When Should AI Evaluation Measure More?

Anonymous authors
Paper under double-blind review

Abstract

When an evaluation result is uncertain, the usual remedy is more of the same evaluation. We argue that evaluation is instead a target-conditioned design problem: an evaluator can change what is measured, how the responses are analysed, or which target and reporting rule the evidence is meant to support, and these levers should be compared at matched cost. Across vision and language-model response banks, we find that the useful action depends on the bottleneck. In one matched-cost vision comparison, the same added budget reduced target error from 1.10 to 1.05 percentage points when spent on deeper replication of existing conditions, but to 0.56 when spent on target-relevant joint conditions. More broadly, reanalysis can outperform substantially more calls under a plug-in analysis for lower-tail targets but not for mean targets, and a rule accurate for a scalar summary can fail on the condition-level responses behind it. In a held-out check, a comparison frozen on development models selected released estimators that reduced equal-query error on held-out models when a historical calibration bank was available. We formalize the distinction between measurement support, statistical inference and decision targets, and provide Reassess, a toolkit for comparing these alternatives from response-level banks. The practical question is therefore not only how much more to evaluate, but what the next unit of evaluation should change.

1 Introduction

Evaluation budgets are usually spent on more of the same: more items, more prompts, more seeds. That habit treats three different problems as one. An evaluation can lack *support*, never observing the conditions its target depends on. It can observe them and still *infer* the target poorly from the responses. Or it can answer a different question from the one the decision needs. A falling error bar cannot tell these apart. More items reduce sampling noise whether or not the measured conditions determine the target, and a summary that is accurate for one target can be wrong for another.

Consider 47 ResNet-50 checkpoints whose accuracy under combined blur and JPEG compression we want to know, starting from their accuracy on clean images and under each corruption alone. About 16,000 more forward passes per model on the single-corruption conditions moved the error of this joint-corruption target from 1.10 to 1.05 percentage points (pp). The same passes on four joint conditions reached 0.56 pp, and spread over all 16 target conditions they reached 0.38 pp (Fig. 1, left). The budget was the same; what changed was which part of the response surface it observed.

We treat evaluation as a target-conditioned design problem (Fig. 1). Given an evaluation context, the responses already collected and a query cost, an evaluator can pull three levers: *measure* new support rather than replicate old support, *infer* a target better from the responses in hand, or *decide* which target and reporting rule the evidence should support. The levers rest on a classical distinction. Whether a set of measurements determines a target is an identification question (Manski, 2003; Micchelli & Rivlin, 1977); how precisely a determined target is estimated from finite responses is a statistical one. This distinction determines the matched-cost comparison: change the measured support, change the inference, or change the target and reporting rule.

New support beat replication when the target depended on unmeasured joint conditions, although on a second corruption pair the gain was smaller and missed its prespecified criterion (Section 4). Reanalysis beat four times the plug-in calls when the responses already covered the target, while its advantage over a structured estimator was unresolved; on public banks it helped tail targets and never the mean (Section 5). The best rule reversed when the target changed from a scalar summary to the

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

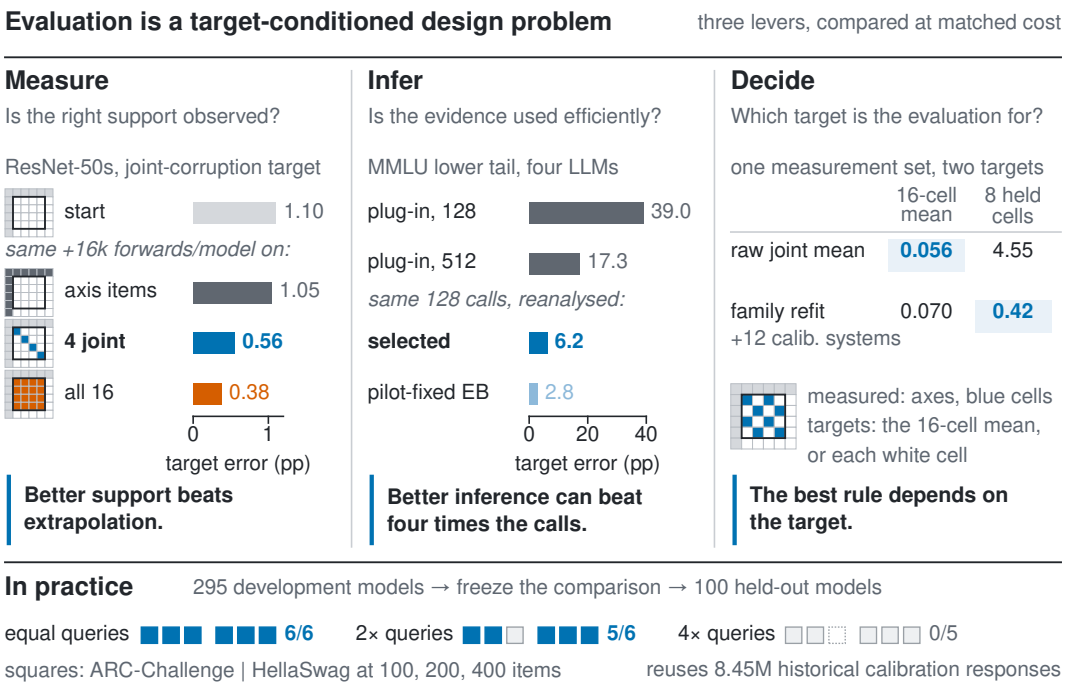

Figure 1: **Evaluation is a target-conditioned design problem.** Three levers address different evaluation bottlenecks: measure new support, infer the target better from the responses in hand, or decide which target and reporting rule the evidence should support. Each column shows one representative matched-cost result, as absolute target error (pp) against a dense reference (Sections 4–6; "all 16" is a post-hoc arm, "selected" uses only the 128 acquired responses, and empirical-Bayes (EB) shrinkage was fixed on a pilot split). The footer is the held-out check of Section 7: filled squares mark a resolved error reduction by the chosen estimator over direct sampling (dotted: $4n$ exceeds the ARC-Challenge pool).

cells behind it (Section 6). A comparison frozen on development models chose released estimators that reduced equal-query error on 100 held-out models, given a reused historical calibration bank (Section 7).

Measurement, inference and reporting are alternative interventions whose value depends on the target and its cost. Reassess compares them from response-level banks while accounting separately for query and calibration information. The component estimators, samplers and stopping rules are established; the useful intervention is not. Before spending more evaluation budget, ask whether the measurements, the estimator or the target needs to change.

2 Related Work

Measurement science for AI evaluation. Validity-centred frameworks ask which claim a measurement supports (Salaudeen et al., 2025; Truong & Koyejo, 2026), and NIST AI 800-3 argues for explicitly defined estimands, statistical models and uncertainty quantification (Keller et al., 2026). Studies of what benchmarks measure (Desai et al., 2026; Hardy et al., 2026; Zhou et al., 2026), interval reporting (Agarwal et al., 2021), benchmark selection effects (Dehghani et al., 2021; Zhang & Hardt, 2024; Alzahrani et al., 2024) and underspecification (D'Amour et al., 2022) all motivate declaring the target before reading a score. SCORE documents sensitivity to prompt paraphrase and answer-option order (Nalbandyan et al., 2025), and FormatSpread to prompt formatting (Sclar et al., 2024). These studies motivate careful specification of what a reported score represents. Controlled-response evaluation has also been used to map conditional detector behavior that aggregate chest-radiograph scores leave unresolved (Anonymous, 2026b). A separate question is what the next unit of evaluation effort should change: the measured conditions, the estimator, or the target and report-

ing rule. The formal tools are optimal recovery of functionals from partial information (Micchelli & Rivlin, 1977; Donoho, 1994), partial identification (Manski, 2003), anchored decompositions (Kuo et al., 2010) and optimal design (Kiefer & Wolfowitz, 1960; Pukelsheim, 1993); Khanbayov et al. (2026) give identifiability results for a log-linear perturbation-response model of AI systems. A contemporaneous study of vision-language-action policies similarly maps local response behavior and disturbance tolerance under controlled off-nominal probes (Anonymous, 2026a).

Efficient evaluation. Efficient-evaluation methods differ in the structure they exploit. Active testing chooses and reweights test points to estimate risk from few labels (Kossen et al., 2021). Several methods learn from historical response matrices of earlier models: tinyBenchmarks fits item-response-theory (IRT) models and predicts full-benchmark accuracy from a small anchor set (Maia Polo et al., 2024a); REEval amortizes item-difficulty estimation from item content and uses it for adaptive testing (Truong et al., 2025); and FAQ combines a Bayesian factor model with active question selection and inference that retains frequentist coverage (Wu et al., 2026). SparseEval chooses and weights anchor items by sparse optimization, and DISCO selects the items on which models disagree most (Zhang et al., 2026; Rubinstein et al., 2026). PromptEval estimates the distribution of performance over prompts, including quantiles, with Rasch-type models and balanced sampling (Maia Polo et al., 2024b); shrinkage baselines go back to empirical Bayes (Efron & Morris, 1975). Each method makes acquisition, inference or both more efficient for a given target. We compare such levers against one another under a declared target and matched cost, and use existing methods as components: PromptEval is our public-bank case (Section 5) and tinyBenchmarks supplies the estimators of the held-out check (Section 7).

Adaptive decisions. Fixed-budget best-arm identification (Audibert & Bubeck, 2010; Karnin et al., 2013), fixed-confidence elimination (Even-Dar et al., 2006; Kaufmann et al., 2016), group-sequential designs (Jennison & Turnbull, 1999), confidence sequences (Howard et al., 2021; Waudby-Smith & Ramdas, 2024) and subset selection with common random numbers (Gupta, 1965; Nelson & Matejcik, 1995; Kim & Nelson, 2001) provide selection and stopping rules, several with formal guarantees, and TRIPLE treats prompt selection as budgeted best-arm identification (Shi et al., 2024). We use such rules to price the trade-off between cost, wrong winners and set size when the target is a choice rather than a score.

3 Evaluation as a target-conditioned design problem

Objects. Fix a system, an item population and a scoring rule. For a controlled condition z (a corruption severity pair, a prompt format), $\theta(z)$ is the expected score under z, so θ is a mean response surface; Θ is a declared class of such surfaces. An evaluation design fixes a measurement map $B(\theta)$, for example the vector of $\theta(z)$ on the measured conditions; the scientific question fixes a scalar target $T(\theta)$, for example the mean of θ over a set of conditions; and an estimator ψ turns finite-sample measurements $y \approx B(\theta)$ into $\widehat{T} = \psi(y)$. The three levers act on different objects: measuring changes B, inferring changes ψ, and deciding specifies the target T and, for choice problems, the reporting rule. Buying more of the same items only tightens y around $B(\theta)$.

Compatible set and target width. For an observation y and a tolerance $\delta \geq 0$ in a fixed norm,

$$\mathcal{C}(y,\delta) = \{\theta \in \Theta : \|B(\theta) - y\| \leq \delta\}, \qquad D_T(y,\delta) = \sup_{\theta \in \mathcal{C}(y,\delta)} T(\theta) - \inf_{\theta \in \mathcal{C}(y,\delta)} T(\theta). \quad (1)$$

$\mathcal{C}$ contains every admissible surface that agrees with the measurements up to δ, and D_T is the spread of the target over those surfaces (defined when $\mathcal{C}$ is nonempty and T is bounded on it; an empty set means the observation is incompatible with Θ). $D_T(y,0) > 0$ means that two admissible surfaces produce exactly the same measurements but different targets: no precision on the measured coordinates separates them.

Identification and recovery. Two classical facts make this precise (Micchelli & Rivlin, 1977; Donoho, 1994; Manski, 2003) (proofs in Appendix A).

(P1) *Identification.* T is identified by B on Θ if and only if T is constant on every fiber $\{\theta \in \Theta : B(\theta) = \beta\}$, equivalently $T = g \circ B$ for some function g on $B(\Theta)$.

(P2) *Scalar midpoint recovery.* If $\mathcal{C}$ is nonempty and the target image $T(\mathcal{C})$ has finite infimum L and supremum U, then

$$\inf_{a\in\mathbb{R}} \sup_{\theta\in\mathcal{C}} |a - T(\theta)| = \tfrac{1}{2}(U - L), \tag{2}$$

attained by the midpoint $a = (L + U)/2$.

Half the width is thus the best worst-case error of *any* report that knows only $\theta \in \mathcal{C}$. It is a property of the admissible class and the design, not a confidence interval, and it says nothing about the statistical efficiency of a fitted ψ. Repeating the same measurements can reduce the finite-sample tolerance δ, but the width then decreases only to the exact fiber width $D_T(B(\theta), 0)$ (Lemma A.3, for compact Θ and continuous B, T). New measurement coordinates can shrink the compatible set itself, and added constraints never increase the width (Proposition A.4).

Two factors: what the axes miss. For conditions indexed by two factor levels (u, v), with $(0, 0)$ the clean condition, write the anchored decomposition (Kuo et al., 2010)

$$\theta(u, v) = \underbrace{\theta(u, 0) + \theta(0, v) - \theta(0, 0)}_{\theta^{\mathrm{add}}(u,v)} + h(u, v), \qquad h(u, 0) = h(0, v) = 0. \tag{3}$$

For a target that averages the joint cells with weights q (uniform on the 4×4 cells of Section 4),

$$T_q(\theta) = \sum_{u,v} q_{uv}\, \theta^{\mathrm{add}}(u, v) + \sum_{u,v} q_{uv}\, h(u, v). \tag{4}$$

The first term is a function of the axis measurements; the second, the weighted joint residual, is not. Hence (Proposition A.5): if Θ forces $h \equiv 0$, the axes identify T_q; if Θ allows $|h| \le \eta$ on the target cells, the exact width from the axes is at most 2η, with equality when the $[0, 1]$ bounds do not bind; on the unrestricted class the axes leave the whole joint mass undetermined. Joint cells reveal h where they are measured. If the residual is constant over the target cells, one joint cell identifies T_q, which is the assumption under which the axis-adjusted mean of Section 4 is exact; measuring every target cell identifies T_q on any class. The three vision actions therefore rest on three structural assumptions: axis separability ($h \equiv 0$, or its multiplicative analogue for the product rule), a constant residual, and none.

A scalar is not a surface. (P1) applies coordinate by coordinate. A scalar functional can be identified while the surface is not: if only $\theta_1 + \theta_2$ is measured, the sum is identified and neither coordinate is. Recovering one declared target from a measurement set implies nothing about recovering a richer target, such as each unmeasured cell, from the same set.

From width to experiments. The class of real response surfaces is unknown, so no experiment can compute D_T. The experiments measure the error of stated estimators against a dense finite reference under stated costs; an empirical error is neither an identified-set width nor a lower bound over estimators. The theory therefore induces three empirical contrasts: new support versus replication (Section 4), better inference versus more responses (Section 5), and one target or reporting rule versus another on the same measurements (Section 6).

Protocol common to all studies. Errors are absolute target errors in pp, averaged over systems, against a dense reference computed on the same bank. For our MMLU (Massive Multitask Language Understanding) bank (Hendrycks et al., 2021) the reference items are disjoint from the queryable items; for the vision and PromptEval banks the reference is the full finite bank and contains the queried subset. The cost unit is one model call on one (item, condition) pair; estimator fitting is not charged. Arms within a comparison share allocation draws, bootstrap resamples and the reference. *Retrospective* comparisons fix their arms before any action-comparison output but use banks that earlier analyses had examined; *prespecified* comparisons fix rule, budget and success criterion before the responses exist; *post hoc* marks analyses added after seeing earlier outputs; *secondary* marks released public banks. Intervals are pointwise paired 95% percentile bootstrap intervals, conditional on the fixed system panels, and treat items within a stratum as exchangeable (Appendix B.2).

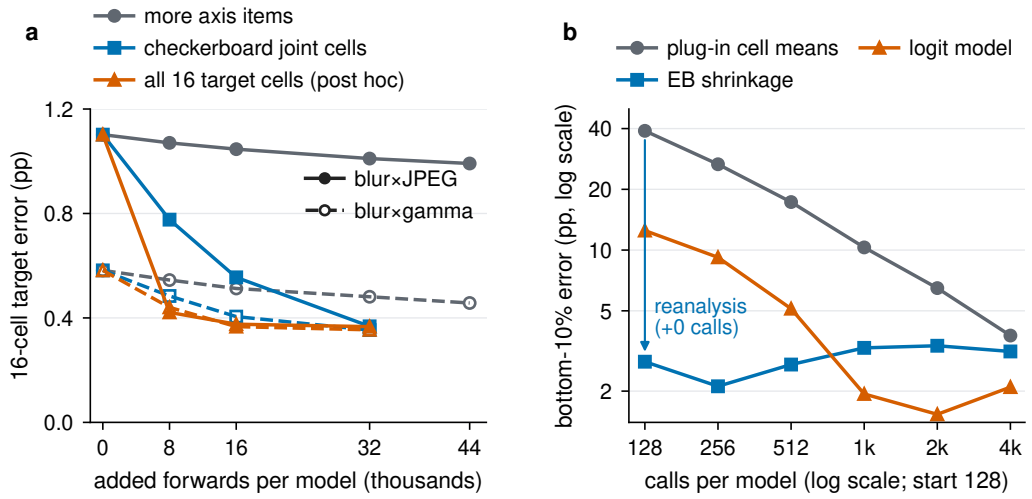

Figure 2: **The same added budget buys different error reductions.** (a) Vision, 47 ResNet-50 checkpoints: error of the mean accuracy over the 4×4 grid of joint-corruption severities, starting from the 11 axis conditions on 4 items per class. Gray: added forwards buy more items under the axis conditions (product rule; the last point uses all items). Blue: the same forwards measure 2, 4 or 8 cells of a fixed checkerboard of the target grid (axis-adjusted mean). Orange (post hoc): the same forwards spread over all 16 target cells (direct mean). Solid, blur×JPEG (retrospective; 200 allocations); dashed, blur×gamma (prespecified; criterion not met; 1,000 allocations). (b) MMLU, mean of the lowest 10% of 48 prompt conditions, four models, against a disjoint 4,096-item reference (retrospective; 50 allocations). The arrow is reanalysis of the 128-call responses; against the structured logit comparator (orange) its advantage over four times the calls was unresolved.

4 Measure: new support versus replication

Design. Forty-seven ResNet-50 checkpoints from ten training-recipe blocks (Gavrikov & Keuper, 2024) were evaluated on a 6×6 grid of Gaussian blur and JPEG severities (level 0 is clean) on 8,000 ImageNet-V2 images, eight per class (Recht et al., 2019). The target is the mean accuracy over the 4×4 grid $\mathcal{T}$ of joint severities at levels 1–4 (Appendix B). The start measures the 11 axis cells (clean and single corruptions) on four items per class, about 44,000 forwards per model. Four actions spend the same added budget, about 16,000 forwards per model. *Replicate the axes*: more items under the 11 axis cells, predicted by a product rule. *Replicate the relevant axes*: the same, restricted to the 9 axis cells that enter the estimator. *Checkerboard joint cells*: $k = 4$ cells drawn from a fixed checkerboard $\mathcal{S} \subset \mathcal{T}$ of eight cells, on the starting items, combined with the axes by the axis-adjusted mean. *Target cells*: the budget spread over all 16 cells of $\mathcal{T}$, averaged directly. The first and third actions form the original, retrospective comparison; the second and fourth are post-hoc arms on the same cached banks, start states and resampling. In the terms of Proposition A.5, the actions rely on axis separability, a constant residual and no structure.

Target coverage beat deeper replication on blur×JPEG. On blur×JPEG, 16,000 more forwards on axis items moved the error from 1.10 to 1.05 pp, and the axes on all 8,000 items still left 0.99 pp (Fig. 2a). Restricting replication to the nine relevant axis cells gave 1.03 pp (difference -0.004 $[-0.022, 0.005]$ pp), so the plateau is not budget wasted on irrelevant conditions; it is consistent with the unmeasured residual of Eq. (4) under the tested axis-only rules. Four checkerboard cells reached 0.56 pp, a paired difference of -0.49 pp from axis replication, favoured by 97.5% of allocations and 8 of 10 recipe blocks. The same forwards on all 16 target cells reached 0.38 pp (-0.17 $[-0.24, -0.14]$ pp relative to the checkerboard; 10 of 10 blocks; Table 4). Reanalysing the axes did not substitute for new support: switching from the additive to the product rule moved the error by 0.08 pp, and a prespecified cross-model ridge head on the axes was worse (Appendix C).

The gain was smaller and heterogeneous on blur×gamma. The comparison was fixed before any new response was generated: blur×gamma on the same checkpoints, items, grid indexing, checkerboard rule and cost unit (Appendix D). Success required a negative mean difference, a negative upper bound of the class-stratified 95% interval, and a gain in at least 6 of the 10 recipe blocks. The difference was -0.108 pp (0.41 vs 0.51 pp) with interval $[-0.205, -0.072]$, and 76% of allocations favoured the checkerboard, but only 5 of 10 blocks gained, so the criterion was not met (Table 3). In the post-hoc arms, the 16 target cells improved on the checkerboard by only -0.04 $[-0.07, -0.03]$ pp (0.37 pp) and beat axis replication in 5 of 10 blocks. The axes already predict blur×gamma well (0.46 pp with all items, half the blur×JPEG value), which may leave less residual to recover. On both pairs, measuring the target cells was on average the strongest use of the budget when they can be queried; when only some can, the checkerboard shows what sparse joint cells recover together with the axes.

5 INFER: REANALYSIS VERSUS MORE CALLS

Design. Here every target condition is measured, so the target is identified in the limit and all of the error comes from the finite sample and the estimator ψ. We filtered classic MMLU (Hendrycks et al., 2021) to 6,505 items (Appendix B) and crossed six prompt formats (a standard format and five PromptEval templates) with the eight label-independent rotations and reflections of the four options, giving 48 conditions. Four models (Qwen3-4B, Phi-4-mini, Qwen3-14B, Granite-3.3-8B) answer 1,024 acquisition items under all 48 conditions with one greedy option-letter token; targets weight the 57 subjects equally and are scored against a disjoint, subject-stratified 4,096-item reference fixed before generation. The lower-tail target is the mean of the $\lceil 0.1 \times 48 \rceil = 5$ lowest condition accuracies. Acquisition follows PromptEval-style two-way balanced sampling at 128–4,096 calls per model, so 128 calls give about 2.7 items per condition. We compare a *plug-in* analysis (per-condition means), empirical-Bayes shrinkage of those means (EB; Efron & Morris, 1975), and a logit model with item and condition main effects as a strong structured comparator. Curves average 50 allocations; paired intervals resample acquisition and reference items within subject and redraw allocations, so a paired contrast need not equal the difference of the two averaged arm errors.

Reanalysis beat four times the calls when the analysis was a plug-in. At 128 calls per model the plug-in lower tail had 39.0 pp error: with a handful of items per condition, the lowest noisy means lie far below the truly lowest conditions. Reanalysing the same responses with EB shrinkage reached 2.8 pp. Four times the plug-in calls reached 17.3 pp (paired difference -14.2 $[-20.2, -10.2]$ pp), and the plug-in had not reached 2.8 pp by 4,096 calls (Fig. 2b). EB was chosen on a dense pilot split of the acquisition bank that an evaluator holding only b responses would not have. A selector that uses only the acquired responses closes this gap: frozen before its intervals were computed, it chose among eight estimators by cross-validation on the b acquired responses and was rerun inside every bootstrap draw (Appendix B.2). At 128 calls it reached 6.2 pp and beat the plug-in at 512 calls by -11.1 $[-18.5, -0.8]$ pp; at 256 calls this contrast did not resolve. Same-data inference thus helps without pilot information, though less than the pilot-selected estimator.

Against a structured estimator the advantage was unresolved. The logit model already had 12.5 pp error at 128 calls. EB at b minus the logit model at $4b$ was -2.5 $[-7.0, 1.3]$ pp for b=128 and 0.0 $[-2.6, 2.5]$ pp for b=256; for the acquired-only selector at 128 it was $+0.6$ $[-6.2, 10.9]$ pp. The large gain is a gain over a plug-in tail estimate, not evidence that reanalysis replaces data. Reanalysis can also choose badly: for the lower tail at 1,024 calls the pilot-based selection had a higher point error than keeping the logit model ($+0.9$ $[-2.7, 3.4]$ pp), and for the mean target at 128–512 calls no action had an interval excluding zero (half-widths 1.5–4 pp). Those comparisons did not resolve a preferred action.

On public banks the target decided which lever paid. The released PromptEval banks for MMLU, BIG-Bench Hard (BBH) and LMentry (Maia Polo et al., 2024b; Suzgun et al., 2023; Efrat et al., 2023) support the same contrast with the released code, sampler, nested budgets and estimators; the released 5th-percentile arrays were first reproduced exactly. For the 5th percentile, PE at b beat the plug-in at $2b$ on MMLU (-6.8 pp at $b = 200$, -6.2 pp at 400) and on BBH (-8.0 and -3.5 pp; Fig. 3a). The advantage is finite: PE at 200 lost to the plug-in at 800 on MMLU ($+4.1$ [1.9, 6.2] pp), and on BBH the contrast was unresolved at $b = 800$. On LMentry no contrast resolved,

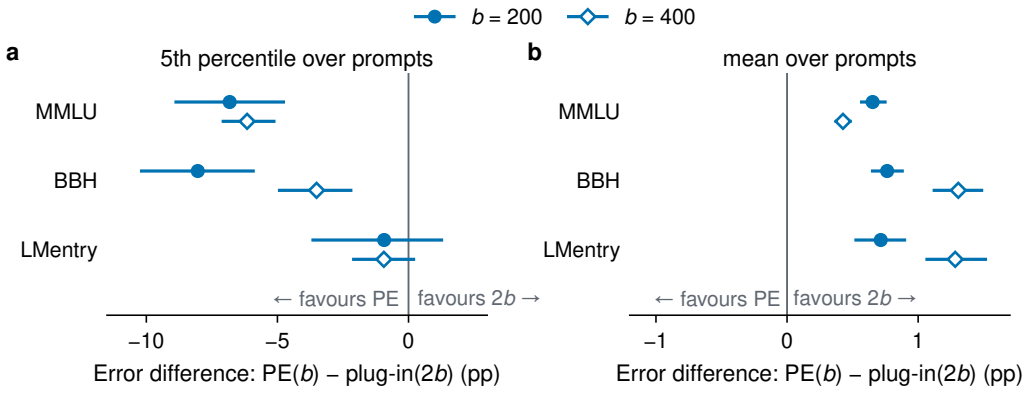

Figure 3: **Public banks: the same responses favour reanalysis for one target and more calls for another.** Released PromptEval banks (Maia Polo et al., 2024b): error of the released PromptEval (PE) Rasch-type estimator at b evaluations per (task, model) minus that of the released plug-in at $2b$ (pp; negative favours reanalysis). (a) 5th percentile over prompts, reproducing the released arrays; (b) mean over prompts, rerun with the released sampler and estimators. The reference is the full in-bank prompt distribution. Bars: 95% task-cluster bootstrap intervals (999 draws), conditional on five averaged seeds; not adjusted across banks or budgets. All budgets in Table 7.

and the covariate variant PE-X became *worse* as the budget grew (7.0 to 9.5 pp). For the mean, the plug-in was at least as accurate as either PE estimator on every bank and budget (Fig. 3b). On one set of responses, the 5th percentile favoured reanalysis and the mean favoured more calls. The check needs no additional model queries, so it can be run before buying calls.

6 Decide: one measurement set, different targets

Design. A learned joint-cell predictor fit on the 47 blur×JPEG checkpoints was transported to 15 new checkpoints from five architecture families (ConvNeXt, ViT/DeiT, Swin, EfficientNet, RegNet; three each), where it failed a prospective transport test. We then compared rules that use one measurement set, the 11 axis cells and the eight checkerboard cells $\mathcal{S}$, measured on every new checkpoint. Two targets are read from the same measurements. The *scalar* target is the 16-cell mean of Section 4; the *held-cell* target is the vector of accuracies on the eight unmeasured cells $\mathcal{T} \setminus \mathcal{S}$, scored by per-cell error. Some rules also use residuals from labelled calibration systems, other new checkpoints whose held cells are known. The comparison is retrospective, added after the transport failure, and five family clusters leave only descriptive intervals.

The simplest rule was among the best for the scalar and the worst for the cells. For the scalar, replacing the frozen predictor by the mean of the eight measured source cells needed no calibration systems and cut error from 0.171 to 0.056 pp (Δ −0.11 [−0.16, −0.07]; Fig. 4a). The same rule assigns that mean to every held cell and had 4.55 pp per-cell error, seven times the frozen predictor's 0.63 pp. The checkerboard places two measured and two held cells in each severity row and column, so their average nearly balances the unmeasured half of the grid while the individual held cells differ by severity. One rule estimated the scalar functional accurately without recovering the surface (Section 3). For the held cells the useful action was calibration: shifting the predictor by the residuals of two labelled same-family systems reduced the error to 0.495 pp (Δ −0.13 [−0.22, −0.05]); one such system, or a shift estimated from twelve other-family systems, did not resolve a gain (Table 5). Refitting on twelve other-family systems reached 0.420 pp, a costlier action. Simplification served the scalar and calibration served the cells.

A choice is a further target. The best model's identity comes from the same responses but can resolve at different costs than the scores. On a near-tie MMLU pair a forced winner at 256 calls was wrong in 17% of replicates; interval rules stopped at a mean of 807 calls and returned both models

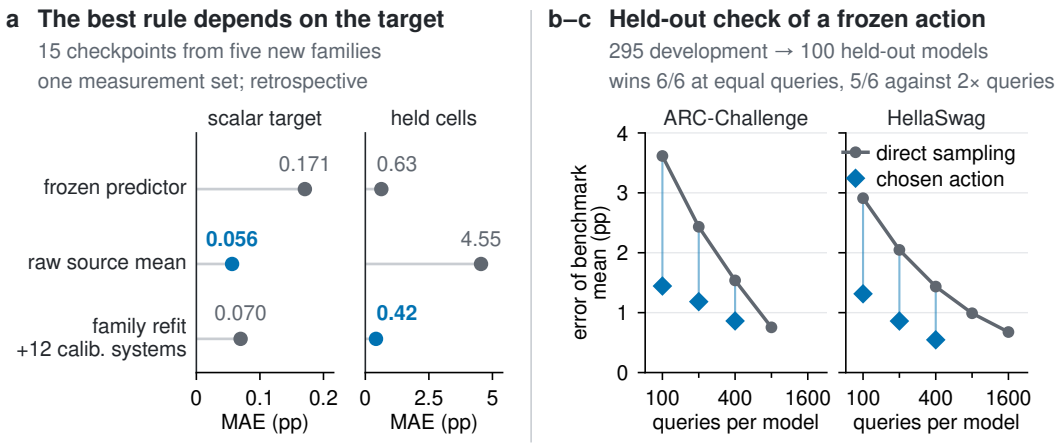

Figure 4: **The useful action depends on the target and on the evidence available.** (a) Retrospective: one measurement set on 15 new-family checkpoints, two targets (mean absolute error, MAE, against the full bank; separate scales). The raw source mean is best for the scalar 16-cell mean and worst for the held cells; the family refit, best for the held cells, uses 12 labelled calibration systems (Table 5). (b, c) Held-out validation: a rule frozen on 295 development models chose among tinyBenchmarks' released estimators, refit on those models only, and was scored once on 100 held-out models. Gray: mean of n uniformly drawn items; blue: chosen estimator on n anchor items. Against $4n$ items it was unresolved in 4 cells and worse in 1. The estimators rely on 8.45M dense historical calibration responses; the saving assumes that bank is reused (Table 9). Errors are evaluation errors, not model performance.

in 59%. The worst of 48 prompt conditions was not resolved although the bottom-10% mean was (Appendix C.3). For such targets, more evidence competes with set-valued reporting and stopping.

7 Reassess in practice

The tool. Reassess runs the comparisons of Sections 4–6 from a response-level bank (models × conditions × items) and a short YAML declaration of target, estimators, plans and candidate actions. It calls exact or released backends unchanged and reports any other method as unsupported. Its action card lists paired errors, intervals and queries and, separately, information beyond queries such as calibration systems or historical responses. Without a reference bank (practitioner mode), reference-dependent claims are `NOT_ASSESSABLE`; the card then diagnoses sensitivity to the analysis without saying which estimate is correct.

Reproduction. Seven worked examples in the package, including the acquired-only MMLU selection of Section 5, regenerate this paper's comparisons from response-level inputs after a clean install. They reproduce the software on the paper's own banks and do not confirm the findings independently.

Held-out use. For external evidence we froze two tasks, budgets and a simple rule before reading any outcome: on 295 development models of the tinyBenchmarks bank (Maia Polo et al., 2024a), adopt the released estimator with the lowest development error if its advantage over direct item sampling is resolved. Scored once on 100 held-out models, the released tutorials' recent-model test split, the chosen IRT estimator reduced the error of the full-benchmark mean on ARC-Challenge and HellaSwag at equal queries in all six task–budget cells (ARC, 100 items: 3.6 to 1.4 pp), beat twice the items in five and never beat four times the items (Fig. 4b–c, Table 9). The estimators are tinyBenchmarks' own; Reassess contributes the frozen comparison and the cost accounting, including 8.45M dense calibration responses from the development models. The savings are marginal savings on new models when a historical bank is already available; building that bank for a small batch of new evaluations would cost more than the queries saved (Appendix D).

Table 1: **Which comparison to run.** Each bottleneck calls for a different matched-cost comparison.

lever	bottleneck	comparison to run	example
Measure	the target depends on conditions not yet observed	new support, including target conditions, vs. replication of the current support	blur×JPEG (Sec. 4)
Infer	target-relevant conditions are observed, but the estimate is unstable	estimators on the same responses vs. more calls	MMLU, PromptEval (Sec. 5)
Decide	a summary adequate for one target is used for another	simplification vs. calibration; more evidence vs. a reported set	scalar vs. cells, near ties (Sec. 6)

8 Discussion

What the design view changes. A single error bar hides which of the three objects of Section 3 limits an evaluation. The same query budget took a target error from 1.10 to 1.05 pp or to 0.38 pp depending on where it was spent; the same 128 acquired responses per model gave 39.0 pp under a plug-in analysis and 6.2 pp with an estimator selected from those responses alone (2.8 pp with pilot-fixed EB); and the same measurements made one rule among the best or the worst depending on the target. Table 1 turns these contrasts into comparisons to run before spending more budget. A smaller interval is not a better evaluation of the claim when the measurements do not determine it, the analysis limits the estimate, or the summary answers another question.

Scope of the evidence. The blur×JPEG comparison, the MMLU contrast against the plug-in and the held-out tinyBenchmarks check carry the main claims. Most other comparisons are retrospective or post hoc, the public-bank results are secondary, and the prespecified blur×gamma test missed its criterion. Reanalysis was not tested prospectively on our own banks. The systems are fixed convenience panels and the external split is not disjoint by base model, so the results need not transfer to other model populations. No experiment computes an identified-set width; the theory fixes the logic of the comparisons, not their magnitudes.

9 Conclusion

Whether a measurement set determines a target is a property of the design; how well the target is estimated is a property of the sample and the estimator; which target matters is a property of the decision. *Measure* when the target depends on conditions not yet observed, *infer* when the target-relevant conditions are already measured but the target is estimated poorly, and *decide* when the decision needs a different quantity from the summary. More evaluation is not one intervention. Before running more evaluation, ask what you need the extra evidence to change.

Reproducibility statement

Appendix A contains the proofs and Appendix B the data, targets, estimators and protocols; the supplement provides the response-bank manifests, code, configurations, tests and scripts that regenerate all result-derived figures and tables.

Ethics statement

The study uses previously released benchmarks (ImageNet-V2, MMLU, the PromptEval and tinyBenchmarks banks) and model checkpoints and does not recruit participants or collect new data from people. The work concerns evaluation methodology and makes no deployment recommendations.

AI use statement

Generative AI tools assisted with research ideation, experiment design, literature search, implementation, analysis, translation, and manuscript and figure preparation. The authors directed the research, reviewed the AI-assisted work and take responsibility for the final submission.

REFERENCES

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron C. Courville, and Marc Bellemare. Deep reinforcement learning at the edge of the statistical precipice. In *Advances in Neural Information Processing Systems*, volume 34, pp. 29304–29320, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/f514cec81cb148559cf475e7426eed5e-Abstract.html.

Norah A. Alzahrani, Hisham Alyahya, Yazeed Alnumay, Sultan AlRashed, Shaykhah Alsubaie, Yousef Almushayqih, Faisal Mirza, Nouf M. Alotaibi, Nora Al-Twairesh, Areeb Alowisheq, M Saiful Bari, and Haidar Khan. When benchmarks are targets: Revealing the sensitivity of large language model leaderboards. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13787–13805. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.744. URL https://aclanthology.org/2024.acl-long.744/.

Anonymous. What success rate hides: Mapping disturbance tolerance in vision-language-action policies. In *Submitted to The Fifteenth International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=Ks45FcVqtO. under review.

Anonymous. Beyond aggregate scores: Revealing conditional model behavior in chest X-Ray nodule detection. In *Submitted to The Fifteenth International Conference on Learning Representations*, 2026b. URL https://openreview.net/forum?id=OCOVgCj6RD. under review.

Jean-Yves Audibert and Sébastien Bubeck. Best arm identification in multi-armed bandits. In *Proceedings of the 23rd Annual Conference on Learning Theory (COLT 2010)*, Haifa, Israel, 2010. URL https://hal.science/hal-00654404.

Alexander D'Amour, Katherine Heller, Dan Moldovan, Ben Adlam, Babak Alipanahi, Alex Beutel, Christina Chen, Jonathan Deaton, Jacob Eisenstein, Matthew D. Hoffman, Farhad Hormozdiari, Neil Houlsby, Shaobo Hou, Ghassen Jerfel, Alan Karthikesalingam, Mario Lucic, Yian Ma, Cory McLean, Diana Mincu, Akinori Mitani, Andrea Montanari, Zachary Nado, Vivek Natarajan, Christopher Nielson, Thomas F. Osborne, Rajiv Raman, Kim Ramasamy, Rory Sayres, Jessica Schrouff, Martin Seneviratne, Shannon Sequeira, Harini Suresh, Victor Veitch, Max Vladymyrov, Xuezhi Wang, Kellie Webster, Steve Yadlowsky, Taedong Yun, Xiaohua Zhai, and D. Sculley. Underspecification presents challenges for credibility in modern machine learning. *Journal of Machine Learning Research*, 23(226):1–61, 2022. URL https://jmlr.org/papers/v23/20-1335.html.

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. URL https://arxiv.org/abs/2107.07002.

Meera Desai, Sang T. Truong, Hanna Wallach, Alex Chouldechova, A. Feder Cooper, Jean Garcia-Gathright, Daniel E. Ho, Abigail Z. Jacobs, Sanmi Koyejo, Nicholas Pangakis, and Angelina Wang. What AI benchmarks actually measure: Adapting convergent and discriminant validity to interrogate fifty-six AI benchmarks. arXiv preprint arXiv:2609.08812, 2026. URL https://arxiv.org/abs/2609.08812.

David L. Donoho. Statistical estimation and optimal recovery. *The Annals of Statistics*, 22(1): 238–270, 1994. doi: 10.1214/aos/1176325367. URL https://doi.org/10.1214/aos/1176325367.

Avia Efrat, Or Honovich, and Omer Levy. LMentry: A language model benchmark of elementary language tasks. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 10476–10501, 2023. doi: 10.18653/v1/2023.findings-acl.666. URL https://aclanthology.org/2023.findings-acl.666/.

Bradley Efron and Carl Morris. Data analysis using Stein's estimator and its generalizations. *Journal of the American Statistical Association*, 70(350):311–319, 1975. doi: 10.1080/01621459.1975.10479864. URL https://doi.org/10.1080/01621459.1975.10479864.

Eyal Even-Dar, Shie Mannor, and Yishay Mansour. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of Machine Learning Research*, 7(39):1079–1105, 2006. URL https://jmlr.org/papers/v7/evendar06a.html.

Paul Gavrikov and Janis Keuper. Can biases in ImageNet models explain generalization? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22184–22194, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Gavrikov_Can_Biases_in_ImageNet_Models_Explain_Generalization_CVPR_2024_paper.html.

Shanti S. Gupta. On some multiple decision (selection and ranking) rules. *Technometrics*, 7(2):225–245, 1965. doi: 10.1080/00401706.1965.10490251. URL https://doi.org/10.1080/00401706.1965.10490251.

Michael Hardy, Anka Reuel, Lijin Zhang, Jodi M. Casabianca, Sang Truong, Yash Dave, Hansol Lee, Benjamin Domingue, and Sanmi Koyejo. AI cartography: Mapping the latent landscape of AI benchmark ecosystems. arXiv preprint arXiv:2605.25272, 2026. URL https://arxiv.org/abs/2605.25272.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2009.03300.

Steven R. Howard, Aaditya Ramdas, Jon McAuliffe, and Jasjeet Sekhon. Time-uniform, nonparametric, nonasymptotic confidence sequences. *The Annals of Statistics*, 49(2):1055–1080, 2021. doi: 10.1214/20-AOS1991. URL https://doi.org/10.1214/20-AOS1991.

Christopher Jennison and Bruce W. Turnbull. *Group Sequential Methods with Applications to Clinical Trials*. Chapman and Hall/CRC, 1999. doi: 10.1201/9780367805326. URL https://doi.org/10.1201/9780367805326.

Zohar Karnin, Tomer Koren, and Oren Somekh. Almost optimal exploration in multi-armed bandits. In *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pp. 1238–1246. PMLR, 2013. URL https://proceedings.mlr.press/v28/karnin13.html.

Emilie Kaufmann, Olivier Cappé, and Aurélien Garivier. On the complexity of best-arm identification in multi-armed bandit models. *Journal of Machine Learning Research*, 17(1):1–42, 2016. URL https://www.jmlr.org/papers/v17/kaufman16a.html.

Andrew Keller, Kweku Kwegyir-Aggrey, Ryan Steed, Anita Rao, Julia Sharp, and Amanda Bergman. Expanding the AI evaluation toolbox with statistical models. NIST AI 800-3, National Institute of Standards and Technology, February 2026. URL https://www.nist.gov/publications/expanding-ai-evaluation-toolbox-statistical-models.

Rasul Khanbayov, Mariam Sohail, Ahmed Abdala, and Hasan Kurban. One perturbation is not enough: Identifiability and blind baselines for behavioral AI evaluation. arXiv preprint arXiv:2609.06190, 2026. URL https://arxiv.org/abs/2609.06190.

J. Kiefer and J. Wolfowitz. The equivalence of two extremum problems. *Canadian Journal of Mathematics*, 12:363–366, 1960. doi: 10.4153/cjm-1960-030-4. URL https://doi.org/10.4153/cjm-1960-030-4.

Seong-Hee Kim and Barry L. Nelson. A fully sequential procedure for indifference-zone selection in simulation. *ACM Transactions on Modeling and Computer Simulation*, 11(3):251–273, 2001. doi: 10.1145/502109.502111. URL https://doi.org/10.1145/502109.502111.

Jannik Kossen, Sebastian Farquhar, Yarin Gal, and Tom Rainforth. Active testing: Sample-efficient model evaluation. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 5753–5763. PMLR, 2021. URL https://proceedings.mlr.press/v139/kossen21a.html.

F. Y. Kuo, I. H. Sloan, G. W. Wasilkowski, and H. Woźniakowski. On decompositions of multivariate functions. *Mathematics of Computation*, 79(270):953–966, 2010. doi: 10.1090/S0025-5718-09-02319-9. URL https://doi.org/10.1090/S0025-5718-09-02319-9.

Felipe Maia Polo, Lucas Weber, Leshem Choshen, Yuekai Sun, Gongjun Xu, and Mikhail Yurochkin. tinyBenchmarks: evaluating LLMs with fewer examples. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 34303–34326. PMLR, 2024a. URL https://proceedings.mlr.press/v235/maia-polo24a.html. arXiv:2402.14992.

Felipe Maia Polo, Ronald Xu, Lucas Weber, Mírian Silva, Onkar Bhardwaj, Leshem Choshen, Allysson Flavio Melo de Oliveira, Yuekai Sun, and Mikhail Yurochkin. Efficient multi-prompt evaluation of LLMs. In *Advances in Neural Information Processing Systems*, volume 37, 2024b. doi: 10.52202/079017-0707. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/28236482f64a72eec43706b6f3a6c511-Abstract-Conference.html. arXiv:2405.17202.

Charles F. Manski. *Partial Identification of Probability Distributions*. Springer Series in Statistics. Springer-Verlag, 2003. doi: 10.1007/b97478. URL https://doi.org/10.1007/b97478.

Charles A. Micchelli and Theodore J. Rivlin. A survey of optimal recovery. In Charles A. Micchelli and Theodore J. Rivlin (eds.), *Optimal Estimation in Approximation Theory*, pp. 1–54. Plenum Press, New York, 1977. doi: 10.1007/978-1-4684-2388-4_1. URL https://doi.org/10.1007/978-1-4684-2388-4_1.

Grigor Nalbandyan, Rima Shahbazyan, and Evelina Bakhturina. SCORE: Systematic COnsistency and robustness evaluation for large language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 3: Industry Track)*, pp. 470–484. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.naacl-industry.39. URL https://aclanthology.org/2025.naacl-industry.39/.

Barry L. Nelson and Frank J. Matejcik. Using common random numbers for indifference-zone selection and multiple comparisons in simulation. *Management Science*, 41(12):1935–1945, 1995. doi: 10.1287/mnsc.41.12.1935. URL https://doi.org/10.1287/mnsc.41.12.1935.

Friedrich Pukelsheim. *Optimal Design of Experiments*. John Wiley & Sons, New York, 1993. URL https://www.math.uni-augsburg.de/htdocs/emeriti/pukelsheim/publikationen.html#1993.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do ImageNet classifiers generalize to ImageNet? In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 5389–5400. PMLR, 2019. URL https://proceedings.mlr.press/v97/recht19a.html.

Joseph P. Romano and Michael Wolf. Exact and approximate stepdown methods for multiple hypothesis testing. *Journal of the American Statistical Association*, 100(469):94–108, 2005. doi: 10.1198/016214504000000539. URL https://doi.org/10.1198/016214504000000539.

Alexander Rubinstein, Benjamin Raible, Martin Gubri, and Seong Joon Oh. DISCO: Diversifying sample condensation for efficient model evaluation. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://iclr.cc/virtual/2026/poster/10009382. arXiv:2510.07959.

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. URL https://arxiv.org/abs/2505.10573.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. Quantifying language models' sensitivity to spurious features in prompt design or: How I learned to start worrying about prompt formatting. In *International Conference on Learning Representations*, 2024. arXiv:2310.11324.

Chengshuai Shi, Kun Yang, Zihan Chen, Jundong Li, Jing Yang, and Cong Shen. Efficient prompt optimization through the lens of best arm identification. In *Advances in Neural Information Processing Systems*, volume 37, pp. 99646–99685, 2024. doi: 10.52202/079017-3161. URL https://doi.org/10.52202/079017-3161.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V. Le, Ed H. Chi, Denny Zhou, and Jason Wei. Challenging BIG-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 13003–13051, 2023. doi: 10.18653/v1/2023.findings-acl.824. URL https://aclanthology.org/2023.findings-acl.824/.

Sang T. Truong and Sanmi Koyejo. AI measurement science. Online textbook, 2026. URL https://aimslab.stanford.edu/textbook/. Accessed September 2026.

Sang T. Truong, Yuheng Tu, Percy Liang, Bo Li, and Sanmi Koyejo. Reliable and efficient amortized model-based evaluation. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 60238–60265. PMLR, 2025. URL https://proceedings.mlr.press/v267/truong25c.html.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. In *Advances in Neural Information Processing Systems 37 (NeurIPS 2024 Datasets and Benchmarks Track)*, pp. 95266–95290, 2024. doi: 10.52202/079017-3018. URL https://doi.org/10.52202/079017-3018.

Ian Waudby-Smith and Aaditya Ramdas. Estimating means of bounded random variables by betting. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 86(1):1–27, 2024. doi: 10.1093/jrsssb/qkad009. URL https://doi.org/10.1093/jrsssb/qkad009.

Skyler Wu, Yash Nair, and Emmanuel J. Candès. Efficient evaluation of LLM performance with statistical guarantees. arXiv preprint arXiv:2601.20251, 2026. URL https://arxiv.org/abs/2601.20251.

Guanhua Zhang and Moritz Hardt. Inherent trade-offs between diversity and stability in multi-task benchmarks. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 58984–59002. PMLR, 2024. URL https://proceedings.mlr.press/v235/zhang24u.html.

Taolin Zhang, Hang Guo, Wang Lu, Tao Dai, Shu-Tao Xia, and Jindong Wang. SparseEval: Efficient evaluation of large language models by sparse optimization. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://iclr.cc/virtual/2026/poster/10010846. arXiv:2602.07909.

Lexin Zhou, Lorenzo Pacchiardi, Fernando Martínez-Plumed, Katherine M. Collins, Yael Moros-Daval, Seraphina Zhang, Qinlin Zhao, Yitian Huang, Luning Sun, Jonathan E. Prunty, Zongqian Li, Pablo Sánchez-García, Kexin Jiang-Chen, Pablo A. M. Casares, Jiyun Zu, John Burden, Behzad Mehrbakhsh, David Stillwell, Manuel Cebrian, Jindong Wang, Peter Henderson, Sherry Tongshuang Wu, Patrick C. Kyllonen, Lucy Cheke, Xing Xie, and José Hernández-Orallo. General scales unlock AI evaluation with explanatory and predictive power. *Nature*, 652:58–67, 2026. doi: 10.1038/s41586-026-10303-2. URL https://doi.org/10.1038/s41586-026-10303-2. Preprint: arXiv:2503.06378.

A Formal statements and proofs

Throughout, Θ is a nonempty set of mean response surfaces $\theta : \mathcal{Z} \to [0,1]$ on a finite condition set $\mathcal{Z}$, $B : \Theta \to \mathcal{Y}$ maps a surface to its noise-free measurements in a normed space, and $T : \Theta \to \mathbb{R}$ is a scalar target. For an observation y and $\delta \geq 0$, $\mathcal{C}(y,\delta)$ and $D_T(y,\delta)$ are defined by Eq. (1). The global modulus is $\omega_T(\varepsilon) = \sup\{|T(\theta) - T(\theta')| : \theta, \theta' \in \Theta,\ \|B(\theta) - B(\theta')\| \leq \varepsilon\}$. These are deterministic statements about Θ and B; sampling enters only through the tolerance δ that one is willing to assume.

Proposition A.1 (Fiber identification)**.** *T is identified by B on Θ, meaning that $B(\theta) = B(\theta')$ implies $T(\theta) = T(\theta')$, if and only if T is constant on every nonempty fiber of B, equivalently $T = g \circ B$ for some $g : B(\Theta) \to \mathbb{R}$, equivalently $\omega_T(0) = 0$. If $B(\theta) = A\theta$ and $T(\theta) = t^\top\theta$, identification holds iff $t^\top v = 0$ for every $v \in (\Theta - \Theta) \cap \ker A$; the condition $t \in \mathrm{range}(A^\top)$ is sufficient and, when Θ has nonempty interior, necessary.*

Proof. If T is constant on fibers, set $g(\beta)$ to the common value of T on $B^{-1}(\beta)$; the converse is immediate, and both say that every equal-observation gap in $\omega_T(0)$ vanishes. In the linear case the equal-observation differences are exactly $(\Theta - \Theta) \cap \ker A$. If $t = A^\top a$ then $T = a^\top B$. If θ_0 is interior and $v \in \ker A$, then $\theta_0 + sv \in \Theta$ for small $s \neq 0$, so identification forces $t^\top v = 0$; hence $t \in (\ker A)^\perp = \mathrm{range}(A^\top)$. The function g need not be continuous or be the estimator actually fitted. □

In the linear case this is the kernel/coset characterisation that Khanbayov et al. (2026) give for log-linear perturbation-response models. Restricted classes matter: on $\{(s,s) : s \in [0,1]\}$, observing the first coordinate identifies the second although the second is not in the ambient row space.

Proposition A.2 (Midpoint recovery)**.** *Let $\mathcal{C} = \mathcal{C}(y,\delta)$ be nonempty and let $L = \inf_{\mathcal{C}} T$ and $U = \sup_{\mathcal{C}} T$ be finite. Then $\inf_{a\in\mathbb{R}} \sup_{\theta\in\mathcal{C}} |a - T(\theta)| = (U - L)/2$, attained at $a = (L + U)/2$. If observations are $B(\theta) + e$ with $\|e\| \leq \delta$ and T is bounded, the minimax absolute error over all deterministic reports is $\omega_T(2\delta)/2$.*

Proof. For any a, sequences approaching L and U give $\sup_{\mathcal{C}} |a - T| \geq \max\{|a - L|, |a - U|\} \geq (U - L)/2$, and the midpoint attains the bound whether or not L and U are attained. Two members of $\mathcal{C}(y,\delta)$ have observations within 2δ, so reporting the midpoint of each nonempty compatible set has worst-case error at most $\omega_T(2\delta)/2$. Conversely, if $\|B(\theta) - B(\theta')\| \leq 2\delta$, then $y = \{B(\theta) + B(\theta')\}/2$ is a possible observation under both surfaces, and any report at y errs by at least $|T(\theta) - T(\theta')|/2$ under one of them. □

The statement imposes no measurability or regularity on the report. It differs from Fisher–Neyman sufficiency: a success count is sufficient for a Bernoulli rate, yet one realised count does not determine the rate. It also differs from a confidence interval, which concerns sampling variability of y for a fixed θ rather than the spread of T over surfaces with the same $B(\theta)$.

Lemma A.3 (Precision limit)**.** *If Θ is compact and B, T are continuous, then $\omega_T(\varepsilon) \downarrow \omega_T(0)$ as $\varepsilon \downarrow 0$, and $D_T(B(\theta),\delta) \downarrow D_T(B(\theta),0)$ as $\delta \downarrow 0$.*

Proof. The feasible pairs $\{(\theta,\theta') : \|B(\theta) - B(\theta')\| \leq \varepsilon\}$ are nonempty, compact and nested, and the continuous gap attains its maximum on each. Maximising pairs along $\varepsilon_n \downarrow 0$ have a convergent subsequence whose limit has equal observations, and continuity preserves the gap; nesting gives the reverse inequality. The local statement is the same argument for the compact sets $\mathcal{C}(B(\theta),\delta)$. □

Repeating the same measurements can lower the attainable δ but cannot push the width below the exact fiber width. Compactness is needed: on $(0,1]$ with $B(r) = r$ and $T(r) = \sin(1/r)$, $\omega_T(0) = 0$ but $\omega_T(\varepsilon) = 2$ for all $\varepsilon > 0$.

Proposition A.4 (Nested augmentation)**.** *If new measurement constraints are added without changing the old ones, so that $\mathcal{C}' \subseteq \mathcal{C}$ with both nonempty and T bounded, then $D_T(\mathcal{C}') \leq D_T(\mathcal{C})$.*

Proof. The target image of $\mathcal{C}'$ is contained in that of $\mathcal{C}$. □

This is monotonicity only. The reduction from a set of new measurements is not in general submodular, so no greedy guarantee follows.

Two factors. Let $\mathcal{Z} = \{0, \dots, 5\}^2$, let $\mathcal{A} = \{(u, v) : u = 0 \text{ or } v = 0\}$ be the axis cells, and let $\mathcal{T} \subseteq \mathcal{Z} \setminus \mathcal{A}$ be the target cells, with weights $q \geq 0$ on $\mathcal{T}$ and $\sum_{\mathcal{T}} q_z = 1$. The anchored residual $h(u, v) = \theta(u, v) - \theta(u, 0) - \theta(0, v) + \theta(0, 0)$ vanishes on $\mathcal{A}$, so Eq. (3) is an identity. It differs from integral ANOVA, whose components satisfy zero-integral constraints under a product measure and would use interior cells (Kuo et al., 2010). Let $B_{\mathcal{A}}(\theta) = (\theta(z))_{z \in \mathcal{A}}$ and $T_q(\theta) = \sum_{\mathcal{T}} q_z \theta(z)$.

Proposition A.5 (What the axes and joint cells determine)**.** (a) *On* $\Theta_0 = \{\theta \in [0, 1]^{\mathcal{Z}} : h \equiv 0\}$, $B_{\mathcal{A}}$ *identifies* θ *and hence* T_q. (b) *On* $\Theta_\eta = \{\theta \in [0, 1]^{\mathcal{Z}} : |h(z)| \leq \eta \ \forall z \in \mathcal{T}\}$, *the exact width from the axes is at most* 2η. *It equals* 2η *at any axis profile for which* $\theta^{\mathrm{add}}(z) \pm \eta \in [0, 1]$ *on* $\mathcal{T}$. (c) *On the unrestricted cube* $[0, 1]^{\mathcal{Z}}$, *the exact width from the axes is* $\sum_{z \notin \mathcal{A}} q_z = 1$. (d) *On* $\Theta_{\mathrm{c}} = \{\theta \in [0, 1]^{\mathcal{Z}} : h \text{ is constant on } \mathcal{T}\}$, *the axes together with any one cell* $z_0 \in \mathcal{T}$ *identify* T_q, *and the axis-adjusted mean of Appendix B.2 equals* T_q *for any nonempty measured subset of* $\mathcal{T}$. (e) *On any* Θ, *measuring every cell of* $\mathcal{T}$ *identifies* T_q.

Proof. By Eq. (4), $T_q(\theta) = \sum_{\mathcal{T}} q_z \theta^{\mathrm{add}}(z) + \sum_{\mathcal{T}} q_z h(z)$, and the first term is a function of $B_{\mathcal{A}}$. (a) $h \equiv 0$ gives $\theta = \theta^{\mathrm{add}}$. (b) Two surfaces with the same axes differ in T_q by $\sum q_z \{h(z) - h'(z)\}$, at most 2η in absolute value; setting $h = +\eta$ and $h = -\eta$ on $\mathcal{T}$ (with any common values on the other off-axis cells) is feasible under the stated slackness and attains 2η. (c) Fix the axes and set all off-axis cells to 0 in one surface and to 1 in the other; each off-axis cell differs by at most 1, which gives the bound. (d) With $h \equiv c$ on $\mathcal{T}$, one measured cell reveals $c = \theta(z_0) - \theta^{\mathrm{add}}(z_0)$, so $T_q = \sum q_z \theta^{\mathrm{add}}(z) + c$. The mean residual of any nonempty measured set $M \subseteq \mathcal{T}$ is c, and $\theta^{\mathrm{add}}(z) + c = \theta(z) \in [0, 1]$, so the clipping in the estimator is inactive. (e) T_q is then a function of the measurements. □

Parts (a)–(c) hold verbatim with θ^{add} replaced by any other function of the axis measurements, such as the product completion $\theta(u, 0)\theta(0, v)/\theta(0, 0)$, and h by the corresponding residual; the product rule used in Section 4 is exact under that multiplicative analogue of (a). The three vision actions of Section 4 correspond to (a) or its multiplicative analogue, (d) and (e). More axis items sharpen θ^{add} but leave the residual term unmeasured. Joint cells measure h where they are placed, which suffices under (d). Measuring the target cells needs no structural class. In finite samples each estimate also carries sampling error. The empirical question is which of these errors dominates at a matched budget, and no data set tells us which class contains real models.

B Data, targets and estimators

B.1 Banks, protocols and query accounting

Table 2 summarises the five response banks; the paragraphs below give the protocols.

Vision. Images are resized (shorter side 256, bilinear) and centre-cropped to 224 pixels, the shared classifier protocol; corruptions are composed in the order blur, then the second corruption. Gaussian blur is applied in linear light with $\sigma \in \{0, 0.5, 1, 1.5, 2, 3\}$ pixels (reflect boundary, truncation 4σ); JPEG quality levels are none, 95, 85, 65, 45 and 20 (no chroma subsampling); gamma levels are 1.0, 1.15, 1.35, 1.65, 2.0 and 2.5. For blur×gamma only the 11 axis cells and the 16 target cells were generated. More axis items are added class-balanced: at $n = 4$ items per class, action A adds 1,455 items to each of the 11 axis cells (16,005 forwards per model), and action B measures four joint cells on the 4,000 starting items (16,000 forwards).

Vision condition support. Index the six levels of each corruption by $0, \dots, 5$ (0 = clean) and write a cell as (u, v), with u the blur level and v the JPEG or gamma level. The analysis code defines the same index sets for both pairs and for the new-family checkpoints:

$$\mathcal{A} = \{(u, v) : u = 0 \text{ or } v = 0\} \text{ (11 axis cells)}, \qquad \mathcal{T} = \{1, 2, 3, 4\}^2 \text{ (16 target cells)},$$
$$\mathcal{S} = \{(u, v) \in \mathcal{T} : u + v \text{ even}\} = \{(1, 1), (1, 3), (2, 2), (2, 4), (3, 1), (3, 3), (4, 2), (4, 4)\},$$
$$\mathcal{T} \setminus \mathcal{S} = \{(1, 2), (1, 4), (2, 1), (2, 3), (3, 2), (3, 4), (4, 1), (4, 3)\}.$$

Table 2: Response banks. "Reference" is the dense evaluation against which errors are scored.

study	systems	conditions	items and reference	cost unit
vision, blur×JPEG	47 ResNet-50 checkpoints, 10 training-recipe blocks	6×6 grid; target: levels 1–4 on both axes (16 cells)	8,000 ImageNet-V2 images (8/class); full bank, contains queried items	forwards / model
vision, blur×gamma	same checkpoints	6×6 grid; 27 cells generated	same items; full bank	forwards / model
vision, new families	15 checkpoints: ConvNeXt, ViT/DeiT, Swin, EfficientNet, RegNet (3 each)	blur×JPEG grid	same item domain; full bank	labelled systems
MMLU	4 acquisition models; 6-model and 2-model decision panels	6 formats $\times$ 8 option arrangements	1,024 queryable items; disjoint 4,096-item reference	calls / model
PromptEval	released MMLU, BBH, LMentry banks	released prompt templates	full in-bank prompt distribution	evals / (task, model)

The target for model m is $\frac{1}{16}\sum_{(u,v)\in\mathcal{T}} a_m(u,v)$, with a_m the class-balanced accuracy on all 8,000 items. The extend-conditions action measures a uniformly drawn k-subset of the checkerboard $\mathcal{S}$ per replicate; the held cells $\mathcal{T} \setminus \mathcal{S}$ are not measured in the original comparison or in the new-family study, and the new-family per-cell error is computed on them; only the post-hoc target-cell arm (Appendix D) measures them. The level-5 axis cells $(5,0)$ and $(0,5)$ are measured and charged in the starting and more-of-the-same costs but do not enter any 16-cell estimator; joint cells with $u = 5$ or $v = 5$ are outside the target.

MMLU. Exact and near-duplicate overlaps with MMLU-Pro (Wang et al., 2024), within-MMLU duplicates and items whose options refer to other options were excluded before any outcome, leaving 6,505 items. The 1,024 acquisition items and the 4,096 reference items are disjoint subject-stratified draws; the reference membership was fixed before generation. Prompts use each publisher's chat template in a single user turn (Qwen3 in non-thinking mode), temperature 0, and one answer token restricted to the option letters (100% parse coverage). Sparse acquisition plans are replayed as masks over the dense acquisition bank, so all estimators and arms see identical responses. A budget counts unique (condition, item) calls per model.

PromptEval. We used the released code, sampler and data (Maia Polo et al., 2024b) and reproduced the released plug-in (`Baseline`), PE and PE-X (discrete covariates) 5th-percentile estimates exactly before any reanalysis. Budgets follow the released nested schedule (200, 400, 800 and 1,600 evaluations per task–model; five seeds). Mean targets rerun the released sampler and estimators because the released arrays contain only quantiles. Error is averaged over seeds and then over task–model pairs.

Query accounting. Costs count model queries only. Estimator fitting, bootstrap computation, the dense references and the calibration systems used in earlier studies are not charged to any arm. Adaptive decision rules are charged their total calls divided by the initial panel size, which is not equal calls per model.

B.2 Estimators and uncertainty

Language estimators. All MMLU estimators produce fixed-subject-weight condition accuracies $\hat{\theta}_m(c)$, from which targets are computed. The bottom-10% target sorts the 48 values of $\theta_m(\cdot)$ and averages the $k = \lceil 0.1 \times 48 \rceil = 5$ smallest with equal weights; ties do not matter because the mean of the five smallest values does not depend on their order. The same function is applied to reference and estimated accuracies. This local functional differs from the PromptEval 5th percentile, which is the released quantile over prompt templates. *Pooled*: weighted mean of all queried outcomes (mean target only). *Plug-in cell means*: fixed-weight per-condition means; empty (condition, subject) cells are filled with the subject's pooled mean. *EB shrinkage*: with μ the mean of the observed cell means and $v_c = \mu(1-\mu)/n_c$, we set $\tau^2 = \max(0, \widehat{\mathrm{var}}_c\,\hat{\theta}_c - \bar{v})$ and shrink $\hat{\theta}_c \mapsto \mu + \tau^2/(\tau^2 + v_c)\,(\hat{\theta}_c - \mu)$.

This is a normal-approximation moment estimator in the spirit of Efron & Morris (1975), not an exact beta-binomial model. *Linear main effects*: a linear probability model with item intercepts plus prompt-format and arrangement main effects (ridge on the condition part). *Logit main effects*: the same structure on the logit scale with L2 penalties. *PE / PE-X*: PromptEval's extended Rasch model with one-hot conditions, or with prompt and arrangement covariates; PE-X is structurally unreliable for the minimum target (wrong winner in 73–100% of draws). The pilot-based selection (reference-free, used for EB in Section 5) simulates allocations on subject halves of the dense acquisition bank (1,024 items under all 48 conditions per model) and never uses the reference; it selected EB for the bottom-10% target up to 512 calls and the linear model at 1,024. It therefore uses pilot responses beyond the b acquired calls, which the call counts do not charge. *Acquired-only selection* (post-hoc fairness closure, frozen before its intervals were computed; no new model inference) chooses among the eight estimators above by five-fold held-out-item Brier loss within subject, using only the b acquired calls, and is rerun in each of the 299 bootstrap draws so that selection uncertainty enters the interval. At 128 calls its error was 6.2 pp; minus the plug-in at 512 calls it was -11.1 $[-18.5, -0.8]$ pp and minus the logit model at 512 it was $+0.6$ $[-6.2, 10.9]$ pp. At 256 calls the two contrasts were -5.8 $[-11.9, 3.4]$ and $+2.3$ $[-2.1, 10.7]$ pp. It chose linear or logit models, never EB, so it is a different estimator from the one behind the 2.8 pp contrast.

Vision estimators. With $a(u, v)$ the class-balanced accuracy of one model on the measured items (the starting items plus any added items), the axes rules predict a target cell $(u, v) \in \mathcal{T}$ by

$$\hat{a}^{\mathrm{add}}(u, v) = a(u, 0) + a(0, v) - a(0, 0), \qquad \hat{a}^{\mathrm{prod}}(u, v) = \frac{a(u, 0)\, a(0, v)}{\max\{a(0, 0), 10^{-6}\}},$$

each clipped to $[0, 1]$; the product rule normalises by clean accuracy. The *axis-adjusted mean* with measured joint cells $M \subseteq \mathcal{S}$ ($|M| = k$, on the starting items) uses the mean additive residual $\bar{r} = \frac{1}{k} \sum_{(u,v)\in M} \{a(u, v) - \hat{a}^{\mathrm{add}}(u, v)\}$, sets $\hat{a}(u, v) = a(u, v)$ on M and $\hat{a}(u, v) = \mathrm{clip}_{[0,1]}\{\hat{a}^{\mathrm{add}}(u, v) + \bar{r}\}$ on $\mathcal{T} \setminus M$, and estimates the target by $\frac{1}{16} \sum_{\mathcal{T}} \hat{a}(u, v)$. The raw joint mean instead fills $\mathcal{T} \setminus M$ with the mean of the measured joint cells. *Ridge heads* are cross-model ridge regressions from measured cells to the 16 target cells, fit on a separate discovery item set, leaving each training-recipe block out.

Resampling. Vision: each replicate redraws class-balanced items and joint cells; intervals for the allocation-averaged contrast resample items within class (999 draws, 50 allocations each, for blur×gamma), with a class-cluster sensitivity. The 2.5–97.5% range of the contrast across individual allocations is reported separately and is *not* a confidence interval. MMLU: 299 draws resample acquisition and reference items within subject and redraw allocations, holding the original-data estimator choice fixed, so selection uncertainty is not included, except in the acquired-only selection, which is rerun in every draw. PromptEval: 999 task-cluster draws (all models of a task kept together), paired across actions and conditional on the five averaged seeds. New-family vision: five family clusters, so intervals are descriptive. All intervals are pointwise and treat items within a stratum as exchangeable.

Decision rules. The fixed-look interval rule computes simultaneous max-t intervals for pairwise differences by an item-cluster bootstrap within subject (200 inner draws), with level $\alpha/4$ at each of four looks, and stops when the winner is separated from all others; otherwise it returns the winner and every model not separated from it. This set is not a simultaneous ranking set. Paired elimination uses the same intervals and drops models separated from a better active model. We do not establish nominal coverage after winner or active-set selection (Romano & Wolf, 2005). Sequential Halving and Successive Rejects use their standard fixed-total schedules. The Hoeffding elimination uses union-bound Hoeffding intervals. For EB tail targets the item bootstrap distorts the prior-variance estimate, which biases the shrunken values and widens the intervals.

C Additional empirical results

C.1 Vision

Further blur×JPEG diagnostics. Levels in Fig. 2a use the 200 canonical allocations; post-hoc contrasts use 1,000. The joint cells needed the axis structure: their raw mean had 2.15 pp error

Table 3: Second corruption pair (blur×gamma): gain of four joint conditions over more axis items by training-recipe block (pp; positive favours joint conditions; mean over checkpoints of the per-checkpoint mean paired gain), with blur×JPEG under the same code. Overall $\Delta = -0.108$ pp; prespecified class-stratified 95% interval [-0.205, -0.072]; class-cluster interval [-0.202, -0.057]; allocation 2.5–97.5% range [-0.413, +0.361] (not a confidence interval). 5 of 10 blocks are positive; the prespecified criterion required 6, so the test did not meet its criterion.

training-recipe block	blur×gamma	blur×JPEG
self supervised linear probe	+0.399	+2.610
standard supervised recipes	+0.258	+0.305
pixmix	+0.205	+0.466
noisymix	+0.063	+0.014
adversarial training grid	+0.016	-0.010
shape texture interventions	-0.061	+0.149
frozen random filters	-0.074	+0.204
augmix deepaugment	-0.083	+0.587
optics augmentation	-0.092	+0.459
prime	-0.107	-0.058

Table 4: **Post-hoc vision arms at matched added forwards** (cached banks; no new model responses). Levels: error of the 16-cell target (pp); blur×JPEG levels use the first 200 allocations, which reproduce the original arms draw for draw, and blur×gamma levels use all 1,000. Contrasts at 16k: mean over 1,000 allocations, class-stratified bootstrap 95% interval (999 draws × 50 allocations), and recipe blocks (of 10) favouring the first arm. All start from the 11 axis cells on 4 items per class; the checkerboard and the first two target-cell arms reuse those items, and the last draws items from the full bank, which is also the reference.

	blur×JPEG			blur×gamma		
arm (estimator)	8k	16k	32k	8k	16k	32k
more axis items, 11 cells (original)	1.07	1.05	1.01	0.55	0.51	0.48
more axis items, 9 relevant cells	1.07	1.03	1.00	0.54	0.50	0.47
checkerboard, axis-adjusted (original)	0.78	0.56	0.37	0.48	0.41	0.35
16 target cells, direct mean	0.42	0.38	0.37	0.44	0.37	0.35
16 target cells, axis-difference	0.42	0.38	0.37	0.41	0.37	0.36
16 target cells, full-bank items	0.27	0.11	0.06	0.28	0.12	0.07
contrast at 16k	Δ [95%]; blocks			Δ [95%]; blocks		
9-cell − 11-cell axes	−0.004 [−0.022, 0.005]; 9/10			−0.008 [−0.029, 0.009]; 10/10		
checkerboard − 9-cell axes	−0.501 [−0.577, −0.439]; 8/10			−0.099 [−0.197, −0.067]; 5/10		
16 cells − checkerboard	−0.169 [−0.236, −0.138]; 10/10			−0.038 [−0.072, −0.031]; 10/10		
16 cells − 11-cell axes	−0.675 [−0.770, −0.636]; 9/10			−0.146 [−0.259, −0.121]; 5/10		

at $k = 4$. Among analyses of the axes alone, a prespecified cross-model ridge head was 0.4–0.5 pp *worse*. The 0.99 pp residual belongs to the tested axes-only rules and is not a bound for every estimator; a post-hoc leave-one-checkpoint-out head reached 0.79 pp. At 32,000 forwards the checkerboard and target-cell actions both reached 0.37 pp. That floor comes from measuring only the 4,000 starting items against a reference on all 8,000: target cells measured on items drawn from the whole bank reached 0.11 pp at 16,000, partly because those items are also in the reference.

Transported predictor. The frozen predictor is a cross-model ridge head fit on the 47 blur×JPEG checkpoints that maps the 11 axis cells and the eight source joint cells to the 16 target cells; on the 15 new-family checkpoints it failed its prospective transport criterion. Residual shifts add the mean held-cell residual of the labelled calibration systems to the frozen prediction; the leave-family-out refit re-estimates the head on the twelve checkpoints of the other four families.

Table 5: New-family vision checkpoints: remedies for a poorly transported predictor (15 checkpoints, five family clusters). Scalar: 16-cell target mean with the eight source cells measured; cells: per-cell error on the eight held cells $\mathcal{T} \setminus \mathcal{S}$. Δ: difference of 15-checkpoint mean errors from the frozen predictor, with paired family-cluster bootstrap intervals (descriptive). Full version of Fig. 4a. Systems: added labelled calibration systems; 0 means no *extra* calibration systems, not zero measurement cost (every rule uses the measured axis and source cells).

action	systems	scalar / cells (pp)	Δ scalar [95%]	Δ cells [95%]
frozen predictor	0	0.171 / 0.630	–	–
frozen axes-only rule	0	0.282 / 0.866	+0.11 [−0.06, 0.29]	+0.24 [−0.02, 0.50]
raw source-joint mean	0	0.056 / 4.551	−0.11 [−0.16, −0.07]	+3.9 [3.4, 4.6]
axis-adjusted source mean	0	0.056 / 1.130	−0.11 [−0.16, −0.07]	+0.50 [0.17, 0.83]
same-family shift, 1 system	1	0.146 / 0.574	−0.02 [−0.13, 0.07]	−0.06 [−0.15, 0.04]
same-family shift, 2 systems	2	0.122 / 0.495	−0.05 [−0.14, 0.03]	−0.13 [−0.22, −0.05]
other-family shift	12	0.120 / 0.526	−0.05 [−0.10, 0.00]	−0.10 [−0.24, 0.07]
leave-family-out refit	12	0.070 / 0.420	−0.10 [−0.14, −0.07]	−0.21 [−0.39, −0.00]

C.2 Language: estimators and public banks

Table 6: MMLU bottom-10%: reanalysis versus more calls (pp). E_0 is the starting estimator; "4×" is E_0 at four times the calls. Matching calls: E_0 calls needed to match the reanalysis error (log-interpolated up to 4,096). Reanalysis is the estimator chosen on the dense pilot split (EB up to 512 calls); acquired-only selection is reported in Appendix B.2.

calls	E_0	E_0	reanalysis	E_0 at 4×	reanalysis − 4× [95%]	matching calls
128	plug-in	38.98	2.80	17.30	−14.2 [−20.2, −10.2]	>4,096
128	logit	12.49	2.80	5.12	−2.5 [−7.0, 1.3]	≈849
256	plug-in	26.62	2.11	10.31	−8.1 [−11.9, −3.3]	>4,096
256	logit	9.20	2.11	1.94	0.0 [−2.6, 2.5]	≈985
1,024	logit	1.94	2.95	–	vs E_0: +0.9 [−2.7, 3.4]	–

Further MMLU diagnostics. Table 6 gives the reanalysis-versus-more-calls contrasts behind Fig. 2b. Redirecting calls to 13–16 conditions while keeping the plug-in helped in some arms (16 of 112 unadjusted pointwise intervals excluded zero) but did not reach the reanalysis error. With the logit model, none of 112 alternative acquisition geometries resolved an advantage over balanced sampling. More calls can also hurt a misspecified estimator: the logit model's error for the mean rose from 2.2 pp at 256 calls to 3.5 pp at 1,024.

Table 7: PromptEval banks: absolute error (pp) of the released estimators by evaluations per (task, model), averaged over five seeds and task–model pairs. 5th-percentile values reproduce the released arrays; mean values rerun the released sampler and estimators because the released arrays contain only quantiles.

		5th percentile				mean			
bank	estimator	200	400	800	1,600	200	400	800	1,600
MMLU	plug-in	46.7	33.4	22.5	14.8	1.49	0.94	0.61	0.39
	PE	26.6	16.3	10.6	6.2	1.59	1.03	0.72	0.44
	PE-X	18.7	10.0	4.2	1.8	4.57	2.18	0.92	0.48
BBH	plug-in	20.7	20.6	15.5	10.2	1.83	1.26	0.86	0.61
	PE	12.5	12.0	10.0	6.9	2.02	2.17	1.44	0.72
	PE-X	7.1	4.6	4.0	3.9	2.98	1.55	0.92	0.63
LMentry	plug-in	7.2	7.2	6.4	4.7	1.80	1.31	0.86	0.58
	PE	6.3	5.4	4.5	3.5	2.03	2.15	1.74	0.75
	PE-X	7.0	8.6	9.3	9.5	3.28	1.65	0.95	0.59

Further PromptEval contrasts. Negative values favour reanalysis at b. MMLU: PE@200 − plug-in@800 +4.08 [1.93, 6.17]; PE-X@200 − plug-in@800 −3.74 [−5.64, −1.62]. BBH: PE@400 − plug-in@1,600 +1.72 [0.04, 3.54]; PE@800 − plug-in@1,600 −0.20 [−1.39, 1.15]. LMentry: PE@200 − plug-in@400 −0.93 [−3.70, 1.32]. On MMLU, PE at $b = 200$ and 400 is worth about 417 and 998 added plug-in evaluations per task–model for the 5th percentile (interpolated only where the plug-in curve is monotone).

C.3 Decisions

Design. On the cached MMLU banks we choose the best model by mean accuracy on three panels: the four acquisition models, six models (adding Mistral-7B and OLMo-2-7B), and the near-tie pair Mistral-7B versus OLMo-2-7B. The reference gap to the runner-up is 5.0 pp on the first two panels and 2.6 pp on the pair. Each of 200 replicates is one nested balanced call sequence shared by all models and rules, with looks at 128, 256, 512 and 1,024 calls per model; the reference winner is computed on the disjoint reference. We compare a *fixed-budget point winner*, Sequential Halving and Successive Rejects at a fixed total budget (Karnin et al., 2013; Audibert & Bubeck, 2010), a *fixed-look interval rule* that stops when simultaneous bootstrap max-t intervals (Bonferroni over the four looks) separate a winner and otherwise returns the unseparated set, and *paired elimination*, which uses the same intervals on common items to drop models as soon as another active model is significantly better (Nelson & Matejcik, 1995; Kim & Nelson, 2001). Neither interval rule is a confidence sequence, and we did not establish nominal coverage after selection; the rates are empirical frequencies. Adaptive rules are charged total calls divided by the initial number of models.

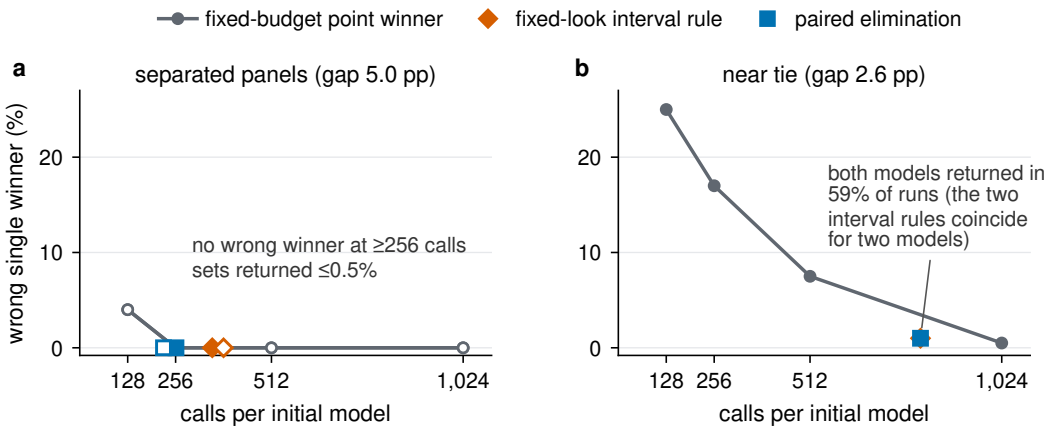

Figure 5: **Cost, wrong winners and unresolved sets in model choice.** Cached MMLU banks, mean accuracy, 200 replicates of shared call sequences; the reference winner is computed on a disjoint 4,096-item reference. x: total calls divided by the initial number of models. y: percentage of replicates returning a single wrong model. (a) Four-model (filled) and six-model (open) panels, reference gap 5.0 pp. (b) Near-tie pair, gap 2.6 pp; interval rules also return both models in 59% of replicates. Frequencies, not guarantees; 0/200 has a one-sided 95% upper bound of 1.5%. Full rates in Table 8.

Separated choices. On the four- and six-model panels, no rule returned a wrong single model at 256 calls or more (Fig. 5a). Paired elimination stopped at a mean of 257 and 224 calls per initial model and returned a two-model set in 0.5% of replicates; our fixed-look rule needed 354 and 384 calls. Validity bought through worst-case bounds can make stopping useless at these sizes: a union-bound Hoeffding elimination returned a set in every replicate, after 976–1,024 calls.

Near ties. On the near-tie pair a forced winner was wrong in 25% of replicates at 128 calls, 17% at 256 and 7.5% at 512, and Sequential Halving and Successive Rejects were wrong in 16.5–17% at 256. The two interval rules, identical for two models, stopped at a mean of 807 calls, returned a single wrong model in 1% of replicates and returned both models in 59% (mean set size 1.59). With

Table 8: Model-choice rules on identical cached MMLU call sequences (200 replicates). Point-winner, Sequential Halving (SH) and Successive Rejects (SR) always return one model; the fixed-look interval rule and paired elimination may return a set. Rates are percentages of replicates (0/200 has a one-sided 95% upper bound of 1.5%); calls are total calls divided by the initial panel size. Panels that share a bank are dependent. A union-bound Hoeffding elimination (mean target) returned a set in every replicate (976–1024 calls per initial model).

		wrong single winner (%)			fixed-look rule			paired elimination		
panel	target / estimator	@256	@1024	SH / SR @256	calls	wrong	set	calls	wrong	set
four models	mean / pooled	0.0	0.0	0.0 / 0.0	354	0.0	0.0	257	0.0	0.5
six models	mean / pooled	0.0	0.0	0.0 / 0.0	384	0.0	0.5	224	0.0	0.5
near-tie pair	mean / pooled	17.0	0.5	17.0 / 16.5	807	1.0	59.0	807	1.0	59.0
four models	bottom-10% / EB	0.0	0.0	0.0 / 0.0	795	0.0	74.0	596	0.0	74.5
six models	bottom-10% / EB	0.0	0.0	0.5 / 0.0	838	0.0	79.0	491	0.0	78.0
near-tie pair	bottom-10% / EB	17.0	3.0	17.0 / 17.0	872	1.0	82.5	872	1.0	82.5
four models	bottom-10% / linear	0.0	0.0	0.0 / 0.0	332	0.0	0.0	246	0.0	0.0
six models	bottom-10% / linear	0.0	0.0	0.0 / 0.0	374	0.0	0.5	220	0.0	0.0
near-tie pair	bottom-10% / linear	17.5	0.5	17.5 / 17.0	801	1.0	59.5	801	1.0	59.5

two candidates a set cannot exclude the best, and the point winner at the full 1,024 calls was wrong in only 0.5% of replicates.

A good point estimator can lack a usable interval. For the bottom-10% target with EB shrinkage, the point winner was almost always correct, yet the interval rules returned sets in 74–79% of separated-panel replicates: the item bootstrap distorts the estimated prior variance, biasing the shrunken values by 1–6 pp. With the linear main-effects estimator the same rules separated the winner at 220–374 calls per initial model.

Worst condition versus lower-tail mean. The identity of the worst of the 48 prompt conditions was not resolved at the tested acquisition and reference sizes: at 1,024 calls every estimator named the reference minimum in at most 6% of replicates, and two halves of the reference agreed on it in only 2–32% of draws. The mean of the lowest 10% of conditions was estimated within 2.3–3.7 pp. This lower-tail mean is a different functional, not a certificate for the worst-case claim.

D Reproducibility and material deviations

Artifact. The anonymous artifact contains the frozen specifications for each comparison, the response banks or their manifests, the analysis scripts, scripts that regenerate every result-derived figure and table in this paper from the bundled final result files, an index mapping each reported number to its source file, and the Reassess package (command `reassess-eval`) with tests and example configurations.

Deviations. Three deviations change how results should be read. (i) *Second corruption pair.* The planned pair (Gaussian noise×contrast) and its technical fallback (brightness×motion blur) were not available in our composable rendering stack, and adding new generators would have changed the stack relative to blur×JPEG. The nearest stack-native version of the primary pair (noise×gamma) failed the stack's model-free validity check (12 of 16 interior cells exceeded clipping or structural-similarity thresholds). Before any model response, we declared the nearest stack-native version of the fallback (gamma for brightness, Gaussian blur for motion blur) with a model-free acceptance rule; it passed on all rendered cells. The pair shares the blur axis with blur×JPEG; the gamma axis and all 16 joint cells are new. (ii) *Post-hoc additions.* The leave-one-checkpoint-out vision head and the new-family remedies were added after earlier outputs. (iii) *Post-hoc vision arms.* The target-cell and nine-axis-cell replication arms (Table 4) were added after the original comparison, with no new model responses: every target cell had already been rendered and scored for every item and model. They reuse its start states, allocation draws and resampling and do not change the status or criterion of the original comparison.

Table 9: Held-out check. A rule frozen on 295 development models chose among tinyBenchmarks' released estimators and was scored once on 100 held-out models. Error of the full-benchmark mean accuracy (pp); Δ is the chosen estimator minus direct sampling of n, $2n$ or $4n$ items [95% paired model-bootstrap interval]. Each row uses n queries per held-out model and no diagnostic queries, plus 8.45M dense calibration responses from the development models. At $n = 400$, $4n$ exceeds the 1,172 ARC-C items.

task	n	direct	chosen	error	Δ vs n	Δ vs $2n$	Δ vs $4n$
ARC-C	100	3.62	gp-IRT	1.44	-2.17 [-2.40, -1.91]	-0.99 [-1.23, -0.72]	-0.09 [-0.34, 0.18]
	200	2.43	p-IRT	1.18	-1.25 [-1.47, -1.00]	-0.35 [-0.57, -0.11]	$+0.43$ [0.21, 0.68]
	400	1.54	p-IRT	0.86	-0.68 [-0.84, -0.48]	$+0.11$ [-0.06, 0.30]	—
HellaSwag	100	2.91	gp-IRT	1.31	-1.60 [-1.84, -1.34]	-0.73 [-0.98, -0.47]	-0.12 [-0.37, 0.14]
	200	2.05	gp-IRT	0.86	-1.19 [-1.36, -0.98]	-0.58 [-0.75, -0.36]	-0.13 [-0.31, 0.09]
	400	1.44	p-IRT	0.55	-0.89 [-1.00, -0.75]	-0.44 [-0.56, -0.31]	-0.13 [-0.24, 0.01]

Reassess. The input is a response bank (models $\times$ conditions $\times$ items in $[0, 1]$) with item strata and optional condition factors, and a YAML declaration of target, estimators, acquisition plans, candidate actions and evidence level. `reassess-eval run` dispatches to backends that call released code unchanged: PromptEval's sampler, PE and PE-X; the tinyBenchmarks anchor-IRT, p-IRT and gp-IRT estimators; and this paper's MMLU estimators, vision completions and calibration remedies. A method without such a backend is reported as unsupported. The action card lists, for each action, the target error with paired intervals, the queries per evaluated system including diagnostic queries, and information beyond queries separately. In practitioner mode the reference-dependent fields are `NOT_ASSESSABLE`; on the MMLU bottom-10% quickstart the card still shows that the plug-in and shrinkage estimates from the same 128 calls differ by more than 30 points for every model. A backend refuses to run if evaluated and calibration systems overlap. Unit tests cover plan nesting, weighting, nested estimator selection, cost accounting, completion rules, `NOT_ASSESSABLE` fields, decision sets and overlap refusal.

External held-out check. We used the tinyBenchmarks response bank of the Open LLM Leaderboard (395 models, 28,659 items) (Maia Polo et al., 2024a). Before reading any outcome we fixed two tasks (ARC-Challenge, HellaSwag), the target (each model's full-benchmark mean accuracy), budgets of 100, 200 and 400 items, the actions (uniform item sampling at n, $2n$ and $4n$, and the released anchor-IRT, p-IRT and gp-IRT estimators on n anchors), the released tutorials' model split (the 100 models the tutorials hold out as recent for testing; the other 295 for development) and the rule: take the variant with the lowest development-validation error and recommend it only if its paired advantage over sampling n items has a 95% interval below zero. The IRT model, anchors and weights were refitted on development models only, because the released anchors and seed had been selected using test-set performance. The rule recommended an IRT estimator in every cell and was scored once (Table 9). It beat sampling at equal queries in all six cells, including on the 69 test models from organisations absent from development; it beat twice the items in five cells, and against four times the items it was unresolved in four and worse in one. The advantage is tinyBenchmarks' published result, rechecked with a development-only refit; Reassess contributes the declaration, the cost accounting and the frozen comparison. The saving in queries on new models is paid for with 8.45M dense calibration responses and an IRT fit of 18.5 minutes. In the existing-bank regime, where the bank was already collected for the leaderboard, the resolved marginal saving is n queries per new model and task in the five cells that beat $2n$. If the bank had to be built for this evaluator alone, an illustrative query-only break-even would need at least 1,729 new models on one task counting only that task's share of the bank, and at least 21,137 counting the whole six-scenario bank; on the 100 test models the largest resolved saving was 60,000 queries. The split is by model, not by base model, and the model-selection target was not run because no candidate panel had been fixed in advance.