAL against mean-z and rank-avg fusion under controlled missingness, using semi-synthetic data so that the target ranking is known. Table 12 reports the *MNAR penalty*: the drop in Spearman and Kendall rank agreement with θ^* when moving from MCAR to MNAR at the same mask rate. Higher values indicate greater sensitivity to biased missingness; small negative values reflect sampling noise. Absolute MCAR agreement is Spearman $\rho \approx 0.95$ – 0.98 for every method, which is why the penalty rather than the raw correlation is the informative quantity.

Generator. The semi-synthetic matrix has 18 models $\times$ 2100 items (800 static, 1300 arena) and is fully dense before masking. Model abilities θ^* and item parameters (a_q^*, b_q^*) are taken from a single DUALEVAL fit pooled over all four domains. Every item carries (a_q, b_q) under the shared latent success probability $p_{i,q} = \sigma(a_q(\theta_i^* - b_q))$ of §3, including arena items, over which the pairwise loss is defined; no separate 2PL fit to binary outcomes is involved, which would not be meaningful for continuous rewards. Static outcomes are then drawn as $y_{i,q} \sim \text{Bernoulli}(\sigma(a_q(\theta_i^* - b_q)))$ and arena rewards as $r_{i,q} = a_q^*(\theta_i^* - b_q^*) + \varepsilon$ with $\varepsilon \sim \mathcal{N}(0, \sigma_{\text{sim}}^2)$. A single fixed σ_{sim} is used for every method and in both the MCAR and MNAR arms, so the reported penalty compares the two missingness mechanisms at one common noise level.

Missingness mechanisms. We mask 40% or 60% of cells with seeds 0–2. Under MCAR each cell is dropped independently with probability equal to the target rate. Under MNAR the per-cell drop propensity is

$$w_{i,q} \propto \text{ability_rank}(i) \times \text{age}(s(q)),$$

where $\text{ability_rank}(i)$ is model i ’s rank by true ability θ^* normalized to $[0, 1]$, so the strongest model takes the value 1.0, and $\text{age}(s)$ is a linear ramp over the evaluation sources ordered from oldest to newest, $\text{age}(s_i) = 1 - i/(n_{\text{sources}} - 1)$, so the oldest source takes the value 1.0 and the newest takes 0. Given a target rate τ , we choose c_τ numerically so that $\frac{1}{MQ} \sum_{i,q} \min\{1, c_\tau w_{i,q}\} = \tau$, and use $\min\{1, c_\tau w_{i,q}\}$ as the drop probability, so the achieved rate matches τ despite the clipping. The mechanism concentrates missingness on (strong model, old source) cells, reproducing the pattern in which newer frontier models are not run on older saturated suites. Ability enters monotonically, so the mechanism does not model the converse pattern of weak models being absent from the hardest suites.

P Contamination-Detection Metric Details

For each domain, modality, injection rate, and random seed, the synthetic contamination protocol defines a binary mask $m_{i,q} \in \{0, 1\}$ over model–item cells, where $m_{i,q} = 1$ denotes an injected cell. Each detector assigns a scalar anomaly score $s_{i,q}$ to every cell using the residual scores defined in §4.3. We then evaluate whether these scores rank injected cells above non-injected cells.

Baseline detectors. Each baseline assigns a scalar suspicion score per cell; their definitions and any information supplied from the joint reference are specified below. *Acc-based* scores a static cell under an independence null: with $p_0 = \text{acc}(m) \times \text{pass_rate}(q)$, the product of model m ’s overall accuracy and item q ’s pass rate across models, the score is $z = (y_{m,q} - p_0) / \sqrt{p_0(1 - p_0)}$, which captures an unexpected solve without fitting latent ability or item discrimination. *Item-LOO* compares model m ’s outcome on item q with the mean outcome of the other 17 models on the same item, again as a z -scored residual, flagging devia-

Paper model	Text-Coding	Text-Math	Text-Expert	Text-Overall
claude-opus-4-6	4	5	2	3
claude-opus-4-7	2	9	3	4
gemini-3.1-pro	10	7	8	6
gpt-5.5	29	4	9	11
claude-sonnet-4-6	11	28	12	22
gpt-5.4	21	33	15	24
gpt-5.4-mini	38	53	27	39
gemini-2.5-pro	81	46	56	49
qwen3-max-thinking	60	52	50	62
deepseek-v3.2	76	63	71	76
mistral-large-3	75	110	105	93
gpt-4.1	97	146	118	94
gemini-2.5-flash	140	99	100	98
claude-haiku-4-5	64	125	69	99
grok-4	121	68	92	100
gpt-5-mini	134	101	122	124
gpt-4.1-mini	127	163	143	135
llama-4-maverick-instruct	186	180	181	197

Table 11: Public Arena ranks for the 18 models used in our external validation analysis. Ranks are taken from the matched public Arena Text subleaderboards used in Table 2: Text-Coding, Text-Math, Text-Expert, and Text-Overall. Lower rank is better.

Method	MNAR penalty (40% / 60% missing)	
	Spearman	Kendall
DUALEVAL	0.016 / -0.005	0.031 / 0.013
Mean-z fusion	0.105 / 0.094	0.139 / 0.139
Rank-avg fusion	0.098 / 0.118	0.122 / 0.161

Table 12: Sensitivity to ability-correlated missingness. Values report the drop in rank agreement from MCAR to MNAR at matched missingness rates; lower is better. Small negative values reflect sampling variation.

tions from an item’s majority pattern without using cross-item structure. *Difficulty-only* scores a static success by the item’s empirical failure rate over the other models, with no per-model parameters at all. *Arena-2PL* is an arena-only IRT residual: item parameters are fit to the arena pairs alone, with no static signal, and the residual is then formed as in §4.3. *BT* is a Bradley–Terry residual with model-level pairwise abilities and no item parameters, so a cell is scored by the gap between its observed pairwise outcomes and those predicted from the two models’ BT scores. *Raw reward* scores an arena cell by its standardized reward directly.

Placing the held-out model for the baselines. No baseline calibrates its reference bank using data from the held-out model. Every reference bank—the joint fit, static-only 2PL fit, arena-only 2PL fit, and Bradley–Terry fit—is estimated on the other 17 models and therefore has no entry for m . Every reference bank — the joint fit, the static-only 2PL fit, the arena-only 2PL fit, and the

Bradley–Terry fit — is estimated on the other 17 models and has no entry for m . To place m on each of these scales we compute one ability estimate against the frozen joint bank from m ’s *clean*, *pre-injection* arena pairs, with the static loss disabled ($\lambda_{\text{static}} = 0$), and then map it affinely onto each baseline’s own scale, yielding the abilities consumed by the static, Arena-2PL and Bradley–Terry detectors. These quantities are precomputed once per domain before any injection, so no contaminated cell enters any baseline ability estimate and every reference bank remains out of pool. Supplying the baselines with a common, uncontaminated ability estimate isolates the residual definition from the quality of the ability estimator, and is favorable to the baselines, which would otherwise have to estimate ability from contaminated cells.

In-pool targeted injection. For static labels we sample cells from the low-probability success region – failed cells whose DUALEVAL success probability is smallest – and flip $y_{i,q}$ from 0 to 1. For arena responses we sample cells in the bottom 30% of the standardized-reward distribution and add 2.0 in standardized-reward units to $z_{i,q}$, then rebuild the soft pairwise targets p_{ijq}^* from the perturbed rewards. Injection rates are $\{0.5, 1, 2, 5\}\%$ with five random seeds per rate, and all 18 models remain in the calibration pool.

Out-of-pool screening. For each held-out model m we refit the clean reference on the remaining 17 models, freeze the resulting item bank, and

estimate θ_m from the modality that is not being screened: the static loss is disabled ($\lambda_{\text{static}} = 0$) when static cells are screened, so θ_m comes from m 's clean arena pairs alone, and the arena loss is disabled ($\lambda_{\text{pref}} = 0$) when arena cells are screened, so θ_m comes from m 's clean static responses alone. Contaminated cells therefore never enter the ability estimate. Injection uses one deterministic draw per (model, rate) rather than seed averaging, and the three domains are pooled within each rate. As a safeguard, folds are dropped when a held-out model has fewer than five eligible cells or when the fit does not converge; in practice no folds were dropped, and all 18 contribute at every rate and on both modalities.

Response-level substitution. For a static cell (i, q) we select a donor model that answered q correctly, substitute its response verbatim as i 's response, and re-grade with the unchanged judging pipeline; the label becomes $y_{i,q} = 1$ only if the judge accepts the substituted response. Across math and miscellaneous the judge rejects 5–9% of these substitutions, which therefore remain labelled $y_{i,q} = 0$ despite being injected – a grading-drop rate that has no analogue in a label-flip protocol, and the sense in which the positive class here is not true by construction. For an arena cell we select a donor from the top quartile by standardized reward on the same prompt, substitute its response, and reuse that response's already-computed reward. Because reward scoring is a single deterministic forward pass over the (prompt, response) pair (Appendix §H) and the substituted pair is byte-identical to the donor's, re-running the reward model would return the same value; the lookup is an exact substitute for re-scoring rather than an approximation. The induced shift is heterogeneous, with mean $\Delta z \approx 0.7$ – 1.3 and occasional non-positive values, in contrast to the fixed $+2.0$ offset of the in-pool protocol. Items that no model solved admit no donor and are excluded as non-contaminatable. Coding is excluded from this protocol entirely: its correctness labels come from execution harnesses rather than the LLM judge, so a verbatim-substituted passing solution would pass by definition and the protocol would reduce to exact label flipping.

AUROC is computed as the probability that a randomly selected injected cell receives a higher score than a randomly selected non-injected cell,

with ties counted halfway:

$$\text{AUROC} = \Pr(s^+ > s^-) + \frac{1}{2} \Pr(s^+ = s^-),$$

where s^+ and s^- denote scores from injected and non-injected cells, respectively.

AUPRC is the area under the precision–recall curve obtained by thresholding the anomaly scores. For a threshold t , precision and recall are

$$\text{Prec}(t) = \frac{\sum_{i,q} \mathbf{1}[s_{i,q} \geq t] m_{i,q}}{\sum_{i,q} \mathbf{1}[s_{i,q} \geq t]},$$

$$\text{Rec}(t) = \frac{\sum_{i,q} \mathbf{1}[s_{i,q} \geq t] m_{i,q}}{\sum_{i,q} m_{i,q}}.$$

We compute the precision–recall curve over all score thresholds and report the area under this curve. Both AUROC and AUPRC are computed at the cell level. Averaging differs by protocol: for the in-pool targeted injection and the response-level substitution, values are averaged over the five random injection seeds at each rate and modality; the out-of-pool screening protocol uses a single deterministic draw per (model, rate) and is instead averaged over the leave-one-model-out folds.

Q In-Pool Contamination Detection Curves

Figure 11 provides the complete domain- and injection-rate-specific results for the in-pool targeted-contamination experiment. The clean-reference detector scores corrupted outcomes against the original uncontaminated fit, whereas the more realistic self-anchored detector is refit after contamination and may therefore absorb part of the injected signal. Across domains and modalities, the clean-reference detector performs best, while the self-anchored detector remains consistently above the random baseline. AUPRC generally increases with the injection rate partly because contaminated cells become less rare, so comparisons between methods are most meaningful at matched rates.

R Examples

Item-Profile Gallery. Table 13 walks the four corners of the (difficulty b_q , sharpness a_q) plane on coding-domain static items. The example items are drawn directly from the full-data DUALEVAL fit; pass counts are observed across the 18 evaluated models. This gallery illustrates qualitatively what the Fisher subset experiment in §4.2 measures

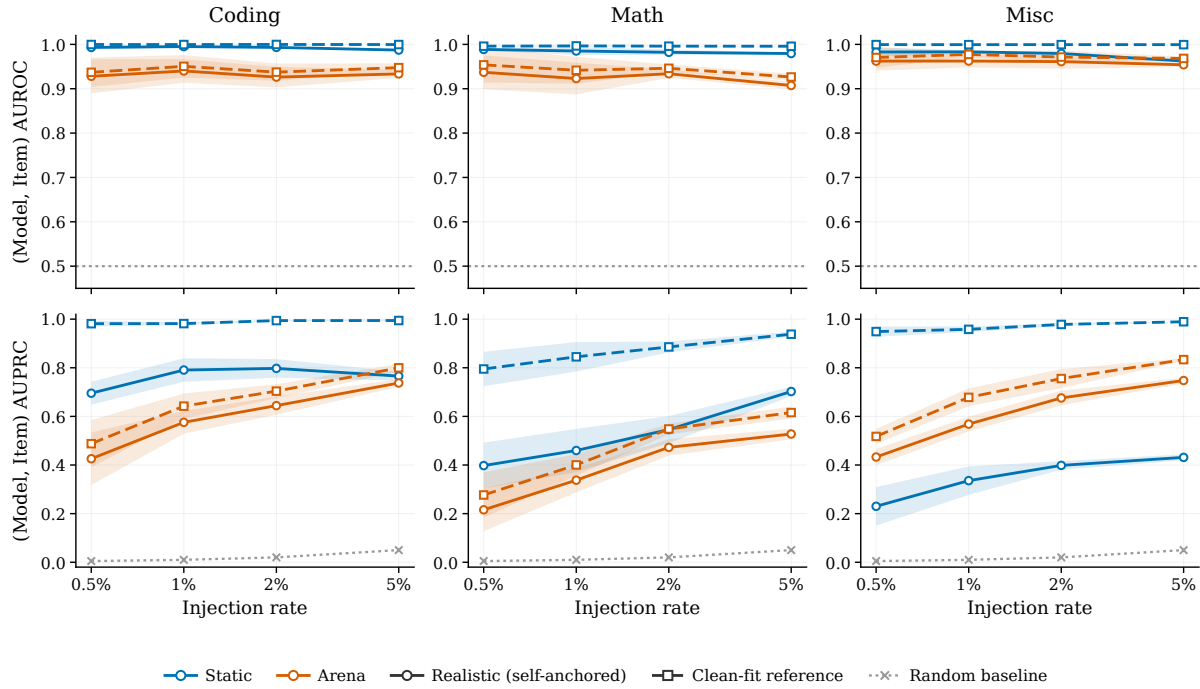


Figure 11: In-pool synthetic contamination detection across coding, math, and miscellaneous domains, separately for static-label and arena-reward injections. Each panel reports model–item cell-level AUROC/AUPRC versus injection rate; the positive class is the set of synthetically contaminated cells. The clean-reference detector is strongest across domains and injection rates, while the contaminated-fit detector remains above the random baseline. All 18 models are in the calibration pool here; Tables 4 and 5 report the stricter out-of-pool protocols.

quantitatively: ranking signal lives in items where both a_q is high and the pass rate is between the extremes, and items in the saturated and uniformly hard corners contribute essentially no ranking information.

Anomaly Candidates. Table 14 lists representative anomaly candidates surfaced by the standardized DUALEVAL residual diagnostic of §4.3, restricted to natural (un-perturbed) responses with $|r_{i,q}| > 3.5\sigma$. These are candidates, not verified contamination: large residuals can also reflect MC guessing on items with few options, judge artifacts, or genuine but narrow capability not captured by the joint fit. Semantic-preserving perturbation audits are the appropriate follow-up to distinguish memorization from robust capability and are left to future work. The patterns observed across the three picks (number-sequence puzzle archives, specialized MC questions, named MBPP function signatures) are nevertheless consistent with prior exposure as a plausible alternative hypothesis.

S Artifact Licenses and Terms of Use

Our study uses public benchmarks, arena-style prompts, model outputs, and reward-model scores

for research evaluation purposes. We follow the licenses and terms of use specified by the original providers of all third-party artifacts. We do not redistribute third-party datasets, proprietary model outputs, reward-model weights, reward-model training data, or any artifacts whose redistribution is restricted.

Our primary arena supervision uses a proprietary, non-public scalar reward model. We describe only high-level information about its architecture, training objective, and training-data scale; we do not release its weights, training data, or associated proprietary data. The reward model is used only to produce scalar response-quality scores for analysis. Any released code, aggregate statistics, or derived metadata will be distributed with an explicit license and will exclude non-redistributable artifacts.

T Use of AI Assistants

Generative AI tools were used to assist with language polishing and the generation and debugging of experimental code. All research decisions, implementations, analyses, and reported results were reviewed and verified by the authors, who take full responsibility for the contents of this paper.

Item	Pass / a_q
A. Frontier-discriminative. LCB-v6 “Stone XOR”. “There are N bags labeled $1, \dots, N$. Bag i contains A_i stones. Choose two bags A, B , and move all stones from A to B ...” <i>Hard but very sharp</i> ($b=0.36$, $a_q=72.7$). <i>Only 5 of 18 models pass; the joint fit identifies them as systematically the strongest — large Fisher information for ranking the frontier.</i>	5/18 $a_q=72.7$
B. Uniformly hard. LCB-v6 “Min of Restricted Sum”. “You are given integers N, M and three integer sequences of length M ...” <i>Hardest tail of the difficulty distribution</i> ($b=1.57$); <i>0 of 18 models pass. The item provides no ranking signal — no variation in outcomes to attribute to ability differences.</i>	0/18 $a_q=4.4$
C. Borderline-discriminative. LCB-v6 “Unique 3-Digit Even Numbers”. “You are given an array of digits. Determine the number of distinct three-digit even numbers that can be formed using these digits...” <i>Easy</i> ($b=-0.48$) <i>but sharp</i> ($a_q=31.2$). <i>16 of 18 models pass; the 2 misses are the two weakest. Informative for separating trailing models, not the frontier.</i>	16/18 $a_q=31.2$
D. Saturated. MBPP-Plus “left insertion point”. “Write a Python function to locate the left insertion point for a specified value in sorted order...” <i>Trivially easy</i> ($b=-1.44$), <i>18 of 18 models pass. Contributes nothing to ranking; a candidate for retirement under benchmark refresh.</i>	18/18 $a_q=5.0$

Table 13: Item-profile gallery (coding domain) walking the four corners of the (difficulty b_q , discrimination a_q) plane. Pass counts are out of the 18 evaluated models. Frontier-discriminative items (A) carry the bulk of ranking signal; uniformly hard (B) and saturated (D) items contribute none and are candidates for retirement; borderline-discriminative items (C) inform only the low-ability tail. The Fisher-information subset experiment (Fig. 2) operationalizes this picture quantitatively.

Anomaly candidate	$\hat{p} / r$
HLE-Math / gpt-4.1 (rank 16/18). “2, 11, 23, 51, 119, () A. 291 B. 285 C. 171 D. 167.” <i>MC sequence-completion item; the pattern is widely archived (puzzle banks/OEIS) and a weak math model recovering the correct option is consistent with prior exposure.</i>	0.04 +4.81
HLE-Bio/Med / llama-4-maverick-instruct (18/18). “Which genetic disorder caused by mutations on chromosome 2 leads to the greatest increases to patients’ basal metabolic rate? A. Alström B. Menkes...” <i>Specialized medical MC. The lowest-ranked model passing where frontier models fail is $> 5\sigma$; consistent with recall from medical-Q&A corpora.</i>	0.03 +5.36
MBPP-Plus / llama-4-maverick-instruct (18/18). “Write a python function to find the sum of the per-digit difference between two integers. Your function must be named digit_distance_nums.” <i>MBPP is among the most-trained-on coding benchmarks; the named signature suggests recall rather than synthesis from the spec.</i>	0.07 +3.57

Table 14: Anomaly candidates surfaced by standardized DUALEVAL residuals $r_{i,q} = (y_{i,q} - \hat{p}_{i,q}) / \sqrt{\hat{p}_{i,q}(1 - \hat{p}_{i,q})}$ on un-perturbed responses ($y_{i,q} = 1$ for all rows). These are candidates, not verified contamination: large residuals can also reflect multiple-choice guessing or judge artifacts. Perturbation audits are the appropriate follow-up to distinguish memorization from robust capability.