Measure the instrument first: detection floors for model selection and for LLM judges

Anonymous Author(s)
Affiliation
Address
email

Abstract

An evaluation reports that a candidate improved a metric. It almost never reports the smallest improvement it could have detected, and those are different claims. We measure the second on a deployed forecasting system with a construction that needs no distributional assumption: the *null lever*, the same recipe refit under a different random seed, changes nothing real, so everything it produces is instrument noise. On a $3{,}087$-row walk-forward pool under five seeds, ten null comparisons manufacture effects up to 0.00207 nats, while the only lever this leg ever shipped is worth 0.0026: re-seeding alone reproduces 80% of it, and the pipeline had 56% power against its own shipped result. That comparison assumes nothing, and we separate it from three readings that do. A variance decomposition puts 63% of the paired noise beyond any seed budget, but on four degrees of freedom that share is bounded only to $[17\%, 83\%]$, so we report it as an estimate rather than a finding. Charged the multiplicity this project's sister audit actually spent, the same floor rises by $1.64\times$. And, most consequentially, **a floor belongs to the instrument that measured the effect**: recomputed on the 793-row pool where this project's one large win was measured, the floor is 0.0119 rather than 0.00362, making that win $4\times$ its own floor and not $13\times$ another pool's, an error we made in an earlier draft of this paper. The protocol transplants: on a deployed LLM classifier, two identical reruns disagree on 5.6% of items and twenty of them move measured agreement by 2.6 points, putting the floor at 2.9 points. We give the protocol, the arithmetic, and five in-the-wild verdicts read from below a floor.

1 The number that is missing from the report

Evaluation reports are asymmetric. They state what changed and by how much; they rarely state what the protocol could have resolved. Without the second number a reader cannot separate three situations that demand different responses: a real improvement, a real improvement too small to have been seen, and noise. In our experience the third is the most common and the least often named.

More error bars will not supply it. An error bar on a single arm describes the spread of the metric, while a candidate must clear the spread of the *difference* under a change that does nothing. Those are different quantities. That quantity is directly measurable, and measuring it is cheap.

We report it for a deployed sports-outcome forecasting system whose model-selection pipeline has been in continuous use for three months, has evaluated several dozen candidate levers across nine documented studies, and has shipped exactly one on the leg studied here. The system is a gradient-boosted ensemble over 118 point-in-time features; selection runs on a walk-forward out-of-fold pool

Submitted to 40th Conference on Neural Information Processing Systems (NeurIPS 2026). Do not distribute.

built from 32 quarterly refits between 2017 and 2024, so a candidate is scored only on rows no fold of it was trained on. None of that matters to the argument. What matters is that the pipeline is stochastic, that it was run five times under five seeds, and that those five runs are enough to say what it can see.

Contributions. (i) The null lever, a protocol for measuring an evaluation instrument's detection floor with no distributional assumption and no experiments beyond re-seeding. (ii) The finding that the largest effect a change-that-does-nothing produced on this pipeline is 80% of the only improvement its winner leg ever shipped, and that the pipeline had 56% power against its own shipped result. (iii) A variance decomposition that says which purchase raises power, reported with the interval its four degrees of freedom permit, wide enough that we mark the conclusion a point estimate rather than a finding. (iv) The insistence, worked through three pools, that *a floor belongs to the instrument that measured the effect*: carrying one pool's floor to another pool's result overstated this project's own best win threefold. (v) Five documented in-the-wild verdicts read from below a floor. (vi) The protocol transplanted to a deployed LLM classifier, where the levers that change nothing are reruns, batch composition and prompt order, and where the same variance structure meets an opposite cost structure.

Related work. That benchmark conclusions are unstable under nuisance variation is well established. Dodge et al. [2019] showed reported results depend on search budget and asked for it to be published; Bouthillier et al. [2021] decomposed benchmark variance into its sources and argued for randomising more of them; Card et al. [2020] computed statistical power for NLP experiments and found much of the literature underpowered; Gorman and Bedrick [2019] showed conclusions drawn on a standard split often fail on random ones. The finding has since been reproduced for language models: Madaan et al. [2024] quantify seed variance across initialisations on evaluation benchmarks, and Hochlehnert et al. [2025] trace reported reasoning gains to decoding parameters, seeds, prompt formatting and hardware. Closest to us in form, Kotawala [2026] treats paired LLM comparison as hypothesis testing and reports a per-pair resolution ratio $q = N/N^\star$, finding many displayed leaderboard comparisons unresolved.

Our contribution is not the construction. Bouthillier et al. [2021] already rerun the pipeline itself (seeds, weight initialisation, data order), and what they estimate is pipeline variance in exactly our sense; Madaan et al. [2024] and Hochlehnert et al. [2025] do the same for language models. What we add is what happens to the estimate afterwards. Those works report variance to argue that a reported effect may be noise. We invert it into one number a report prints beside each decision, charge the multiplicity a *selection* pipeline spent rather than the multiplicity a test declared (the charge Benjamini and Yekutieli [2001] formalise for testing, applied where it is usually not applied at all), insist that a floor belongs to the instrument that measured the effect rather than to the project, and give five in-the-wild cases where a deployed pipeline read a verdict from below its own floor.

The item-resampling objection applies to one neighbour specifically. Kotawala [2026]'s resolution ratio is the closest published relative of our effect/floor ratio, and it is estimated by bootstrapping evaluation items. That holds the refit fixed while the items vary, so it cannot see refit stochasticity — 37% of the paired variance on the pool below, and the term that decides which purchase raises power.

2 The null lever

Let a *lever* be any change to a recipe that a pipeline may accept or reject, and let $\widehat{\Delta}$ be the paired difference in the selection metric between the levered and unlevered arms on the same evaluation rows. A *null lever* is a change that alters no modelling decision: here, the random seed passed to the learners.

Under a null lever $\mathbb{E}[\widehat{\Delta}] = 0$ by construction. Its realised distribution is therefore the pipeline's noise distribution, measured rather than modelled, and two different thresholds follow from it. An *observed* effect must exceed $z_{1-\alpha}\,\mathrm{SE}(\widehat{\Delta})$ to be called a detection at all. A *true* effect must be at least

$$\mathrm{MDE} \;=\; \left(z_{1-\alpha} + z_{\mathrm{power}}\right) \cdot \mathrm{SE}(\widehat{\Delta}), \tag{1}$$

for that to happen with probability power. The two are 0.00239 and 0.00362 below — a significance boundary and a planning number. We call the second the floor and quote both, since an effect

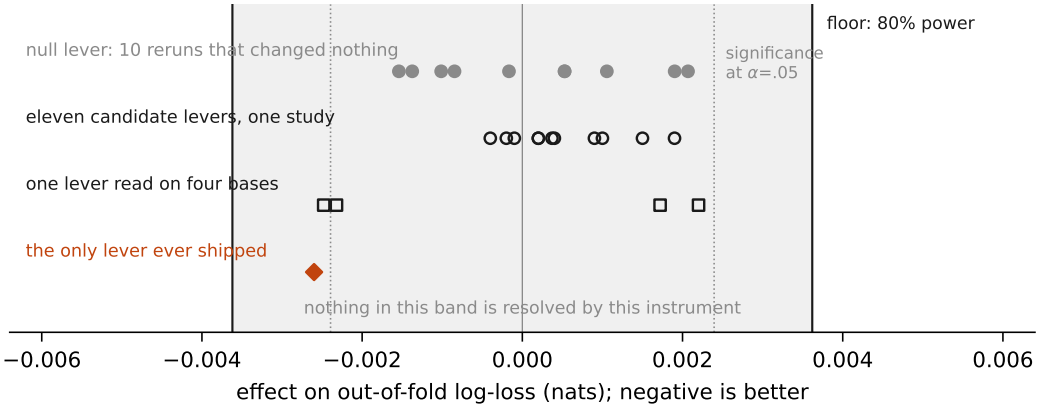

Figure 1: Everything this pipeline has measured on the main pool, against what a change that does nothing produces. Grey: the ten null-lever comparisons. Open circles: the eleven candidate levers of one study. Squares: one lever read on four selection bases, which reversed sign. Diamond: the only lever the leg ever shipped. No point in the figure lies outside the floor, and the largest null-lever effect (0.00207) is larger than the largest candidate effect (0.0019).

quantity	value
evaluation rows (walk-forward, out-of-fold)	3,087
per-seed metric, five seeds	0.64791 – 0.64998
sd of the metric across seeds	0.00088
unpaired SE of the metric	0.00462
paired SE, row component	0.00116
null-lever comparisons	10
largest effect a null lever manufactured	**0.00207**
variance that is row-level (no seed budget touches it)	63%
variance that is seed-level	37%
detection floor, one seed	0.00362
detection floor, five seeds	0.00304
detection floor, infinite seeds	0.00288
detection floor, one seed, 84 questions	0.00594
the only lever this leg ever shipped	**0.0026**
largest effect the system has resolved (method leg, §6)	0.048

Table 1: The instrument, and the two candidates worth putting beside it. Metric is out-of-fold log-loss in nats; lower is better; floors are one-sided at $\alpha = 0.05$ and 80% power. Every figure except the two in bold is recomputed from the pool by the script that also reconciles it against the lab's own report; those two are effect sizes quoted from the lab reports that measured them.

between them is detectable, just not often. SE is estimated from the null lever itself. Five seeds give $\binom{5}{2} = 10$ null comparisons at no cost beyond reruns a careful project already performs.

Two properties make this preferable to the alternatives. It requires no assumption about the metric's sampling distribution, because it draws from the real one. And it is *paired on evaluation rows*, which removes the shared difficulty of the examples, the dominant term in any absolute metric and the term that makes single-arm error bars look reassuring while saying nothing about the comparison actually being made. On this pool the unpaired standard error is 0.00462 and the paired one 0.00116: a fourfold difference, and the unpaired number is the one a naive report would quote.

3 What this instrument can see

Table 1 is the measurement and Figure 1 is the argument. Read the line above the floor first: **the largest difference produced by a change that did nothing is** 0.00207 **nats**, and the only improve-

ment this leg has ever shipped is worth 0.0026. Re-seeding reproduces 80% of the project's single genuine winner-leg result. Had that lever been evaluated once, under one seed, against one baseline, its measured effect would have been hard to tell from a rerun.

The floor is conservative against these levers, not generous. The null lever compares two *independent* seeds, while the eleven levers of Section 7 and the shipped lever were each scored against a baseline at the *same* seed, where the seed term is shared and partly cancels. The floor applying to them therefore lies between the cross-seed 0.00362 and the seed-free asymptote of 0.00288. The shipped lever's 0.0026 is below both, so nothing central turns on which end is right.

One consistency check is worth stating, because the decomposition in the next section depends on the noise being as ordinary as we assume. The largest null-lever effect is the range of the five per-seed metrics, and the expected range of five draws from a normal distribution is 2.33σ. With the measured $\sigma = 0.00088$ that predicts 0.00205; the observed range is 0.00207, within 1%. The spread behaves like five draws from the distribution the sd describes, which is the cheapest available evidence that Equation 2 is describing this pipeline rather than being imposed on it.

4 Buying seeds walks to a floor, not past it

The standard response to a noisy pipeline is to average more seeds. Decomposing the null lever's variance says how much that helps. Writing σ^2_{row} for the component that comes from which rows the pool happens to contain and σ^2_{seed} for training stochasticity, averaging k seeds per arm gives

$$\text{SE}_k \;=\; \sqrt{\sigma^2_{\text{row}} + \sigma^2_{\text{seed}}/k}\,, \tag{2}$$

which has a horizontal asymptote at $\sqrt{\sigma^2_{\text{row}}}$. On this pipeline $\sigma^2_{\text{row}} = 1.34 \times 10^{-6}$ and $\sigma^2_{\text{seed}} = 7.80 \times 10^{-7}$: 63% **of the variance is untouchable by any seed budget**. The floor falls from 0.00362 at one seed to 0.00288 at infinitely many — a factor of 1.26. The five seeds actually run reach 0.00304; every remaining seed in the world is worth 0.00016 nats after that (Figure 2, left).

What four degrees of freedom establish. Five seeds estimate σ^2_{seed} on four degrees of freedom, where a 95% interval spans a factor of 23. Propagating it, the row share is 63% with an interval of $[17\%, 83\%]$, and the single-seed floor is 0.00362 with an interval of $[0.00316, 0.00694]$. Two claims survive that interval and one does not, and the paper is better for separating them. The shipped lever at 0.0026 sits below the floor across the *entire* interval, so that comparison holds however σ^2_{seed} actually falls. The largest null-lever effect, 0.00207, is an observed maximum and carries no interval at all: it is the assumption-free half of this paper and the number to argue from. But *which purchase to make* rests on the share, and the share is not established: its interval straddles 50%, so we mark that conclusion as a point estimate rather than a finding. Narrowing it costs seeds, which is the one thing seeds are reliably worth buying.

The practical reading is a budgeting rule. If a candidate's expected effect is below the asymptote, no number of reruns will make it visible, and the only purchases that help are more evaluation rows or a lower-variance metric. If it is between the asymptote and the one-seed floor, a handful of seeds is exactly the right purchase and a large number is waste. A project that does not know which regime it is in will buy seeds in both, and in the first regime it will buy them forever.

5 Every question the pipeline asks raises the floor

A detection floor is usually quoted as though it were a property of the dataset. It is a property of the dataset *and* the number of questions put to it. A pipeline that evaluates one pre-specified candidate and one that searches eighty-four are not the same instrument, and the difference is large enough to change verdicts.

The same project ran a pre-registered search of 84 hypotheses — on a different pool of $1{,}787$ priced bouts, not on this one. The transfer is a counterfactual and we mark it as one: *were this leg charged the multiplicity its sister audit actually spent*, the per-test critical value would rise from 1.645 to 3.242 and the floor from 0.00362 to 0.00594 nats, a factor of 1.64 on top of everything in Section 4 (Table 2, Figure 2 right). At that charge the floor sits above *every* candidate this leg has ever evaluated, including the one it shipped. The counterfactual is not idle: the two pools belong to one project, and a question asked of either is a question asked of the model both are scoring.

questions asked	1	6	12	84
detection floor, one seed	0.00362	0.00471	0.00507	0.00594
relative to a single question	1.00×	1.30×	1.40×	1.64×

Table 2: The multiplicity charge a floor calculation admits, on the same pool. Six is the size of the calibration comparison in Section 7; twelve is an unremarkable ablation table; 84 is the size of the search this project ran on its other pool, carried here as a counterfactual.

This is the number most often missing from a leaderboard or an ablation table. An ablation with twelve rows is a twelve-hypothesis search whether or not it is described as one, and quoting a single-comparison floor overstates what the table supports by roughly 40%. The correction is arithmetic and costs nothing; not making it is a choice about what the reader may conclude. Bonferroni is the charge a floor calculation admits, and adaptive procedures recover power against it, so these factors bound the charge from above.

6 The instrument says yes, too

A floor exists to be cleared as well as missed, and a paper that only reported refusals would be describing a broken pipeline instead of a working one. The same *construction*, applied as it must be to the instrument that did the measuring, resolves the one large effect this project has found.

About 19% of the slate reached production with no conditional method model at all: the model that predicts *how* a fight ends had never been fitted on rows where one fighter is making a debut, and those bouts took their numbers from a simulator running on defaults. Fitting that segment is worth -0.048 nats on 793 walk-forward rows, sign-stable across the three seeds it was run under.

That effect has to be judged against its own instrument, not this paper's. It was measured on a different pool with a different metric, a three-class log-loss averaging 0.96 nats against the winner leg's 0.65. And a floor belongs to the instrument that produced the number, not to the project that owns both. Running the null lever on *that* pool gives a paired SE of 0.00331, a seed sd of 0.00345, and a floor of $\mathbf{0.0119}$: more than three times the main pool's. The 0.048 is 4.0 times its own floor, not the 13.3 times that quoting Table 1's figure would have produced. Carrying this paper's headline floor to a result measured elsewhere would have overstated it threefold, which is precisely the error the paper exists to name, and we made it in an earlier draft.

The assumption-free comparison needs one correction first, and it is the same lesson one level down. The largest effect a null lever manufactured there is 0.00676, but that is a *range over three seeds*, where the main pool's 0.00207 is a range over five, and a range grows with the number of draws taken. Section 3's own check supplies the conversion: the expected range of k normal draws is $d_k\sigma$, with $d_3 = 1.69$ against $d_5 = 2.33$. On the main pool's footing the debut null is 0.0093 and the effect is 5.2 times it, not the 7.1 the raw maximum gives. **No statistic travels between instruments unless it is put on the same footing — the null-lever maximum no less than the floor.** We would have led with the 7.1, and it flatters for precisely the reason a maximum over three draws does.

Corrected, the two routes agree where it matters and the modelled one barely constrains: on three seeds its interval is $[0.9, 5.1]$, wide enough to include *unresolved*, while the assumption-free figure is 5.2. We rest the claim on the second, as Section 4 said we should.

The contrast is the point. The effect is large for a plain reason: it supplies a model where there was none, and the modelling itself is unremarkable. An instrument that cannot separate 0.0026 from a rerun separates 0.048 without difficulty, and a report printing the floor beside both makes that legible at a glance rather than after an argument.

7 What happens below the floor

An argmin over five families. Before the walk-forward pool existed, this system selected on a held-out split of 429 rows. A calibration study fit six transform families on that split and read its argmin:

family (parameters)	selection metric	test metric	selection is right
identity (0)	0.6203	0.6198	—
temperature (1)	0.6171	**0.6165**	89.6%
cubic (2)	0.6170	0.6173	55.0%
beta (2)	0.6164	0.6181	—
beta$_{c_0}$ (2)	0.6166	0.6171	61.6%
piecewise (3) — *selected*	**0.6163**	0.6193	36.5%

The split selected the piecewise family, the worst of the five non-trivial candidates on test and barely better than doing nothing. The last column, which resamples the selection split, refits, and re-scores the same test set, shows this was not bad luck: at one parameter the selection is right 89.6% of the time, at two it is a coin flip, and at three it is worse than not selecting at all.

We stress what this case is evidence for. It is not that the candidates were bad, nor that the metric was wrong: every arm is a legitimate calibrator and the metric is the deployed one. The failure is one of resolution, and on this pool it can be quoted rather than asserted. The five non-trivial families are separated on the selection split by 0.00087 nats in total — that is the entire signal the argmin had to work with. The study's own bootstrap says what noise sat on top of it: resampling the 429-row split, refitting, and re-scoring the same test set moves the resulting test log-loss over a range of 0.0050 for the one-parameter family and 0.0236 for a two-parameter one. **The quantity being selected on is between six and twenty-seven times smaller than the movement in what the selection produces.**

No null lever is available here — the calibrators are deterministic, and there is no seed to vary. Split resampling is the same construction reached by another route: refitting on a resampled selection split changes nothing about which family is truly best, so everything it produces is instrument noise. The study ran exactly that and printed the answer in the last column above, and the pipeline selected anyway. A floor read before the argmin would have said a six-way comparison at this separation was not affordable, and it would have said so while the answer was still hidden.

Eleven levers, all under the floor. The study that built the walk-forward pool evaluated eleven levers on it (both fighter orderings, a leaked feature dropped, two recency weightings, two LightGBM variants, seed bagging, three age-throttle settings), with effects from -0.0004 to $+0.0019$ nats (Figure 1, open circles). Every one is below the floor, and the largest is below what a null lever manufactured. Their bootstrap "fraction of resamples that improve" runs from 8% to 80% and reads like information; against this floor it is the shape of the noise. One of the eleven is seed bagging, a null lever wearing a lever's name, and it came back where a null lever should.

A sign that flips between bases. The most recent study fit one coefficient for a demographic term in a post-blend corrector. The bias it targets is real, large and stable across three eras. Three selection bases passed it: cross-fit -0.00232, forward-chained -0.00248, held-out $+0.00172$. A rolling retrain on identical origins then returned $+0.0022$. All four readings are under the floor. The reading is therefore not "three bases beat one": it is that nothing was resolved, and that a sign flipping between bases is the same condition as a sign flipping between seeds. The lever was refused.

What replaced argmin-on-one-split is agreement in sign across partly independent readings. Agreement is weaker than power, since four correlated readings of an underpowered instrument still fall short of four experiments. It catches what a single reading cannot, which is exactly how the demographic term was caught.

8 The same data, a different protocol

The floor says what an instrument can resolve. It does not say the verdict is unique, and on this system it is not. The same forecaster was audited against a betting market's closing line on a second pool of $1{,}787$ priced bouts, under 84 pre-registered hypotheses about where the model beats the price [Anonymous, 2026]. That audit is worth one paragraph here because of what it does to the present argument.

That analysis was fixed-n, with multiplicity charged by Benjamini–Hochberg, and it computed and published its own power: 10–15% against a three-point edge, 0–1% *once multiplicity was charged*, and $\sim 89{,}900$ bouts needed against the $1{,}787$ available. The answers were read anyway. Re-asked as a sequential test the same data give different verdicts — the one favourable discovery fails under

quantity	value
items scored by all twenty-six runs	500
agreement with the shipped labelling, twenty identical reruns	0.8660 – 0.8920
spread across twenty runs that differ by nothing	**0.0260**
items whose label is not stable across the twenty	18.0%
two identical reruns disagree on	5.6% of items
variance that is item-level (no rerun budget touches it)	74% [58, 83]
variance that is run-level (no corpus size touches it)	26% [17, 42]
detection floor, one run per arm	**0.0293** [0.0276, 0.0332]
detection floor, infinite runs per arm	0.0252
detection floor, one run, six prompt variants tried	0.0381
batch composition, as a candidate lever	−0.0050 (0.17×)
category order, as a candidate lever	−0.0010 (0.03×)

Table 3: The same protocol on a deployed LLM classifier. Metric is agreement with the shipped labelling; floors are one-sided at $\alpha = 0.05$ and 80% power; the bracketed figures are effect / floor.

the procedures whose guarantee actually holds for overlapping slices of one pool, and a segment reported as failing to replicate at $p = 0.30$ turns out to be short of its threshold rather than refuted.

The lesson this paper takes has little to do with e-values. **Computing the diagnostic is not the binding constraint.** There the power calculation existed, was correct, and sat in the same document as the conclusions it undercut; here no such number existed until nine studies had shipped verdicts. Different failure, same outcome, which is why Section 10 puts the number beside the decision rather than in a methods appendix.

9 The same protocol, an LLM judge

The construction needs a stochastic pipeline and a paired evaluation. It does not need gradient boosting. We ran it unchanged on a second deployed system in the same project: a news classifier that assigns each incoming article one of eight categories, built on a hosted LLM (`claude-haiku-4-5`, called through a structured-output schema), in production for months, with 2,337 items classified in the live database at the time of the run.

Here the lever that changes nothing takes three forms. A correct classifier is invariant to all three.

- **Rerun.** The identical request, sent again. Production sets no temperature, so the provider default makes a repeated call genuinely stochastic.
- **Batch composition.** Items are classified ten to a request; which nine items an item is scored beside is an implementation detail.
- **Category order.** The eight category definitions appear in a fixed order in the system prompt, and the order is arbitrary. Only the sequence is permuted; each definition's wording is byte-identical.

We sampled 500 items and ran twenty identical reruns, three batch reshuffles and three category permutations: twenty-six runs, about seventeen minutes of wall clock (Table 3). Twenty rather than five because reruns here cost seconds and cents, and Section 4's decomposition was the weakest claim in this paper for want of degrees of freedom: at twenty runs the item share is established rather than merely estimated. The metric is agreement with a fixed reference labelling, the label production shipped. Because the reference is held fixed across every run it cancels from every paired difference, so the floor measures the pipeline and not the reference's quality; recomputing against the reruns' own majority vote moves the floor from 0.0293 to 0.0328 and the item share from 74% to 63%, both inside the interval quoted below, and both in the direction one expects when the reference is itself a smoothed draw from the process being measured.

What a rerun does. Twenty identical runs score between 0.8660 and 0.8920 — a spread of 2.6 points with nothing changed between them. 18.0% of items do not hold a stable label across the twenty, and two identical runs disagree on 5.6% of items. That last figure is the one to carry: the

share of unstable items grows with how many reruns you look at, while pairwise disagreement does not, and it was 5.6% at five runs too. The one-run detection floor is 2.93 points. *Any prompt change reported as a two-point improvement on five hundred items is not a result.*

What the two levers do. Batch composition moves the metric by -0.50 points and category order by -0.10, or 0.17 and 0.03 of the floor. Neither is resolved. That is the outcome a null lever should produce, It checks that the construction is not manufacturing effects; it does not prove the two are zero.

The structure is the same; the affordable purchase differs. It would be neat to report that this pipeline needs the opposite purchase to Section 4's, and it does not: both put most of their variance in the data-sampling term (63% row-level there, 74% item-level here), so both should buy evaluation data first. Resolving a one-point improvement needs $3{,}186$ items on that term alone, against the $2{,}337$ the corpus held. What differs is the second term and what it costs to buy down. Run noise alone puts 1.48 points on the floor, so from a single run per arm a one-point effect is unresolvable at *any* corpus size — but a rerun costs forty seconds where a refit costs a pipeline run. The structures rhyme; the budgets do not.

Relation to judge-reliability work. Yagubyan [2026] reports pairwise preference reversals averaging 13.6% for LLM judges, and that eleven repeated trials are needed for a majority vote to recover a fifty-trial reference verdict. Our 5.6% pairwise disagreement is the classification analogue at the item level. What the null lever adds is the step from item instability to a metric-level floor, the number a prompt-engineering session has to clear, on the corpus it actually has, after charging the variants it actually tried.

10 A reporting recipe

The protocol is four lines and we would like to see it become unremarkable.

1. **Run a null lever.** Re-run the pipeline changing only something that changes nothing — a seed, a rerun, an item's batch, the order of options in a prompt. Report the largest $|\widehat{\Delta}|$ it produces. This one number is often enough to end a discussion.
2. **Decompose.** Split the null lever's variance into the part a budget can buy and the part it cannot. Report the share that reruns cannot touch, and the asymptotic floor.
3. **Charge the questions.** Raise the critical value by the number of comparisons the report actually contains, ablation rows included.
4. **Report the ratio.** Next to every claimed improvement, print effect / floor. A ratio below 1 is not a result, however small its p-value.

The construction needs only a stochastic pipeline and a paired evaluation, the situation for anything evaluated under sampling, whether the sampling is over seeds, data order, decoding temperature, prompt order, or a judge. Section 9 is these four lines on a different pipeline, and the arithmetic did not change.

11 Limitations

The decomposition assumes additive components and a seed term shrinking as $1/k$; both are standard, and their uncertainty is quantified in Section 4 rather than deferred here. Equation 1 is a normal approximation, though the standard error entering it is measured, and the multiplicity charge is Bonferroni-flavoured and an upper bound.

The LLM arm is one classifier, one model and one corpus; a different pairing needs its own measurement, which is the position we are arguing for. Its reference is a product of the same pipeline, which does not bias a paired difference, the reference being held fixed across runs, but does mean the level of agreement is not accuracy against truth. And all of this is one project: these cases illustrate the asymmetry, they do not estimate how often it bites. Establishing that needs the null lever run where it has not been, which is the point.

References

Anonymous. Internal evaluation reports for the system studied here, 2026. Prior work by the present authors; withheld for double-blind review and supplied on request.

Y. Benjamini and D. Yekutieli. The control of the false discovery rate in multiple testing under dependency. *Annals of Statistics*, 29(4):1165–1188, 2001.

X. Bouthillier, P. Delaunay, M. Bronzi, et al. Accounting for variance in machine learning benchmarks. In *MLSys*, 2021.

D. Card, P. Henderson, U. Khandelwal, R. Jia, K. Mahowald, and D. Jurafsky. With little power comes great responsibility. In *EMNLP*, 2020.

J. Dodge, S. Gururangan, D. Card, R. Schwartz, and N. A. Smith. Show your work: improved reporting of experimental results. In *EMNLP*, 2019.

K. Gorman and S. Bedrick. We need to talk about standard splits. In *ACL*, 2019.

A. Hochlehnert, H. Bhatnagar, V. Udandarao, S. Albanie, A. Prabhu, and M. Bethge. A sober look at progress in language model reasoning: pitfalls and paths to reproducibility. In *COLM*, 2025.

A. Kotawala. Resolution diagnostics for paired LLM evaluation. arXiv:2605.30315, 2026.

L. Madaan, A. K. Singh, R. Schaeffer, A. Poulton, S. Koyejo, P. Stenetorp, S. Narang, and D. Hupkes. Quantifying variance in evaluation benchmarks. arXiv:2406.10229, 2024.

A. Yagubyan. The coin flip judge? Reliability and bias in LLM-as-a-judge evaluation. arXiv:2606.13685, 2026.

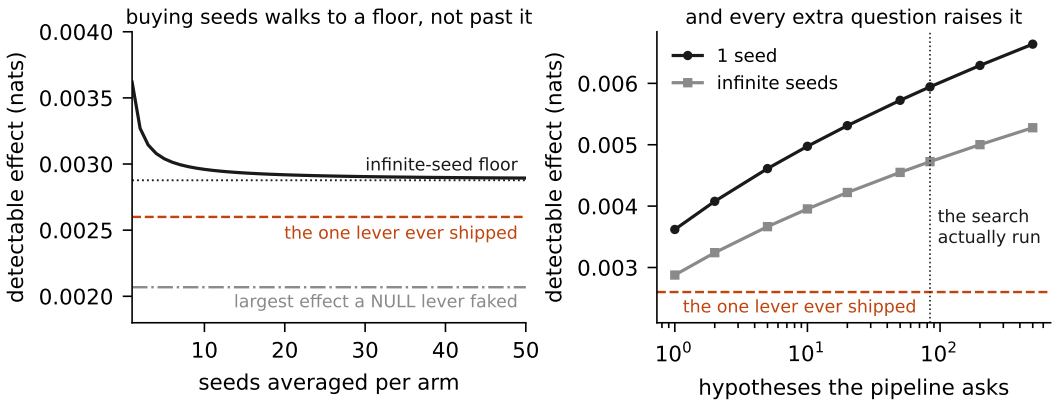

Figure 2: Left: the detection floor against the number of seeds averaged per arm. The curve flattens quickly because most of the variance is not seed variance, and the whole available gain between one seed and an infinite budget is $1.26\times$. Right: the floor against the number of hypotheses the pipeline asks, at one seed and at an infinite seed budget. The shipped lever and the eleven levers of Section 7 sit below almost every point on both panels; the one resolvable effect the system found sits far above them, off the top of both.

A Reproducing the floor

Pool. Walk-forward out-of-fold, 32 quarterly origins from 2017-01 to 2024-10. Per origin: train on events before origin $-$ 12 months, use the trailing twelve months for early stopping and blending, score the following three months. 3,087 scored bouts per seed; seeds 7, 13, 42, 99, 2024; 15,435 baseline rows in total. No fold of a candidate is scored on rows it trained on.

Estimator. With $\ell_i(s)$ the per-row log-loss of seed s on row i, the row component is the paired standard error of $\ell(s)-\ell(s_0)$ against a fixed reference seed $s_0 = 7$, averaged over the four remaining seeds; the seed component is the sd of the five arm-level metrics. Then $\mathrm{SE}_1 = \sqrt{\sigma^2_{\text{row}} + \sigma^2_{\text{seed}}}$ and Equation 1 follows with $z_{0.95} + z_{0.80} = 2.4865$.

Reconciliation. The lab report this pipeline maintains gives $\sigma_{\text{seed}} = 0.00088$, $\sigma_{\text{row}} = 0.00116$, and floors of 0.00363 and 0.00288. Recomputing from the pool gives 0.000883, 0.001157, 0.00362 and 0.00288. The single-seed floor differs in the last digit because that report rounds $z_{0.95} + z_{0.80}$ to 2.49; the exact constant is used here.

Robustness to the reference seed. Estimating the row component from all $\binom{5}{2} = 10$ pairs rather than the four against s_0 gives 0.001173 instead of 0.001157, moving the single-seed floor by less than 0.4%.

The debut pool. The 0.048 of Section 6 is measured on a separate walk-forward pool: 793 rows, three seeds, and a three-class log-loss over {KO, submission, decision} rather than the winner leg's binary one. Same estimator, its own rows: paired SE 0.00331, seed sd 0.00345, floor 0.0119. The mean per-row loss there is 0.96 nats against the winner pool's 0.65, which is most of why the floor is three times higher — a point worth making explicitly, since rescaling the main pool's floor by $\sqrt{3087/793}$ alone would have given 0.0061 and still been wrong.

B Reproducing the LLM arm

Sample. 500 items drawn from the 2,337 classified in the production database, ordered by a hash of the item identifier so the sample is not a recency or topic slice. Every item was labelled by all twenty-six runs; none were dropped.

Runs. Twenty identical reruns, three batch reshuffles, three category permutations. Requests carry ten items, exactly as production sends them, through the same structured-output schema; no temperature is set, in production or here, so the provider default governs. The permutation lever splits the system prompt on the eight category markers and reorders the blocks, leaving each definition

byte-identical. Twenty-six runs took $1{,}045$ seconds. The reruns were collected in two batches; the second pins the sample by item identifier rather than re-drawing it, because the corpus grew between them and a hash-ordered redraw would have returned a different five hundred.

Metric and reference. Agreement with the labelling production shipped, averaged over items. Because that reference is fixed across every run it cancels from each paired difference; scoring instead against the majority vote of the twenty reruns gives 0.0328 against 0.0293.

Estimator. Identical to Appendix A with reruns in place of seeds: the item component is the paired standard error of the per-item indicator difference between two reruns, and the run component is the sd of the twenty run-level agreement rates. Nineteen degrees of freedom put the item share at 74% with a 95% interval of $[58\%, 83\%]$, against the four degrees of freedom and $[17\%, 83\%]$ the seed arm has to work with.