Below the Reliability Floor: Recovering True Success from Judge-Gated Loops

Anonymous authors
Paper under double-blind review

Abstract

LLM judges are increasingly placed *inside* an agent's loop, scoring the agent's own attempts and re-prompting until one passes. We show this quietly corrupts measurement: retry-until-PASS is *optional stopping* against a noisy classifier—it keeps drawing until the judge slips—so the reported pass rate is an upward-biased estimator of true success. We make this exact. The cap-K gate is a binary classifier with closed-form sensitivity/specificity, and its bias is governed by one coefficient, the gate's Youden index J: as $J \to 0$ the gated rate becomes uninformative about π, so recovery must fall back to gold labels and no estimator beats the gold-only mean. Across 44 capable-agent loops on GSM8K, MATH, and code with objective ground truth (no authored weakness; a separate terse-agent stress set is excluded here) the inflation is systematic (median slip +0.16; worst on code, where the judge cannot run the candidate, a true 0.74 inflated to a reported 0.98) and obeys a closed-form law predicting the slip from per-attempt statistics (pooled r=0.95; errors-in-variables slope $0.765\,[0.70, 0.84]$, excluding 1). To recover true success we benchmark Rogan–Gladen against prediction-powered inference: **PPI++ dominates in aggregate** (mean recovery MAE 0.050 vs. 0.149, and 0.081 vs. 0.241 as gold becomes scarce; the advantage concentrates in the high-bias regime, while per-gate differences on balanced gates are within noise), because it escapes the $1/J^2$ variance that makes the classical correction fragile. Beyond verifiable gold, we *measure* recovery on public, human-labeled *non-verifiable* gates—response safety and summary quality—where PPI++ recovers the true rate to mean MAE 0.043 versus naive 0.165 ($\sim 4\times$ in aggregate, up to $> 10\times$ on the most-biased gates); and—in the motivating regime, a non-verifiable safety judge *inside* a real retry loop (n=400, against a pre-registered 3-model strong-LLM panel—a disclosed proxy, human-anchored at raw 0.90 agreement)—a lenient gate ships 6.8% truly-unsafe responses (95% CI $[0.05, 0.10]$) while the calibrated correction recovers the panel safe-rate $\sim 3.5\times$ more accurately than naive. The deliverable is a recipe: report a PPI++ estimate alongside J as a reliability/identifiability diagnostic, measure at K=1, and—via a label-free drift detector (ROC-AUC 0.80)—de-bias only when calibration transfers. We release all code and content-free data.

1 Introduction

A growing class of agentic systems puts a language-model judge in the control loop. The agent produces work; an LLM judge scores it; on a FAIL the agent revises or re-runs and is judged again; the loop stops when the judge returns PASS (or a retry budget is exhausted). The fraction of items that end in PASS is then reported as the system's *success rate*—the headline number on dashboards, in model cards, and in procurement decisions.

This number is a measurement, and the conditions under which it is a *valid* measurement have not been examined. The judge is not a fixed instrument: re-running the same judge on the same item at temperature zero flips its verdict at a non-trivial rate (Feuer et al., 2025; Guerdan et al., 2025), and that noise concentrates on the hard, consequential cases. Crucially, the loop does not sample the judge once—it samples *until it gets a PASS*. Retrying never turns a PASS into a FAIL; it can only convert a FAIL into a PASS. So any item the judge would pass on *some* run eventually ships, including a genuine violation the judge catches

only intermittently. This is optional stopping against a noisy classifier: like repeated significance testing on accumulating data (Armitage et al., 1969) or reusing a held-out evaluator until it agrees (Dwork et al., 2015; Berk et al., 2013), the stopping rule biases the reported quantity even though no individual judgment is dishonest.

We treat this as a measurement-science problem and contribute a constructive fix rather than only a diagnosis. The estimator we use is classical—our contribution is not a new statistic but the identification of loop-gated success as a prevalence-under-imperfect-test problem, the result that retrying actively erodes the gate's discrimination (so more retries means a worse measurement), and the resulting operational recipe. Our contributions:

- **The gate is a classifier (§2).** We show a cap-K retry-until-PASS gate is a binary classifier of true compliance with closed-form, K-dependent sensitivity $\mathrm{Se}(K)$ and specificity $\mathrm{Sp}(K)$, so the reported success rate $R(K) = \pi\mathrm{Se}(K) + (1-\pi)(1-\mathrm{Sp}(K))$ is a biased estimator of true success π.
- **Retrying erodes discrimination (§3).** The gate's Youden index $J(K) = \mathrm{Se}(K) + \mathrm{Sp}(K) - 1$ is non-increasing in K (under a mild stochastic-dominance condition, Prop. 1) and converges to a *reliability floor* $J(\infty)$, because retries add false passes on violations faster than they rescue true passes on already-passing compliant work.
- **Recovery, identifiability, and the right estimator (§4).** We recover π by correcting for the gate's class-conditional rates from a small gold-labeled calibration set, and benchmark estimators head-to-head: *PPI++* (Angelopoulos et al., 2023a) dominates the classical Rogan–Gladen correction (Rogan & Gladen, 1978), which pays a $1/J^2$ variance penalty. We thus recommend a PPI++ estimate reported with J as the gate's *informativeness* coefficient—at J=0 the gated rate is uninformative and no estimator beats the gold-only mean—and show the measurement-optimal cap is $K^\star$=1.
- **Real agent loops, objective ground truth, three tasks (§5.1).** On 52 genuine revise-then-rejudge loops across GSM8K, MATH, and code generation (agent×judge×rubric), retrying inflates reported success up to 2.7× over objectively-true delivered success; 45–73% of retry gains on a fallible agent are judge slips, not fixes; the effect is natural (capable agents, strict rubrics) and *largest on code* (where the judge cannot run the candidate); even GPT-4.1 leaks; the correction recovers truth up to 11× better; and a closed form predicts the slip across all 52 configs (r=0.95). (The 2.7×/11× extremes are the terse stress agent; the capable-agent median slip is +0.16 and its worst group is code, true 0.74 to reported 0.98 (Table 3).)
- **Breadth: non-verifiable domains, a stopping-rule taxonomy, a transfer check (§5).** Beyond verifiable gold we *measure* recovery on public, human-labeled *non-verifiable* gates (response safety, summary quality), where PPI++ beats naive $\sim 4\times$ (the regime the method is built for); we show the bias generalizes across judge-aggregation rules (retry, majority, unanimous) with oppositely-signed bias; and we give a *label-free* detector (ROC-AUC 0.80) for when the correction can be trusted—across twelve public gates (17,280 rulings).

The takeaway for practitioners is a recipe, not just a warning: a loop-gated pass rate should be reported with its calibrated bias and reliability J rather than raw; de-biasing helps decisively when that bias is large (and is within noise when it is small, where the diagnostic correctly says not to bother), the calibration needs only a handful of labeled items, and—counter to the intuition that more retries means a better system—more retries make the gate *less* informative, never more.

2 The loop-gated success rate is a biased estimator

Setup. An agent attempts n items drawn i.i.d. Item i has a true label $y_i \in \{\text{compliant}, \text{violation}\}$; let $\pi = \Pr[y_i = \text{compliant}]$ be the *true success rate* we wish to measure (estimated by the sample mean). A judge ruling on item i returns PASS with item-specific probability p_i (the judge's run-to-run instability, estimated from repeated temperature-zero rulings). Under *retry-until-PASS* with cap K, item i ships (ends in PASS) iff the judge passes it on at least one of up to K independent rulings:

$$s_i(K) \;=\; \Pr[\text{ship within } K] \;=\; 1 - (1-p_i)^K. \tag{1}$$

The reported success rate is $R(K) = \frac{1}{n}\sum_i s_i(K)$. Equation (1) models the K rulings as independent draws with a fixed per-item rate p_i; we do not assume this, we check it. Across the public re-judgment gates (§5.2) the lag-1 same-verdict rate across reruns matches the value predicted under independence ($\mathbb{E}[\text{same} \mid p_i] = p_i^2 + (1-p_i)^2$) to within 0.004 on every gate—no detectable serial correlation—and a substantial fraction of items are genuinely "flippy" ($0 < p_i < 1$) rather than deterministic. We estimate each p_i from repeated rulings and propagate that finite-sample noise through a two-level bootstrap (§5), and we cap all empirical caps at the number of rulings so no quantity is extrapolated.

The gate is a classifier. Group items by true label. Define the gate's class-conditional ship rates

$$\mathrm{Se}(K) = \mathop{\mathbb{E}}_{y_i=\text{comp}} s_i(K), \qquad 1-\mathrm{Sp}(K) = \mathop{\mathbb{E}}_{y_i=\text{viol}} s_i(K), \tag{2}$$

i.e. $\mathrm{Se}(K)$ is the probability a truly compliant item is (correctly) shipped and $1-\mathrm{Sp}(K)$ is the probability a true violation is (wrongly) shipped—the *slip* or leak rate. The cap-K gate is then a binary classifier "ship = predict-compliant," and its expected apparent positive rate is the law of total probability:

$$\mathbb{E}[R(K)] \;=\; \pi\,\mathrm{Se}(K) \;+\; (1-\pi)\big(1-\mathrm{Sp}(K)\big). \tag{3}$$

Equation (3) is the crux: $R(K)$ equals π only if the gate is perfect ($\mathrm{Se} = \mathrm{Sp} = 1$). For any imperfect judge it is a biased measurement whose bias, $R(K) - \pi = (1-\pi)(1-\mathrm{Sp}(K)) - \pi(1-\mathrm{Se}(K))$, mixes two error channels—shipped violations and rejected compliant work—in proportions that move with the retry cap.

3 Retrying erodes the gate's discrimination

Both class-conditional ship rates rise with K (Eq. 1 is increasing in K for every p_i). What matters for measurement is their *difference*, the gate's Youden index

$$J(K) \;=\; \mathrm{Se}(K) + \mathrm{Sp}(K) - 1 \;=\; \mathop{\mathbb{E}}_{y_i=\text{comp}} s_i(K) \;-\; \mathop{\mathbb{E}}_{y_i=\text{viol}} s_i(K), \tag{4}$$

the standard scalar discrimination of a binary test (Youden, 1950).

Proposition 1 (Discrimination is non-increasing in the retry cap)**.** *By definition $J(K{+}1) - J(K) = \mathbb{E}_{comp}[\Delta s] - \mathbb{E}_{viol}[\Delta s]$, with the per-attempt gain $\Delta s(p) = s_i(K{+}1) - s_i(K) = p(1-p)^K$. A* sufficient *condition for $J(K{+}1) \le J(K)$ is that both class pass-rate distributions are supported on the decreasing branch $p \ge 1/(K{+}1)$ and the compliant class first-order stochastically dominates (FOSD) the violation class.*

Proof. $\Delta s(p) = p(1-p)^K$ is unimodal with peak at $p{=}1/(K{+}1)$, so on $p \ge 1/(K{+}1)$ it is monotone decreasing. FOSD of compliant over violation then reverses the expectation order for a decreasing function: $\mathbb{E}_{\text{comp}}[\Delta s] \le \mathbb{E}_{\text{viol}}[\Delta s]$, i.e. $J(K{+}1) - J(K) \le 0$. □

This condition is *sufficient*, not universal: it characterizes pass-prone, prevalence-skewed gates rather than every gate. In that regime the gate's Sp collapses with retries while Se barely moves—each retry leaks more violations than it rescues compliant work—and $J(K)$ decreases monotonically toward a *reliability floor*

$$J(\infty) \;=\; \Pr_{\text{comp}}[p_i > 0] \;-\; \Pr_{\text{viol}}[p_i > 0], \tag{5}$$

the gap between the fractions of each class the judge *ever* passes (by bounded convergence; this floor can even be negative—an anti-discriminative gate—if violations are more often ever-passable than compliant work). Retrying drives the gate toward this floor: in the limit the gate ships every item it can ever pass, and all discrimination beyond $J(\infty)$ is lost. This is the loop-level face of optional stopping—the stopping rule consumes the instrument's discriminating power. We do not assert this universally: measuring the exact controlling quantity $J(K{+}1) - J(K) = \mathbb{E}_{\text{comp}}[\Delta s] - \mathbb{E}_{\text{viol}}[\Delta s]$ on all twelve public gates—which are balanced ($\pi{=}0.5$)—it stays near-zero ($|\Delta J| < 0.02$), exactly as the sufficient condition predicts for non-skewed gates, while the synthetic prevalence-skewed gate of §4 exhibits the full collapse ($J : 0.80 \to 0.01$). We therefore present erosion as a *gate-dependent* regularity established by Prop. 1 and the synthetic construction—biting exactly in the pass-prone, prevalence-skewed regime where the loop is most consequential—not a universal law.

Beyond retry: a taxonomy of stopping-rule bias. Retry-until-PASS is one judge-aggregation rule among several, and the bias is a property of the family. Re-aggregating the same six rulings on each public gate, the signed bias is ordered by how pass-prone the rule is: any-pass/retry is the most inflationary, majority-vote intermediate, and unanimous the most conservative—so on a fixed judge and system the reported success rate is a free parameter of the aggregation choice, and two rules can give *oppositely-signed* bias on the same gate (e.g., GSM8K/GPT-4o-mini under a lenient rubric: any-pass +0.017, unanimous −0.033). Crucially, retry is *not* a fixed parallel-OR test: the cap is a *data-dependent stopping time*, so Se is inflated by the stopping rule itself and the classical fixed-Se/Sp screening algebra under-predicts the leak. Recovery is rule-agnostic—each rule induces its own Se, Sp—so the same J diagnostic and PPI++/Rogan–Gladen correction apply throughout.

4 Recovering true success, and when it is possible

The correction. Equation (3) is linear in π and inverts directly. Given the gate's class-conditional rates, the *Rogan–Gladen* estimator (Rogan & Gladen, 1978)—the classical correction of an observed prevalence for an imperfect diagnostic test—recovers true success:

$$\hat{\pi}(K) \;=\; \frac{R(K) - \big(1 - \mathrm{Sp}(K)\big)}{\mathrm{Se}(K) + \mathrm{Sp}(K) - 1} \;=\; \frac{R(K) - \big(1 - \mathrm{Sp}(K)\big)}{J(K)}. \tag{6}$$

In practice $\mathrm{Se}(K)$ and $\mathrm{Sp}(K)$ are unknown but estimable from a small *calibration* set of gold-labeled items run through the same gate; $\hat{\pi}$ is then applied to the unlabeled bulk. This is a measurement-design analogue of recent externally-valid judge estimators (Guerdan et al., 2025): a few labels fix the instrument's class-conditional behavior, and the rest of the data is corrected, not trusted raw.

Which estimator, and the true role of J. Rogan–Gladen is the most *interpretable* corrector—it exposes Se, Sp and hence J—but it is not the most accurate, and saying so sharpens the contribution. Benchmarked out of sample against the prevalence-under-noise alternatives—prediction-powered inference (PPI/PPI++) (Angelopoulos et al., 2023a) and a Bayesian Se/Sp-uncertainty estimator—across all twelve gates, **PPI++ dominates**: mean recovery MAE 0.050 vs. Rogan–Gladen 0.149 and naive 0.149 at a 50% calibration split, and the gap widens as gold becomes scarce (Rogan–Gladen 0.194/0.241 at 20%/10% calibration while PPI++ holds 0.061/0.081). The reason is structural: Rogan–Gladen's variance scales as $1/J^2$ (its empirical spread tracks $1/J$ across gates, r=0.93, and explodes at low J), whereas PPI++'s variance is governed by the judge–gold correlation and tracks $1/J$ far more weakly (r=0.66), so it *escapes* the $1/J^2$ penalty. This resolves an apparent tension between estimator and theory: J is not the variance of any one estimator but the gate's *informativeness* coefficient—at J=0 the gated rate carries no information about π, so every estimator falls back to the gold-only mean and the gate's *efficiency* gain vanishes (gold still identifies π; it is the gate that becomes useless). We therefore recommend reporting **a PPI++ point estimate alongside J as the reliability diagnostic**, with Rogan–Gladen as the bridge that makes J explicit; $K^\star$=1 is the right measurement cap for the whole pipeline, because retrying only lowers J—making the gate less informative for *any* estimator, PPI++ included—never more. We confirm this empirically (App. A): out-of-sample recovery MAE is minimized at K=1 for *both* estimators (PPI++ flat at 0.046, Rogan–Gladen 0.119→0.120 across K=1–6), so retrying never improves recovery for the recommended estimator either.

Identifiability and the cost of retrying. With the calibrated Se, Sp treated as known, the delta method gives $\mathrm{Var}(\hat{\pi}(K)) \approx \mathrm{Var}(R(K))/J(K)^2$, and since $s_i \in [0, 1]$,

$$\mathrm{Var}\big(\hat{\pi}(K)\big) \;\lesssim\; \frac{R(K)\big(1 - R(K)\big)}{n\, J(K)^2}. \tag{7}$$

Estimating Se, Sp from a finite calibration set adds further terms that also scale with $1/J^2$; we therefore do not rely on Eq. (7) for our intervals but report the empirical spread of the out-of-sample estimator (§5), which includes calibration noise. The qualitative content is what matters: the recovered estimate's precision scales with $1/J^2$. For any $J > 0$, π is point-identified by Eq. (6); as $J \to 0$ the inversion becomes

estimator	MAE @ 50% cal	@ 20%	@ 10%
naive (report gated rate)	0.149	0.148	0.147
Rogan–Gladen	0.149	0.194	0.241
PPI	0.049	0.066	0.091
PPI++	**0.050**	**0.061**	**0.081**
Bayesian (Se, Sp posterior)	0.142	0.146	0.139

Table 1: Estimator bake-off: out-of-sample recovery MAE vs. true π across the twelve re-judgment gates, by gold-calibration fraction (300 splits). PPI++ dominates and is stable as gold becomes scarce, while Rogan–Gladen pays the $1/J^2$ penalty (degrades to 0.241). We therefore recommend a PPI++ point estimate with J reported as the identifiability diagnostic.

ill-conditioned and π is non-identifiable exactly at J=0, where two agents of different quality give the same gated rate. A synthetic gate makes this concrete (compliant pass-rate 0.90, violation pass-rate $q \to 0.90$): J falls $0.80 \to 0.01$, the gated rates of two agents differing by 0.20 in true success collapse from a 0.16 gap to 0.002, and $1/J$ explodes to 100×. Real gates sit above this floor (§5.2: J=0.16–0.45), so we see inflated variance, not collapse—but the floor is what a practitioner must monitor.

Because $J(K)$ is non-increasing (Prop. 1), the variance is minimized at the smallest cap: **the variance-optimal measurement policy is $K^\star = 1$.** This yields a two-part recipe. *Prevention:* to *measure* success, run the gate once and report the corrected estimate—retrying only erodes J. *Cure:* when a system is already locked into a high-retry loop (the rate cannot be re-collected at K=1), or when the gate is biased even at K=1, the correction recovers what the raw rate cannot. The distinction from agent *utility* is deliberate: a retry policy may well help the agent succeed; our claim is only about *measuring* that success.

A label-free transfer check. The correction is only as good as the calibration set's transfer to the evaluation set, and this can fail—calibrating on a 0.96-solve agent and recovering a 0.30-solve agent gives MAE 0.31, worse than naive. We give a deploy-time detector that needs *no* evaluation gold: the observed gated-rate shift $|R_{\text{cal}} - R_{\text{eval}}|$. Across 76 cross-agent transfers it predicts whether the correction will beat naive with ROC-AUC 0.80; trusting the correction only when the shift is below 0.05 gives precision 0.94, and one falls back to the raw rate (or re-calibrates) above it. We report this threshold as an *illustrative* operating point computed in-sample over the 76 transfers, not an out-of-sample-validated guarantee; the takeaway is that a cheap, label-free signal carries real information about transfer failure. Calibration transfer is thus a *monitorable operating condition*, not a silent assumption.

5 Experiments

We study the loop in three settings: **real revise-then-rejudge agent loops** with objective ground truth (§5.1, the most direct test), a **cross-judge study** over twelve public judges (§5.2), and a **non-verifiable, human-labeled** domain (§5.3). Throughout, "naive" means reporting the gated rate as the success rate, and the out-of-sample protocol estimates the calibration quantities on a random gold-labeled half and recovers on the held-out half (300 splits).

5.1 Real revise-then-rejudge agent loops

The closed-form analysis above models retries as re-judgments of fixed work. We now test the genuine case—an agent that *revises* between attempts—with *objective* ground truth. On GSM8K, an agent solves a problem; an LLM judge gates PASS/FAIL; on FAIL the agent regenerates with rejection feedback and is re-judged, to a cap K=5; the loop ships the first PASS. True correctness is exact match to the gold integer (no human gold, no κ). We cross three agent-capability levels (a terse no-scratchpad agent and careful GPT-4o-mini and GPT-4o; objective solve rates $\approx$ 0.34/0.92/0.96), four judges (GPT-4o-mini, GPT-4o, GPT-4.1, Gemini-2.5-Flash), and two rubrics (strict; lenient "approve unless clearly wrong")—24 GSM8K loop configurations, n=250, ∼30k agent+judge calls; with the MATH (16) and code (12) sweeps below this

	judge (strict rubric) — slip $R-\pi$ at K=5			
agent (true solve rate)	GPT-4o-mini	GPT-4o	GPT-4.1	Gemini-Flash
terse (≈ 0.34)	+0.15	+0.49	+0.46	+0.38
careful GPT-4o-mini (≈ 0.92)	+0.06	+0.07	+0.07	+0.07
careful GPT-4o (≈ 0.96)	+0.03	+0.04	+0.04	+0.02

Table 2: Real agent loops, slip (reported − objectively-true delivered success) at K=5. Slip scales with agent failure × judge leak: large for a fallible agent, small for a strong one. Judge discrimination J against objective truth spans 0.007 (GPT-4o-mini/lenient—a coin flip) to 0.65; even the strongest judge (GPT-4.1, $J \approx 0.25$–0.37) has nonzero leak, so no judge eliminates the slip.

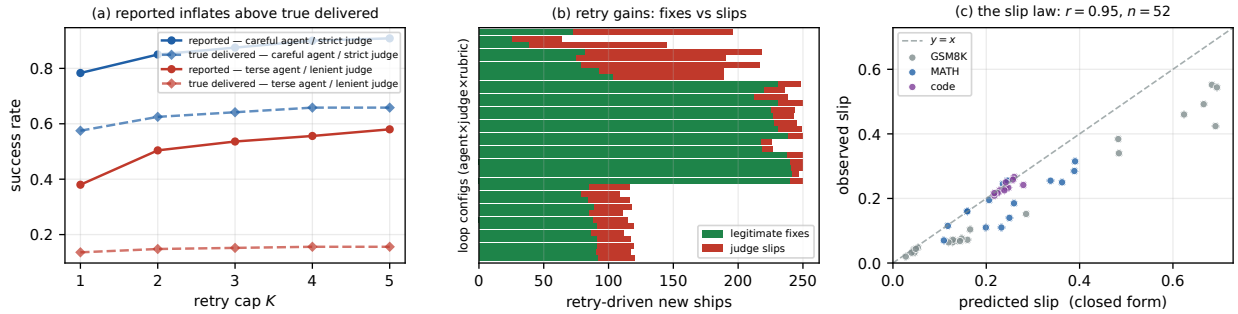

Figure 1: Real agent loops. **(a)** Reported success inflates above objectively-true delivered success as the loop retries (pathological: terse agent / lenient judge; benign: careful agent / strict judge). **(b)** Retry-driven new ships split into legitimate fixes vs. judge slips—on the fallible agent, 45–73% of retry gains are slips. **(c)** The slip obeys a predictive *law*, not a corner: a parameter-free forward model from per-attempt (a, s, ℓ) tracks observed slip across all 52 configs (pooled r=0.95) and within each task (mean r=0.86); regression slope 0.74 (< 1, the gap = revision benefit), confirming it is a real prediction, not the slip=reported×(1−PPV) identity.

totals 52 loop configurations (44 excluding the deliberate terse stress agent). Because we log every attempt's verdict *and* objective correctness, we can replay any cap, decompose retry gains, and recover true success.

Retrying inflates reported success far above truth. A fallible agent (terse; true success ≈ 0.29) gated by a GPT-4o judge reports success climbing from 0.62 to 0.78 as K goes $1 \to 5$, while objectively-true delivered success barely moves ($0.26 \to 0.29$): a slip of **+0.49**. Decomposing the retry-driven new ships, 63% **are judge slips**—the judge waved through a still-wrong answer—not legitimate fixes. The agent is not improving; retrying manufactures false successes by giving an imperfect judge repeated chances to pass a wrong answer. Manual inspection confirms the mechanism: the terse agent emits a bare wrong number (e.g. `Answer: 65000` when the truth is 70000) and the judge, with no reasoning to check, rubber-stamps it—so *agent capability modulates the judge's leak*, not merely its own correctness (the careful agent, which shows work, draws a markedly lower leak). We stress that these loop quantities are computed *directly* from each attempt's logged verdict and objective correctness; they do not rely on the closed-form draw model of §2 (which we invoke only for the fixed-work re-judgment gates). This terse/lenient config is a deliberate *stress case*; the effect is not an artifact of an artificially weak agent: a capable agent (GPT-4o-mini, ≈ 0.92 solve rate) under the same judges still slips a real +0.06 to +0.10, and the strong GPT-4o agent +0.02 to +0.05 (Table 2)—small but nonzero wherever the judge leaks, which is everywhere.

Recovery is decisive here. The loop ships only PASS, so true delivered success is reported × PPV, with PPV = Pr[correct | shipped] estimable from a small gold-labeled calibration set. Out of sample, the corrected estimate's error is *bounded*—MAE 0.02–0.05 across all 24 configs—while the naive report's error tracks the slip, up to 0.55. On the headline config the correction is **11**× more accurate (MAE 0.044 vs. 0.494).

task	rubric	n	true π	reported	mean slip	slip range	MAE corr/naive
GSM8K	strict	8	0.92	0.97	+0.05	$[+0.02, +0.07]$	0.02/0.05
GSM8K	lenient	8	0.92	0.98	+0.06	$[+0.03, +0.10]$	0.02/0.06
MATH	strict	8	0.71	0.86	+0.14	$[+0.07, +0.26]$	0.04/0.14
MATH	lenient	8	0.71	0.95	+0.24	$[+0.16, +0.31]$	0.05/0.24
code	strict	6	0.72	0.95	+0.23	$[+0.21, +0.26]$	0.06/0.23
code	lenient	6	0.74	0.98	+0.24	$[+0.22, +0.27]$	0.06/0.24

Table 3: The slip across capable-agent loops (K=5), grouped by task and rubric—the full distribution behind the headline. Slip is task-ordered (small on GSM8K, large on code where the judge cannot execute) and present even under *strict* judging. In every cell the calibrated correction (corr) beats the naive report, cutting MAE $\sim$4$\times$ overall (mean 0.04 vs 0.16). The terse stress case (Table 2) is excluded here.

Unlike the balanced re-judgment gates of §5.2 (where the per-gate win was within noise), here the slip is large and the correction's advantage is decisive and significant. The benign corner—strong agent, accurate judge—shows small slip and little need for correction, exactly as the reliability diagnostic prescribes: report J and the calibrated bias, and correct when they are large. The bootstrap intervals are well-calibrated: on a synthetic gate with known π, empirical 95%-CI coverage stays in $[0.92, 0.97]$ across $J \in [0.1, 0.8]$, and the reliability floor manifests as interval *width* ($\propto 1/J$: 0.20 at J=0.8 growing to 0.92 at J=0.1), not as broken coverage.

The inflation is a distribution, not a corner: MATH and code. The headline numbers above are deliberate stress cases, so the fair question is what the slip looks like *typically*. To answer it we replicate on competition MATH (integer-answer subset) and code generation (MBPP, with *execution* ground truth—the canonical agentic setting) using only *capable* agents (natural solve 0.5–0.96) and no terse trick, and report the full per-task distribution (Table 3). Across all 44 capable-agent configurations the median slip is +0.16 and the mean +0.15 (range +0.02 to +0.32)—so the effect is not a single adversarial corner but a systematic, task-ordered bias: small on easy GSM8K (+0.05), substantial on MATH even under the *strict* rubric (+0.14), and largest on *code* (+0.23, J as low as 0.07), where a judge that cannot run the candidate rubber-stamps plausible-looking programs. Critically, the correction is decisive across the *whole* distribution, not just the corners: it recovers true delivered success to MAE 0.04 versus naive 0.16 on average—a $\sim$4$\times$ reduction that holds in every task–rubric cell of Table 3.

A closed-form law predicts the slip. What makes the bias a measurement quantity rather than a list of anecdotes is that it obeys a *predictive law*—and we are careful to separate what is definitional from what is empirical. The accounting identity slip = reported $\times(1 - \text{PPV})$ holds by construction for any ship-on-PASS loop and is *not* the claim. The empirical claim is that a parameter-free *forward* model predicts the slip from the loop's per-attempt statistics: modeling each attempt as independently correct with rate a, judge sensitivity s, and leak ℓ gives $\text{reported}(K) = 1 - (1 - \rho)^K$ and $\text{slip}(K) = \text{reported}\times(1 - a)\ell/\rho$ (ρ=as+$(1-a)\ell$). Each config's (a, s, ℓ) are estimated only from *that config's own* attempt logs—no parameters are shared or fit across configs—so the comparison is out-of-sample across the 52 loops by construction. The prediction tracks the slip measured against *objective* truth not only across all 52 configs (pooled r=0.95, bootstrap 95% CI $[0.92, 0.97]$) but *within* each task (mean within-task r=0.86: GSM8K 0.99, MATH 0.82, code 0.78; Fig. 1c), so the fit is not an artifact of between-task spread, and the predictive slip-MAE is 0.05. Decisively, the predicted-vs-observed regression slope is 0.74 with bootstrap 95% CI $[0.67, 0.80]$—excluding 1 ($p < 0.001$), which a definitional identity would force—so the law is a genuine forward prediction, not the slip=reported $\times$ $(1 - \text{PPV})$ identity restated. The i.i.d.-attempt model systematically *over*-predicts slip by this margin, and that residual is precisely the benefit of the agent's *revision*. The law thus turns the path-dependence of revised attempts into a measured quantity rather than an unmodeled assumption, and confirms retries mostly manufacture slips rather than fixes. A regression-dilution check rules out the innocuous alternative that the < 1 slope is mere predictor noise: the predictor reliability is 0.977 and the errors-in-variables (Deming) slope is $0.765\,[0.70, 0.84]$, still excluding 1, and the slope stays below 1 *within* task (GSM8K 0.76, MATH 0.79; code ≈ 1, i.e. revision cannot help where the judge cannot execute). Per

gate (judge, lenient grading)	Se(1)	leak 1−Sp	true π	bias $R-\pi$
GSM8K / Gemini-Flash	0.969	0.733	0.50	+0.351
TruthfulQA / GPT-4o-mini	0.821	0.596	0.50	+0.208
HaluEval / Gemini-Flash	0.844	0.431	0.50	+0.137
TruthfulQA / Gemini-Flash	0.824	0.404	0.50	+0.114
HaluEval / GPT-4o-mini	0.690	0.394	0.50	+0.042

Table 4: The pass-prone inflation regime on public data. Lenient judges leak violations (leak $\approx$ 0.4–0.7) and inflate reported success above true success—by up to +0.35. Across all twelve gates the calibrated bias spans −0.35 to +0.35 (both directions present publicly).

task×rubric cell, the recovery gain (naive−corrected MAE) is positive in all six cells with bootstrap CIs excluding zero (Holm-corrected); the within-noise outcome is confined to the balanced re-judgment gates, where a power analysis shows the evaluation (n=240) can only resolve gaps above ≈ 0.009.

The motivating regime, measured: a powered, pre-registered non-verifiable loop. The setting the method is built for is a non-verifiable gate *inside* a real loop—a safety/compliance judge that cannot mechanically check the work, where a slip is a shipped *unsafe* response. We measure it directly and at scale on BeaverTails (n=400 prompts): an agent revises under a weak gpt-4o-mini safety gate to cap K=5, with ground truth a *pre-registered* 3-judge strong panel (GPT-4.1, Claude, Gemini; majority-safe, the rule fixed in advance). The phenomenon is clear and statistically decisive. A *lenient* gate ships every response (J=0) yet 6.8% are truly unsafe (slip +0.068, Wilson 95% CI $[0.047, 0.096]$, excluding zero); a *strict* gate (J=0.35) shows the slip *grow with retries*—+0.037 at K=1 to +0.042 at K=5 (reported 0.91→0.98, true-safe 0.88→0.94; 4% of retry-driven ships are judge slips), the optional-stopping inflation in its motivating form. In both, the calibrated PPV correction is *decisively* bounded (MAE 0.017–0.019) while naive tracks the slip (0.04–0.07): a 2.4–3.5× recovery, now in the non-verifiable, looped regime rather than only on verifiable tasks. We anchor the panel gold to humans: on a 120-response subset a human labeler agrees with the panel at raw 0.90 (Cohen's κ=0.45, depressed by the $\sim$7% unsafe base rate; PABAK 0.80), with a human-gold slip (+0.042) of the same sign and order. The weak in-loop judge agrees with humans far less (κ=0.15)—it rubber-stamps the safe majority while missing the unsafe minority, exactly the leak we study. Honest scope: ground truth here is a strong-model panel human-validated on a subset, not human adjudication at full scale—a remaining annotation cost, not a missing result.

5.2 A cross-judge measurement study

The framework also applies to gates that re-judge fixed work, where the closed-form $\mathrm{Se}(K), \mathrm{Sp}(K)$ hold directly. To test the account across many judges on fully auditable data, we replicate the full pipeline across **twelve public gates** (dataset × judge × grading-style) over three datasets—TruthfulQA (Lin et al., 2022), HaluEval-QA (Li et al., 2023), GSM8K (Cobbe et al., 2021)—two judges (GPT-4o-mini, Gemini-2.5-Flash), and two grading styles (a strict correctness rubric and a *lenient* "approve unless clearly wrong" rubric), each ruling six times at temperature zero (17,280 rulings; same correct-vs-incorrect-answer construction). Independence held on every gate (lag-1 same-verdict within 0.004 of the i.i.d. prediction); $J(1)$ ranges 0.16 (GPT-4o-mini grading math) to 0.45.

Both inflation and deflation appear publicly. Under a strict rubric public judges are fail-prone (retries move *toward* truth); under a lenient rubric they are pass-prone and inflate reported success above truth (Table 4). This places the phenomenon on fully reproducible data, in both bias directions.

Naive error is bias; correction trades it for variance. Two facts across the twelve gates. First, near-definitionally, the naive report's error equals the gate's calibrated bias (corr = 0.998; both are $|R(1) - \pi|$ on gold)—we state it not as a discovery but to register that this bias is *large and real*, up to 0.35, and present on public gates. Second, and genuinely empirical: the de-biased estimate's error is decoupled from bias (corr = 0.17) and instead tracks inverse reliability, $\mathrm{corr}(\mathrm{MAE}, 1/J) = 0.93$ (bootstrap 95% CI $[0.88, 1.00]$;

domain	judge / rubric	π	naive bias	naive	RG	PPI++
safety	gpt4o-mini / strict	0.50	-0.04	0.038	0.052	0.041
safety	gpt4o-mini / lenient	0.50	$+0.10$	0.097	0.054	**0.038**
safety	gemini / strict	0.50	-0.21	0.210	0.102	**0.044**
safety	gemini / lenient	0.50	$+0.09$	0.095	0.057	**0.039**
quality	gpt4o-mini / strict	0.62	-0.54	0.538	0.204	**0.050**
quality	gpt4o-mini / lenient	0.62	-0.22	0.226	0.090	**0.049**
quality	gemini / strict	0.62	-0.10	0.096	0.060	**0.045**
quality	gemini / lenient	0.62	$+0.01$	0.020	0.055	0.040

Table 5: Recovery on *non-verifiable*, human-labeled gates (BeaverTails safety; SummEval quality), out of sample vs. human gold. Bias runs both ways (strict judges fail-prone, lenient pass-prone); PPI++ recovers the true rate to MAE ≈ 0.04 throughout—$\sim 4\times$ better than naive on average and $>10\times$ on the most biased gate. The two gates where naive ties are exactly those with $|\text{bias}| \leq 0.04$, where the bias/J diagnostic says not to de-bias.

0.99 with the GSM8K gates removed, so not an outlier artifact). So the correction does exactly one thing: it removes the bias and pays $1/J^2$ variance. Whether that is a net win on a *single* gate depends on whether |bias| exceeds the recovery noise, and at our evaluation size (n=240) the per-gate paired MAE differences are within noise (bootstrap CIs include zero). We therefore do not claim a per-gate win; we claim the operative quantities. The bias is large, real, and exactly measured (up to 0.35), and J governs the cost of removing it. *Report both:* trust the raw rate when the calibrated bias is small; de-bias when it is large and J is not too low; and when J is very low (GSM8K/GPT-4o-mini, J=0.16), the irreducible recovery error is itself the verdict that the judge is too unreliable to gate on. The reliability coefficient and the calibrated bias, not a single point estimate, are the deliverable. (Recovery also holds at skewed prevalence: subsampling public gates to π=0.2 and 0.8 still recovers the true rate, e.g. $\hat{\pi}$=0.80 at π=0.8.)

5.3 Recovery in a non-verifiable domain (safety and summary quality)

All evidence above uses *verifiable* gold (exact match, execution), yet the motivating regime—a compliance gate—is precisely one where the judge *cannot* mechanically check the work. We therefore test recovery directly on two public, *human*-labeled, non-verifiable gates: response **safety** (BeaverTails (Ji et al., 2023), n=300, human `is_safe` gold) and **summary quality** (SummEval (Fabbri et al., 2021), $n\approx$240, expert consistency scores, binarized), each judged six times at temperature zero under strict and lenient rubrics by two judges. These are re-judgment gates over fixed responses; combining non-verifiable judging *with* a revise-then-rejudge loop is measured in §5.1, anchored on human gold (n=120 shipped responses) with a strong-model panel used for scale (Table 6). The bias here runs in *both* directions: strict safety/quality judges are *fail*-prone (a strict summary judge under-passes good summaries, naive bias up to -0.54), lenient judges pass-prone—and naive reporting inherits whichever it is. Out of sample (50% human-gold calibration, 300 splits), **PPI++ recovers the true rate to mean MAE** 0.043 **versus naive** 0.165 across the eight gates ($\sim 4\times$ better; $>10\times$ on the most biased, where naive MAE 0.54 falls to 0.05), beating naive on 6/8 gates. The two exceptions are exactly the gates where the bias is already small ($|R-\pi| \leq 0.04$)—the regime where the bias/J diagnostic correctly says *not* to de-bias. This moves the central claim from a structural argument to a *measured* result in the non-verifiable, human-gold regime the method is built for. As a third-provider robustness check, an Anthropic Claude safety judge (a third model family beyond the OpenAI and Google judges) is *well-calibrated*: $J\approx 0.58$ and $|\text{bias}| \leq 0.04$ on BeaverTails (strict/lenient), so naive ties the correction (PPI++ MAE 0.042). This is exactly the low-bias regime where the J/bias diagnostic says *not* to de-bias—and across all three judge families PPI++ is never worse than naive, confirming the selectivity of the recipe rather than a blanket correction.

6 Related work

Measurement science and judge reliability. A growing line treats benchmark scores as measurements with explicit validity and reliability (Jacobs & Wallach, 2021; Salaudeen et al., 2025; Polo et al., 2024), and documents that LLM judges flip across reruns and design choices (Feuer et al., 2025; Zheng et al., 2023); Guerdan et al. (2025) correct judge estimates for rating indeterminacy in the *isolated* judging setting; concurrent single-shot debiasing of imperfect judges includes Lee et al. (2025) (sensitivity/specificity bias with calibration-aware intervals), Chen et al. (2026) (a semiparametric-efficient estimator), and Collot et al. (2025) (Youden's J to *select* a judge)—each scoring a judge that rules *once* per item. We move one level out, to the judge *embedded in a stopping rule*, where the bias is generated by the loop and J becomes a cap-dependent $J(K)$ that retrying erodes—supplying the reliability coefficient and the correction it implies.

Optional stopping and verifier-guided loops. Retry-until-PASS is optional stopping against a noisy classifier, sharing its pathology with repeated significance testing (Armitage et al., 1969), adaptive holdout reuse (Dwork et al., 2015), post-selection inference (Berk et al., 2013; Gelman & Loken, 2013), and—in ML evaluation—adaptive overfitting to a reused test signal. Verifier-guided methods (Reflexion, self-refine, best-of-N) gate attempts by a learned/LLM verifier (Shinn et al., 2023; Madaan et al., 2023); closest, Dorner et al. (2025) characterize how verifier imperfection bounds the *achievable accuracy* of resampling. We ask the inverse, *measurement* question: their object is a policy's attainable performance (an optimization target), ours the estimator's bias and identifiability (a measurement defect). They give no correction recovering true prevalence from a gated rate, nor identify J as the quantity that governs when recovery is possible. To our knowledge no prior work isolates the *stopping rule* (rather than any single ruling) as the generator of the measurement bias, nor identifies the *cap-dependent* $J(K)$ that retrying erodes as the coefficient governing recoverability in the loop.

Quantification, label shift, and prediction-powered inference. Recovering true success π from a noisy gated rate is formally the *quantification* problem—estimating class prevalence in aggregate rather than per item (Forman, 2008; González et al., 2017)—and our class-conditional correction is exactly Adjusted Classify & Count / Rogan–Gladen, with $\mathrm{Var} \sim 1/J^2$ making Youden's J the visible reliability of the count. The calibration-transfer condition the correction rests on (a stable confusion structure from calibration to deployment) is the *label-shift* assumption (Saerens et al., 2002; Lipton et al., 2018; Alexandari et al., 2020), and our label-free $|R_{\mathrm{cal}} - R_{\mathrm{eval}}|$ diagnostic is a quantification shift test in the spirit of BBSE. For the point estimate we adopt PPI++ (Angelopoulos et al., 2023a;b), which dominates Rogan–Gladen (§4). Recent work debiases a *single* noisy judge (Lee et al., 2025; Chen et al., 2026), uses Youden's J to *select* a judge (Collot et al., 2025), or proposes J and ΔJ as reliability diagnostics for single-shot corrected judge estimates (Fiedler, 2026)—all in the single-ruling regime; we instead study the *cap-dependent* $J(K)$ that a stopping rule erodes. Closest to our taxonomy, Ullah & Serwadda (2026) derive the operating characteristics of the *unanimous* jury—one cell of our rule taxonomy, which we generalize and correct; for deployment monitoring, Zhang et al. (2026a) give anytime-valid risk bounds under shift, but consume labels, whereas our transfer gate is label-free. Concurrently, Zhang et al. (2026b) report a Rogan–Gladen undercount in a single deployed LLM-judge gate (degenerating at zero apparent defects, the $J{=}0$ boundary of Eq. (6)); we instead isolate the stopping rule as the bias generator and give the cap-K, $J(K)$-diagnosed recovery recipe.

7 Limitations

We state each central claim with the regime where it is *measured* versus argued (Table 6).

The recovery rests on an identifying assumption: the gate's class-conditional rates (Se, Sp, or PPV in a loop) estimated on the calibration set must *transfer* to the evaluation set. We tested this directly—calibrating on one agent and recovering a very different one (a $0.96 \to 0.30$ solve-rate shift) degrades loop recovery to MAE 0.31, worse than naive—so a practitioner must calibrate on data representative of the system measured; within-distribution it is accurate. Our genuine revise-then-rejudge evidence with objective ground truth covers GSM8K, MATH, and code—all *verifiable* domains; recovery in the non-verifiable regime is validated on human-labeled re-judgment gates (§5.3) and the powered in-loop safety study (§5.1), but we do not

claim	strongest evidence	scope / caveat
bias exists, is K-dependent	52 loops + 12 public gates	holds throughout
recovery: PPI++ $\gg$ naive	verifiable *and* non-verifiable human-gold gates	needs calibration transfer (gated, §4)
J-erosion with retries	Prop. 1 + synthetic gate	prevalence-skewed gates; ≈flat on balanced public gates
slip law / revision benefit	52 loops; Deming slope $0.765\,[0.70, 0.84]$	verifiable-domain attempt logs
non-identifiability ($J\to 0$)	synthetic + low-J real gates ($J\geq 0.12$)	full J=0 only synthetic

Table 6: Claim × evidence scope: each central claim, its strongest support, and the regime where it is established rather than argued.

validate it on open-ended multi-step trajectories, where the extension rests on the structural argument rather than a measured result. On the balanced re-judgment gates the per-gate net MAE win is within bootstrap noise at n=240 (decisive only on the high-slip real loops), so we report the bias and J as the deliverables. The reliability floor $J(\infty)$ and the full $J\to 0$ regime are established by Prop. 1 and a synthetic prevalence-skewed gate rather than measured on a real prevalence-skewed gate (our public gates are balanced, $J \geq 0.16$). Finally, the public re-judgment gates use a controlled lenient rubric and single-shot judging; we model such retries as re-judgments of fixed work, whereas a retry that genuinely *repairs* a violation is a legitimate pass, not a slip—our target is the measurement of success, not the agent's policy.

8 Conclusion

A loop-gated pass rate is a measurement taken with an unstable instrument and a stopping rule that consumes the instrument's discrimination. Read raw, it is biased and—in the limit of many retries—non-identifiable. The conceptual shift is to read it not as a number but as the output of an instrument with a known reliability: its bias is then a calibrated, removable quantity, and J says in advance whether removing it is worth the variance. Two numbers—the calibrated bias and the reliability J—replace one untrustworthy point estimate. The guidance is crisp: measure at K=1, report J and the calibrated bias, de-bias when that bias exceeds the recovery standard error ($\propto 1/J$), and treat a very low J as itself the verdict that the gate is too unreliable to trust. We release the estimator and content-free data to make this the default.

Broader Impact Statement

This work is a measurement-and-auditing method: it exposes that LLM-judge-gated success rates—including safety gates that decide which responses ship—are upward-biased and, past a reliability floor, non-identifiable, and it supplies a calibrated correction. The intended effect is to reduce harm, by making it harder for genuinely unsafe or non-compliant outputs to be reported as "passed." We see two risks to flag. First, a calibrated estimate can be misread as a safety *guarantee*; we stress that the correction is only valid when calibration transfers (we provide a label-free check for this) and that a low J is itself a signal not to trust the gate rather than a license to ship. Second, our non-verifiable evaluation uses potentially unsafe prompts (BeaverTails) and a strong-model panel as a scalable gold proxy human-anchored on a subset; the gold is not human adjudication at full scale, so reported absolute harm rates should be treated as panel-relative estimates, not certified ground truth. No new harmful capability or dataset is released; we release only the estimator and content-free (label/score-only) data.

References

Amr Alexandari, Anshul Kundaje, and Avanti Shrikumar. Maximum likelihood with bias-corrected calibration is hard-to-beat at label shift adaptation. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020.

Anastasios N. Angelopoulos, Stephen Bates, Clara Fannjiang, Michael I. Jordan, and Tijana Zrnic. Prediction-powered inference. *Science*, 382(6671):669–674, 2023a.

Anastasios N. Angelopoulos, John C. Duchi, and Tijana Zrnic. PPI++: Efficient prediction-powered inference. *arXiv preprint arXiv:2311.01453*, 2023b.

Peter Armitage, CK McPherson, and BC Rowe. Repeated significance tests on accumulating data. *Journal of the Royal Statistical Society: Series A*, 132(2):235–244, 1969.

Richard Berk, Lawrence Brown, Andreas Buja, Kai Zhang, and Linda Zhao. Valid post-selection inference. *The Annals of Statistics*, 41(2):802–837, 2013.

Yiqun T. Chen, Sizhu Lu, Sijia Li, Moran Guo, and Shengyi Li. Efficient inference for noisy LLM-as-a-judge evaluation. *arXiv preprint arXiv:2601.05420*, 2026.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Stephane Collot, Colin Fraser, Justin Zhao, William F. Shen, Timon Willi, and Ilias Leontiadis. Balanced accuracy: The right metric for evaluating LLM judges—explained through youden's J statistic. *arXiv preprint arXiv:2512.08121*, 2025.

Florian E Dorner, Yatong Chen, André F Cruz, and Fanny Yang. ROC-n-reroll: How verifier imperfection affects test-time scaling. *arXiv preprint arXiv:2507.12399*, 2025.

Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Roth. The reusable holdout: Preserving validity in adaptive data analysis. *Science*, 349(6248):636–638, 2015.

Alexander R. Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. SummEval: Re-evaluating summarization evaluation. *Transactions of the Association for Computational Linguistics (TACL)*, 9:391–409, 2021.

Benjamin Feuer, Chiung-Yi Tseng, Astitwa Sarthak Lathe, Oussama Elachqar, and John P. Dickerson. When judgment becomes noise: How design failures in llm judge benchmarks silently undermine validity. *arXiv preprint arXiv:2509.20293*, 2025.

James Fiedler. Bias and uncertainty in llm-as-a-judge estimation. *arXiv preprint arXiv:2605.06939*, 2026.

George Forman. Quantifying counts and costs via classification. *Data Mining and Knowledge Discovery*, 17 (2):164–206, 2008.

Andrew Gelman and Eric Loken. The garden of forking paths: Why multiple comparisons can be a problem, even when there is no "fishing expedition". *Department of Statistics, Columbia University (unpublished manuscript)*, 2013.

Pablo González, Alberto Castaño, Nitesh V. Chawla, and Juan José del Coz. A review on quantification learning. *ACM Computing Surveys*, 50(5):74:1–74:40, 2017.

Luke Guerdan et al. Validating llm-as-a-judge systems under rating indeterminacy. *arXiv preprint arXiv:2503.05965*, 2025.

Abigail Z Jacobs and Hanna Wallach. Measurement and fairness. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency (FAccT)*, pp. 375–385, 2021. arXiv:1912.05511.

Jiaming Ji, Mickel Liu, Juntao Dai, Xuehai Pan, Chi Zhang, Ce Bian, Boyuan Chen, Ruiyang Sun, Yizhou Wang, and Yaodong Yang. BeaverTails: Towards improved safety alignment of LLM via a human-preference dataset. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. arXiv:2307.04657.

Chungpa Lee, Thomas Zeng, Jongwon Jeong, Jy-yong Sohn, and Kangwook Lee. How to correctly report LLM-as-a-judge evaluations. *arXiv preprint arXiv:2511.21140*, 2025.

Junyi Li, Xiaoxue Cheng, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. HaluEval: A large-scale hallucination evaluation benchmark for large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2023. arXiv:2305.11747.

Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2022. arXiv:2109.07958.

Zachary C. Lipton, Yu-Xiang Wang, and Alexander J. Smola. Detecting and correcting for label shift with black box predictors. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, 2018. arXiv:1802.03916.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. arXiv:2303.17651.

Felipe Maia Polo, Lucas Weber, Leshem Choshen, Yuekai Sun, Gongjun Xu, and Mikhail Yurochkin. tinybenchmarks: evaluating llms with fewer examples. *Proceedings of the 41st International Conference on Machine Learning (ICML)*, 2024. arXiv:2402.14992.

J. Reiczigel, J. Földi, and L. Ózsvári. Exact confidence limits for prevalence of a disease with an imperfect diagnostic test. *Epidemiology and Infection*, 138(11):1674–1678, 2010.

Walter J Rogan and Beth Gladen. Estimating prevalence from the results of a screening test. *American Journal of Epidemiology*, 107(1):71–76, 1978.

Marco Saerens, Patrice Latinne, and Christine Decaestecker. Adjusting the outputs of a classifier to new a priori probabilities: A simple procedure. *Neural Computation*, 14(1):21–41, 2002.

Olawale Salaudeen et al. Measurement to meaning: A validity-centered framework for ai evaluation. *arXiv preprint arXiv:2505.10573*, 2025.

Noah Shinn, Federico Cassano, Edward Berman, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. arXiv:2303.11366.

Muhammad Aziz Ullah and Abdul Serwadda. Vibe coding on trial: Operating characteristics of unanimous LLM juries. *arXiv preprint arXiv:2602.18492*, 2026.

William J Youden. Index for rating diagnostic tests. *Cancer*, 3(1):32–35, 1950.

Guangyi Zhang, Yunlong Cai, Guanding Yu, and Osvaldo Simeone. Prediction-powered risk monitoring of deployed models for detecting harmful distribution shifts. *arXiv preprint arXiv:2602.02229*, 2026a.

Sawyer Zhang, Alexander Wang, and Sophie Lei. Catching one in five: Llm-as-judge blind spots in production multi-turn transaction agents. *arXiv preprint arXiv:2606.10315*, 2026b.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P Xing, et al. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. arXiv:2306.05685.

A Finite-sample variance and the optimality of $K^\star = 1$

The main-text remark after Eq. (7) propagates only the bulk-rate sampling error and treats the calibrated $\mathrm{Se}(K), \mathrm{Sp}(K)$ as known. Here we discharge that simplification: we give the full delta-method variance of the Rogan–Gladen estimator

$$\hat{\pi}(K) = \frac{R(K) - \big(1 - \mathrm{Sp}(K)\big)}{\mathrm{Se}(K) + \mathrm{Sp}(K) - 1} = \frac{R(K) - \big(1 - \mathrm{Sp}(K)\big)}{J(K)}$$

when $\mathrm{Se}(K)$ and $\mathrm{Sp}(K)$ are themselves *estimated* from a finite gold-labeled calibration set, and we show every variance term is $\Theta\big(1/J(K)^2\big)$. This is the finite-sample correction underlying the exact prevalence interval of Reiczigel et al. (2010). We then show the variance scales as $\Theta(1/J(K)^2)$ in the Prop. 1 regime, whose factor is minimized at the smallest cap, so the variance-optimal measurement cap is $K^\star = 1$ (with exact monotonicity under an additional condition on the numerator, and empirical confirmation for PPI++).

Calibration design. Fix a cap K. The unlabeled bulk supplies $R \equiv R(K) = \frac{1}{n}\sum_i s_i(K)$ over n items. A disjoint calibration set of m gold-labeled items (m_1 truly-compliant, m_0 truly-violating) run through the *same* cap-K gate yields

$$\widehat{\mathrm{Se}} = \frac{1}{m_1}\sum_{i:\, y_i=\mathrm{comp}} s_i(K), \qquad \widehat{\mathrm{Sp}} = 1 - \frac{1}{m_0}\sum_{i:\, y_i=\mathrm{viol}} s_i(K).$$

The three estimators are computed on disjoint item sets and are mutually independent, with binomial-type variances $\mathrm{Var}(\widehat{R}) = R(1-R)/n$, $\mathrm{Var}(\widehat{\mathrm{Se}}) = \mathrm{Se}(1-\mathrm{Se})/m_1$, $\mathrm{Var}(\widehat{\mathrm{Sp}}) = \mathrm{Sp}(1-\mathrm{Sp})/m_0$.

Lemma 1 (Finite-sample delta-method variance)**.** *Write $J = J(K) = \mathrm{Se} + \mathrm{Sp} - 1$ and assume $J > 0$. With $\widehat{R}, \widehat{\mathrm{Se}}, \widehat{\mathrm{Sp}}$ independent as above, the first-order delta method gives*

$$\mathrm{Var}\big(\hat{\pi}(K)\big) \;\approx\; \frac{1}{J^2}\left[\frac{R(1-R)}{n} + \pi^2\,\frac{\mathrm{Se}(1-\mathrm{Se})}{m_1} + (1-\pi)^2\,\frac{\mathrm{Sp}(1-\mathrm{Sp})}{m_0}\right]. \tag{B.1}$$

Every term carries the same $1/J^2$ factor; the two calibration terms are the finite-m corrections omitted from Eq. (7).

Proof. Treat $\hat{\pi} = g(\widehat{R}, \widehat{\mathrm{Se}}, \widehat{\mathrm{Sp}})$ with $g = (R-1+\mathrm{Sp})/(\mathrm{Se}+\mathrm{Sp}-1)$. With $J = \mathrm{Se}+\mathrm{Sp}-1$ and $R-(1-\mathrm{Sp}) = \pi J$ at the truth, $\partial g/\partial R = 1/J$, $\partial g/\partial \mathrm{Se} = -\pi/J$, and $\partial g/\partial \mathrm{Sp} = (1-\pi)/J$. By independence the delta method sums the squared derivatives times the binomial variances; factoring the common $1/J^2$ yields (B.1). Cross-terms vanish because the three estimators use disjoint items. □

Remark 1 (Consistency with the main text)*.* Dropping the calibration terms ($m_1, m_0 \to \infty$) recovers Eq. (7). Equation (B.1) is exactly the variance Reiczigel et al. (2010) replace with an exact construction when J is small or counts are sparse, because the Gaussian interval can under-cover near the $J \to 0$ floor; the $1/J^2$ point-variance scaling is identical.

Proposition 2 (Variance scaling is minimized at $K^\star = 1$)**.** *Suppose the Prop. 1 regime holds (so $J(K)$ is non-increasing in K) and the per-cap calibration budget n, m_0, m_1 is fixed. Then $\mathrm{Var}(\hat{\pi}(K)) = \Theta\big(1/J(K)^2\big)$, whose $1/J(K)^2$ factor is minimized at K=1, so the leading-order variance is minimized at $K^\star = 1$. If in addition the numerator $V(K)$ is non-increasing in K, then $\mathrm{Var}(\hat{\pi}(K))$ is exactly non-decreasing in K.*

Proof. Write (B.1) as $V(K)/J(K)^2$. Each summand of $V(K)$ is the variance of a bounded $[0,1]$ quantity, so $V(K) \le \frac{1}{4}(1/n + \pi^2/m_1 + (1-\pi)^2/m_0) < \infty$ uniformly in K, and (for a non-degenerate gate) $V(K)$ is bounded below by a positive constant; hence $\mathrm{Var}(\hat{\pi}(K)) = \Theta(1/J(K)^2)$. By Prop. 1, $J(K{+}1) \le J(K)$, so $1/J(K)^2$ is non-decreasing in K and is minimized at K=1, and the leading-order variance is therefore minimized at $K^\star$=1. *Exact* monotonicity of $\mathrm{Var}(\hat{\pi}(K))$ additionally requires $V(K)$ non-increasing, which we do not prove but observe to hold empirically; and for the recommended PPI++ estimator $K^\star$=1 is confirmed empirically (Remark 2), not via this $1/J^2$ bound. As $J(K) \to 0$ the variance diverges, recovering the identifiability boundary. □

Remark 2 (Empirical confirmation for PPI++)*.* Prop. 2 is proved for Rogan–Gladen. Measuring out-of-sample recovery MAE at each cap across all gates, PPI++ is *likewise* minimized at K=1 (flat at 0.046 vs. Rogan–Gladen 0.120 → 0.122 over K=1–6): as K grows the gate ships more, $f_i = 1 - (1-p_i)^K$ saturates toward 1, and the judge–gold correlation PPI++ exploits collapses, so its efficiency edge erodes. Hence $K^\star$=1 is the right measurement cap for the recommended estimator too, even though PPI++'s variance is not the $1/J^2$ form of (B.1).

Remark 3 (Scope)*.* $K^\star = 1$ concerns *measurement* variance, not agent utility: extra retries may raise true success but, because every variance channel in (B.1) is inflated by the same $1/J(K)^2$ and $J(K)$ only falls, they strictly worsen the recovered estimate. This is the variance-side companion to the bias-side argument of §3–§4, and is why we report a PPI++ point estimate with J as the reliability diagnostic, collected at K=1 whenever the rate can be re-collected.