CERTIFIED REUSE OR RECALIBRATION OF LLM JUDGES AFTER OUTPUT SHIFT

Anonymous authors
Paper under double-blind review

ABSTRACT

An LLM judge is often calibrated on one model release and then reused after the evaluated outputs change. Its reliability map may continue to transfer, but it can also become stale when visible response properties change their relationship with quality. Recalibration is not automatically safer: generator replacement need not make a repair useful, and a small audit can favor a noisy repair. We study this as a finite-sample reuse-or-recalibration decision. Controlled style conflict provides a positive case, prespecified generator transitions test whether reuse remains adequate, and expert-scored long-form updates provide an external case. On a specified target window, a disjoint audit deploys a repair only when it certifies lower risk; reuse otherwise means only that no registered repair is resolved as beneficial at the current radius. Observable source–repair disagreement determines the paired-loss range, yielding safe near-oracle selection up to certification radii and an exact continuous minimax allocation. A fixed-repair binary-Brier lower bound matches the resulting worst-case label-complexity rate. Simulations verify finite-sample control; grouped replays show that certification suppresses held-out harm from validation-selected repairs while retaining power under drift.

1 INTRODUCTION

LLM judges are widely used for benchmarking, model selection, alignment, and production evaluation (Zheng et al., 2023; Chiang et al., 2024; Lambert et al., 2025; Tan et al., 2025). A common workflow calibrates a fixed judge on one model release and carries the resulting map forward when the evaluated model changes. That map converts verdicts or scores into estimated quality probabilities. The maintenance question is whether it should still be trusted after the outputs change.

This is not automatic. Responses can change in length, structure, reasoning, refusal, and quality even when the benchmark and judge prompt stay fixed (Chen et al., 2023; Dubois et al., 2024; Bharadwaj et al., 2025). Optimization can also alter cue–quality associations (Lamparth et al., 2026). Relabeling every release is costly, but reusing a stale map is risky, and a repair chosen from a small audit can overfit.

The relevant object is a conditional relationship, not the generator identity itself. Before target labels are opened, the evaluator registers a finite calibration contract C built from the verdict and response attributes. For a registered state c and binary label Y, source-environment population reliability is $\eta_s(c) = P_s(Y = 1 \mid C = c)$; the fitted source map estimates η_s.

We call a change from $P_s(Y \mid C)$ to $P_t(Y \mid C)$ that accompanies a changed presentation–quality relationship a *style–quality association shift*. Formatting, for example, may be informative in the calibration pool and become misleading after a model update. The judge continues to emit verdicts, but the reliability attached to them can become stale in target-important states.

We study three settings for different reasons. RM-Bench (Liu et al., 2025b) creates a controlled conflict between style and quality. JETTS (Zhou et al., 2025) asks whether changing the generator alone, with prompts and objective verification fixed, is enough to require repair. RealDeepResearch (RealDR) (Chen et al., 2026a) adds a less controlled long-form setting. The common question is whether target evidence justifies replacing the old map.

Unlabeled traffic can reveal state frequencies and missing support, but not reliability within a state. Target labels are therefore needed both to fit a repair and to decide whether that repair is actually

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

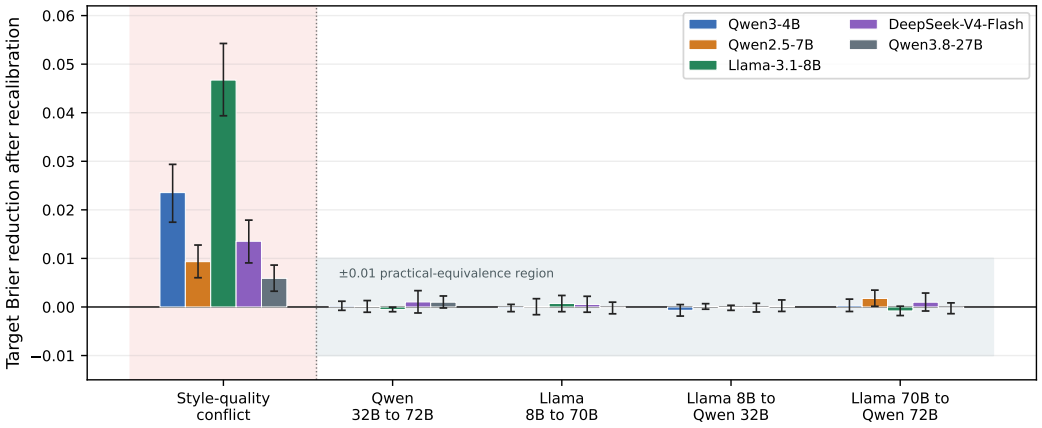

Figure 1: **A controlled positive case and a generator-change boundary persist across five diverse judges.** Bars show five-fold out-of-fold (OOF) target Brier-risk reduction from replacing the source map with a target-refit map; positive values favor recalibration. Error bars are 95% prompt-cluster bootstrap intervals. In the RM-Bench style conflict (Liu et al., 2025b), the preferred response has a weaker style signal than the rejected response, and all five intervals exclude zero. For four prespecified JETTS scale and family shifts (Zhou et al., 2025), the shading marks the registered ± 0.01 Brier-risk equivalence margin. All 20 judge–transition 90% intervals lie within this margin; one canonical transition also has a positive 95% lower endpoint. Qwen3.8-27B was evaluated as a protocol-locked recent-model replication.

better. Reusing the same small sample for both steps invites overfitting. The operational problem is to make that decision with finite-sample safety while retaining enough power to act when the drift is meaningful.

Our protocol keeps repair fitting and certification separate. One target split fits a prespecified menu of standard calibration repairs (Platt, 1999; Zadrozny & Elkan, 2002; Kull et al., 2017); a disjoint split measures target Brier improvement over source reuse (Brier, 1950). The key observation is that the paired Brier-loss range is determined exactly by observable source–repair disagreement. This gives simultaneous lower bounds, safe near-oracle selection, and an exact continuous minimax allocation for the repair menu. For K repairs, a simpler disagreement envelope has an exact greedy integer solution under cell capacities and is within a factor $\sqrt{K}$ of the optimal worst-candidate radius. For fixed-repair binary-Brier certification, a testing lower bound matches the worst-case dependence on repair movement, gain, and confidence.

Our contributions are:

1. We formulate post-update LLM-judge calibration as a reuse-or-recalibration decision. Controlled, boundary, and external cases show that output change neither guarantees transfer nor implies that recalibration is necessary.
2. For a fixed target window, we give a finite-sample, distribution-free certificate for safe near-oracle selection among target-fitted repairs, up to certification radii. Observable disagreement yields the exact continuous minimax audit allocation for a repair menu; an exactly integer-optimizable envelope is $\sqrt{K}$-approximate in radius.
3. For fixed-repair binary-Brier certification, we establish a matching worst-case rate and evaluate the safety–power tradeoff through controlled simulation and group-disjoint deployment replays.

2 Related Work

LLM judges under output shift. LLM judges and reward models exhibit position, length, style, generator, and distribution-shift failures (Dubois et al., 2024; Bharadwaj et al., 2025; Panickssery et al., 2024; Dev et al., 2026; Pikus et al., 2023; Lamparth et al., 2026). Prior work studies debi-

asing, shelf life, evaluator replacement, degradation, and unlabeled correction (Huang et al., 2025; Dadkhahi et al., 2026; Singh et al., 2026; Yang et al., 2026; Khomiakov & Frellsen, 2026; Li et al., 2025). We study the keep-or-replace decision for an existing map.

Evaluation with imperfect judges. Limited-gold methods use trusted labels for population inference or judge uncertainty (Angelopoulos et al., 2023; Guerdan et al., 2025; Feng et al., 2026; Chen et al., 2026b; Landesberg & Narayan, 2025; Radharapu et al., 2026; Li, 2026). We instead select between source and target conditional reliability maps.

Calibration and safe selection under shift. Post-hoc and shift-aware calibration use standard maps and conditional invariances (Platt, 1999; Zadrozny & Elkan, 2002; Kull et al., 2017; Park et al., 2020; Wang et al., 2020; Popordanoska et al., 2024; Sun et al., 2023). Style–quality shift can violate them. Learn then Test (LTT) controls finite-sample risk over fixed candidates, while Adaptive LTT tests them sequentially (Angelopoulos et al., 2025; Zecchin et al., 2025). Our i.i.d. comparator is the corresponding repair-versus-reuse test; our audit also allocates labels across target states.

Active model comparison. Active comparison chooses instances under a label budget (Sawade et al., 2012); its asymptotic design uses the unknown label conditional, with absolute disagreement as a squared-loss limit. Our post-fit, label-free range design controls a repair menu and matches the fixed-repair binary-Brier rate.

Stratified allocation and sequential inference. Neyman stratification and confidence sequences use variance for allocation or stopping (Neyman, 1934; Carpentier & Munos, 2013; Morikawa et al., 2026; Waudby-Smith & Ramdas, 2022; Ryu & Wornell, 2024; Turner & Grünwald, 2023). For our fixed budget, post-fit disagreement gives a label-free range envelope.

3 METHOD

3.1 SETTING AND SHIFT DIAGNOSTIC

An evaluation item contains a prompt X, generated answer or answer pair R, a fixed judge output J, and a binary target label Y supplied by a human, benchmark preference, or objective verifier. The judge, rubric, parser, answer ordering, and label definition are fixed between a source environment s and a target environment t. Model weights, decoding, system prompts, tools, or the task mixture may change the outputs reaching the judge.

Let S collect prespecified deployment-observable attributes such as task, length, format, refusal, or confidence; continuous attributes are discretized by cutoffs fixed before target labels are opened. Define $C = (S, J) \in \mathcal{C}$. This finite registered contract fixes the map inputs and certified claim. Write

$$q_c = P_t(C = c), \qquad \eta_e(c) = P_e(Y = 1 \mid C = c), \quad e \in \{s, t\}.$$

For a fixed-window claim, P_t is the uniform empirical law of the registered window; in population extensions, it denotes the target population law. The contract may use source data and unlabeled target attributes, and is fixed before certification labels are opened. Finer contracts localize subgroup drift but fragment support and raise certification cost; a coarse contract certifies only aggregate map risk. Target-label-adaptive state discovery is a separate problem. It is useful to separate three consequences of an output change. A *usage shift* changes q_c and is visible without target labels. A *calibration shift* changes $\eta_t(c)$ in a target-important state. A *support shift* puts target mass on a state absent from source calibration. Generator, decoding, style, and task-mixture changes describe how deployment changed; these quantities describe what that change means for maintaining the judge map.

The following decomposition shows why frequency monitoring alone is insufficient. Let $b_c = \eta_s(c) - \eta_t(c)$, let $a_s(c) = \mathbb{E}\{\widehat{\eta}_s(c)\} - \eta_s(c)$ be source-estimation bias, and let $v_s(c) = \mathrm{Var}\{\widehat{\eta}_s(c)\}$.

Proposition 1 (Source-transfer error). *For* $\mathcal{E}_t(\widehat{\eta}_s) = \sum_c q_c\{\widehat{\eta}_s(c) - \eta_t(c)\}^2$,

$$\mathbb{E}\{\mathcal{E}_t(\widehat{\eta}_s)\} = \sum_c q_c\big[\{b_c + a_s(c)\}^2 + v_s(c)\big].$$

For an unbiased source estimator, transfer error is the sum of target-weighted conditional drift $\sum_c q_c b_c^2$ *and source-estimation variance* $\sum_c q_c v_s(c)$.

The proposition shows that a source map can fail simply because target traffic puts more weight on cells with large conditional drift, even when every target state is already supported. The proof is in Appendix B.

Unlabeled target traffic identifies changes in q_c and missing source support. It does not identify a change in $\eta_t(c)$ without a structural invariance assumption; two target distributions can share the same q_c and have different $Y \mid C$. Label-shift correction is one special case: if $P_t(C \mid Y) = P_s(C \mid Y)$, Bayes' rule transports η_s through the target class prior. Appendix B states this result and the non-identifiability argument formally.

We call a change from $P_s(Y \mid C)$ to $P_t(Y \mid C)$ that accompanies a changed association between response presentation and quality a *style–quality association shift*. We use this term for an LLM-specific mechanism producing conditional calibration shift; style may be a cue, a consequence of generation, or a proxy. Generator identity records a source of change; $\eta_s \neq \eta_t$ records the calibration failure relevant to repair.

We diagnose stale source calibration by grouped cross-fitting. In every fold, a source map and a target-refit map are fitted on source and target training prompts, respectively, and both are evaluated on the same held-out target prompts:

$$\widehat{G}_{\mathrm{OOF}} = \widehat{L}_t(\widehat{\eta}_s) - \widehat{L}_t(\widehat{\eta}_t), \qquad L_t(f) = \mathbb{E}_t\{Y - f(C)\}^2.$$

Here $\widehat{L}_t$ is held-out empirical target loss. Prompt counterparts and A/B order variants share a fold. Positive gain thus measures environment-specific calibration without a source–target split.

3.2 Single-repair certification

Let f_s be a fitted source map, let f_r be a target repair, and let $\ell(y, f)$ be the deployment loss. The maps are fixed before certification labels are opened. Define the target gain and its statewise observation as

$$\Delta = \mathbb{E}_t\{\ell(Y, f_s(C)) - \ell(Y, f_r(C))\}, \qquad D_c(Y) = \ell\{Y, f_s(c)\} - \ell\{Y, f_r(c)\}.$$

For candidate k, define $D_{c,k}$ by replacing f_r with $f_{r,k}$ and let $\mu_{c,k} = \mathbb{E}_t\{D_{c,k}(Y) \mid C = c\}$. If $D_{c,k}(Y) \in [l_{c,k}, u_{c,k}]$, write $w_{c,k} = u_{c,k} - l_{c,k}$. Set $f_{r,k} = f_s$ on inactive cells; then $D_{c,k} \equiv 0$ and $\Delta_k = \sum_{c \in \mathcal{C}} q_c \mu_{c,k}$ remains the full-window comparative gain, without renormalizing q. For one candidate, suppress k and write $D_c(Y) \in [l_c, u_c]$ and $w_c = u_c - l_c$. The range width is known from the fixed maps and loss; it need not use certification labels. Our primary setting is a fixed evaluation or release-gate window with exact empirical mass q_c from the entire preassigned certification window, excluding repair-fitting and test groups. Conditioning on the label-blind partition fixes this finite population before fitting. Future-population certificates in Appendix B also account for estimating q_c.

Theorem 1 (Range-weighted safe repair)**.** *Condition on all source data, repair-fit data, unlabeled target states, the resulting* f_s, f_r*, and a fixed feasible active set* $\mathcal{A}$ *satisfying* $f_r = f_s$ *outside* $\mathcal{A}$ *and* $q_c w_c > 0$ *within it; all following sums range over* $\mathcal{A}$*. In every* $c \in \mathcal{A}$*, sample* $n_c \geq 1$ *certification labels independently from the target conditional distribution or uniformly without replacement from a finite target window, and define*

$$\widehat{\Delta} = \sum_c q_c \frac{1}{n_c} \sum_{i=1}^{n_c} D_c(Y_{ci}), \qquad r_\delta = \sqrt{\frac{\log(2/\delta)}{2} \sum_c \frac{q_c^2 w_c^2}{n_c}}.$$

Deploy f_r *when* $\widehat{\Delta} - r_\delta > 0$*. With probability at least* $1 - \delta$*, a deployment lowers target loss; the repair is deployed whenever* $\Delta > 2r_\delta$*. For budget* $B = \sum_c n_c$*, let* $N_c \in [1, \infty]$ *be the available cell capacity, with* $N_c = \infty$ *for independent conditional sampling. The continuous allocation under* $1 \leq n_c \leq N_c$*, assumed feasible, has the unique minimizer*

$$n_c^* = \mathrm{clip}_{[1, N_c]}(\lambda q_c w_c),$$

where $\lambda > 0$ *is chosen so that* $\sum_c n_c^* = B$*. Without active upper caps this reduces to* $\max\{1, \lambda q_c w_c\}$*. If neither bound is active, then* $n_c^* = B q_c w_c / W$ *for* $W = \sum_c q_c w_c$*, giving* $r_\delta = W\sqrt{\log(2/\delta)/(2B)}$ *and making* $B > 2W^2 \log(2/\delta)/\Delta^2$ *sufficient for deployment.*

With sampling without replacement, μ_c and Δ denote realized finite-window means; a census may use radius zero. Our replays retain the displayed uncorrected radius as a common conservative protocol.

Appendix B combines stratified Hoeffding concentration (Hoeffding, 1963) with convex allocation. Greedy allocation exactly handles integer capacities for a fixed active set. Sampling finite windows without replacement is conservative; the optimized quantity remains the displayed range radius, not a finite-population correction.

For binary Brier loss, let $a_c = f_r(c) - f_s(c)$. Then

$$D_c(Y) = a_c\{2Y - f_s(c) - f_r(c)\}, \qquad w_c = 2|a_c|.$$

Writing $\Gamma = \sum_c q_c|a_c|$ gives $W = 2\Gamma$, so the theorem allocates $n_c \propto q_c|a_c|$ and recovers the stated Γ^2/Δ^2 certification rate. The same construction covers bounded continuous, multiclass, ordinal, and clipped log losses after registering the corresponding range widths, given in Appendix B.5.

This allocation is the worst-case counterpart of Neyman allocation. If $Y \mid C = c \sim \mathrm{Bernoulli}(\eta_t(c))$, then $\mathrm{sd}\{D_c(Y)\} = 2|a_c|\sqrt{\eta_t(c)\{1-\eta_t(c)\}} \le |a_c|$. Neyman allocation would use the unknown standard deviation; disagreement allocation uses its sharp distribution-free envelope and is fixed before certification outcomes are observed.

3.3 Repair-menu selection

Several target-fitted repairs can share the same certification labels, so there is no need to audit them separately. We choose one allocation that minimizes the worst radius over the registered menu.

Theorem 2 (Minimax repair-menu allocation)**.** *For K fixed repairs on shared active set $\mathcal{A}_\cup$, let $N_c \ge 1$ be cell capacities, $w_c^{\max} = \max_k w_{c,k}$, and $\mathcal{S}_K = \{\lambda \in \mathbb{R}_+^K : \sum_k \lambda_k = 1\}$; all sums below range over $\mathcal{A}_\cup$. Define $H_k(n) = \sum_c q_c^2 w_{c,k}^2/n_c$ and $\mathcal{N}_B = \{n : 1 \le n_c \le N_c, \sum_c n_c = B\}$, assumed nonempty. The continuous shared-audit problem $\min_{n\in\mathcal{N}_B} \max_k H_k(n)$ has an optimum whose positive aggregate-coefficient cells satisfy*

$$n_c^* = \mathrm{clip}_{[1,N_c]}\left(\alpha q_c \sqrt{\sum_k \lambda_k^* w_{c,k}^2}\right), \quad \lambda^* \in \mathcal{S}_K,$$

where λ^* solves a K-dimensional concave dual. Zero-coefficient cells stay at one unless positive-coefficient cells saturate; any residual budget is assigned among them to recover a primal optimum. The multiplier α enforces the remaining budget on positive-coefficient cells.

Proof sketch. Write $\max_k H_k = \max_{\lambda\in\mathcal{S}_K} \sum_k \lambda_k H_k$ and, by the minimax theorem, exchange min and max, leaving a separable convex allocation for fixed λ. The Karush–Kuhn–Tucker (KKT) conditions equalize marginal decreases on unsaturated cells and clip at capacities. For the integer envelope, the next label in cell c gains $q_c^2(w_c^{\max})^2/[n_c(n_c+1)]$; diminishing gains make largest-gain greedy exchange-optimal. Thus λ weights worst-radius candidates and $K = 1$ recovers disagreement allocation; floor-and-fill and the envelope are $\sqrt{2}$- and $\sqrt{K}$-approximate in radius. Appendix B gives formal proofs.

If one candidate k^* attains $w_c^{\max}$ in every active cell, the envelope objective equals $H_{k^*}(n) = \max_k H_k(n)$ and is exactly minimax. Gains are possible only when worst movement directions are split across candidates; Section 4 tests this distinction.

Corollary 1 (Safe near-oracle repair selection)**.** *Let $r_k = \{\log(2K/\delta)H_k(n)/2\}^{1/2}$ and $L_k = \widehat{\Delta}_k - r_k$. Using registered ties, let $\widehat{k} \in \arg\max_k L_k$; deploy it with $\Delta_{\mathrm{dep}} = \Delta_{\widehat{k}}$ if $L_{\widehat{k}} > 0$, and otherwise reuse with $\Delta_{\mathrm{dep}} = 0$. With probability at least $1-\delta$,*

$$\Delta_{\mathrm{dep}} \ge \max\left\{0, \max_k(\Delta_k - 2r_k)\right\}.$$

If the gate reuses, then $\Delta_k \le 2r_k$ for all registered repairs.

Thus selecting among the registered repairs does not invalidate the certificate: the gate either deploys a near-best certifiable repair or reuses while every candidate remains below its current resolution.

A fixed-budget empirical-Bernstein refinement retains the label-free design and simultaneous control (Proposition 6), but its $1/n_c$ term can dominate small cells; adaptation or stopping needs anytime-valid inference.

Proposition 2 (Matching fixed-repair binary-Brier certification cost)**.** *For fixed $\beta \in (0,1)$, $\delta \in (0, 1-\beta)$, and binary Brier loss, any rule with deployment probability at most δ for every non-improving repair and power at least $1-\beta$ for gain $0 < \Delta \le \Gamma/2$ requires, on some target distribution,*

$$B = \Omega\left(\frac{\Gamma^2}{\Delta^2} \log \frac{1}{\delta}\right).$$

Thus the fixed-repair binary-Brier rate is optimal up to constants.

Appendix B gives variance-adaptive, future-population, and estimated-mass extensions.

3.4 Fit–certify–deploy protocol

Algorithm 1: simultaneous certified repair. Inputs are source groups, an unlabeled target pool, total grouped target-label budget B_{tot}, a prespecified menu of repair procedures, fit fraction ρ, and level δ.

1. Register C, the candidate menu, confidence level, prompt-level audit unit, support rule, and tie order before opening target labels. Partition target groups into fitting, certification, and untouched-test pools. Fit f_s. For a fixed-window claim, compute q_c over the entire certification pool. For a future-population claim, use $\widehat{q}_c$ from a registered unlabeled sample and register $\|\widehat{q} - q\|_1 \le \epsilon_q$ and $M \ge \max_{c,k} |\mu_{c,k}|$, or reserve the independent state-only sample in Proposition 5.
2. Set $B_{\mathrm{fit}} = \mathrm{round}(\rho B_{\mathrm{tot}})$ and $B_{\mathrm{cert}} = B_{\mathrm{tot}} - B_{\mathrm{fit}}$. Spend B_{fit} labels to fit the candidates and freeze them. Retain states with $q_c > 0$, nonzero disagreement, and positive certification capacity; use f_s elsewhere.
3. Allocate the B_{cert} certification labels; this is the budget B in Theorems 1 and 2. Apply floor-and-fill rounding to the continuous minimax design: floor its counts, then assign residual labels by greedy reduction in $\max_k H_k$. The envelope is exactly integer-optimized. If needed, fix the feasible active set by retaining the largest $q_c w_c^{\max}$; the variance-adaptive option requires two labels per state.
4. Compute $\widehat{\Delta}_k - r_k$, subtracting $M\epsilon_q$ when q is replaced by $\widehat{q}$ (Appendix B.9), and use either the range radius or Proposition 6. Deploy the candidate with the largest positive bound, using the registered tie order; otherwise retain f_s.
5. Evaluate once on untouched groups and report risk, switch rate, held-out test harm, confidence radius, and budget. Test risk is a separate generalization diagnostic, not the certified window risk.

Validity holds for every fixed feasible allocation; minimax and its approximations affect efficiency, not safety. Certification labels remain independent of fitting, support, and allocation, which use only source, unlabeled-target, and disjoint fit data.

4 Experiments and Discussion

4.1 Experimental setup

RM-Bench, JETTS, and RealDR cover controlled, boundary, and external cases; MATH500 and HumanEval isolate presentation. Appendix Table 4 records their estimands.

Core shifts. RM-Bench (Liu et al., 2025b) supplies 15,924 ordered pairs from 1,327 prompt groups. The source pairs match style levels across preferred and rejected responses; the target pairs deliberately give the rejected response the stronger style signal. JETTS (Zhou et al., 2025) instead provides verifier-labeled math, code, and instruction-following outputs. Before collecting complete-fold outputs, we froze four transitions: Qwen 32B-to-72B, Llama 8B-to-70B, Llama 8B-to-Qwen

32B, and Llama 70B-to-Qwen 72B. We retain every shared prompt with at least one correct and one incorrect response in both regimes: 1,229 prompts and 9,960 order-expanded rows, with one pair selected deterministically per prompt–generator regime. The frozen 0.01 absolute Brier margin defines practical equivalence only on this jointly pair-eligible set; Appendix Table 10 quantifies its retention scope. RealDR (Chen et al., 2026a) contains 40 research tasks, four report generators, and four generation prompts per task. Its 640 reports carry weighted expert scores; expert and judge scores are divided by 10 and analyzed with the $[0, 1]$ squared-loss extension in Appendix B.5. We prespecify detailed-prompt-to-weakened, authoritative, and raw-query shifts plus four leave-one-generator-out folds, all with task-grouped inference.

Judges and controls. The full RM-Bench and JETTS canonical panel contains Qwen3-4B, Qwen2.5-7B, Llama-3.1-8B, and the fixed-date DeepSeek-V4-Flash API under one structured pairwise prompt (Qwen Team, 2025; Yang et al., 2024; Llama Team, 2024; DeepSeek-AI, 2026). Skywork Reward V2 tests a scalar interface (Liu et al., 2025a); Prometheus-7B-v2 uses its native template (Kim et al., 2024); Qwen3.8-27B is a protocol-locked recent-model replication under the same prompt, folds, and estimand (Qwen Team, 2026). MATH500 and HumanEval preserve objective labels under presentation changes (Hendrycks et al., 2021; Chen et al., 2021). LLMBar, RewardBench, JudgeBench, and a position-balanced English non-tie subset of LMArena's `arena-human-preference-100k` data provide high-, low-, and support-shift checks (Zeng et al., 2024; Lambert et al., 2025; Tan et al., 2025; Tang et al., 2025; Chiang et al., 2024). The Appendix exposure audit separates pre-RM-Bench checkpoints from interface replications; Skywork reports RM-Bench evaluation and is not contamination-controlled evidence.

Baselines, splits, and metrics. Source reuse is compared with six repairs, in registered tie order: source-prior shrinkage, target-only cells, Platt, isotonic, beta, and Platt+cell (Platt, 1999; Zadrozny & Elkan, 2002; Kull et al., 2017). Unlabeled comparators are label-shift expectation–maximization (EM), a fixed length penalty, and locally weighted smoothing (LOWESS) reward correction (Saerens et al., 2002; Huang et al., 2025; Cleveland, 1979). Deployment rules include naive validation, which samples the certification share uniformly and selects the largest positive empirical gain without a radius, as well as LTT–IID Hoeffding, paired empirical Bernstein, uniform, target-mass, minimax-menu and disagreement-envelope allocation, and an oracle-variance ceiling (Angelopoulos et al., 2025; Maurer & Pontil, 2009). Each canonical RM-Bench judge is replayed 100 times at total target budgets $\{25, 50, 100, 200, 400\}$; reported budgets are B_{tot}, while theoretical B is the certification share. Source-history, target fit, certification, and test prompt groups are disjoint; A/B variants never cross partitions. We report Brier risk, OOF recalibration gain with prompt-cluster intervals, switch rate, and the fraction of all replays that deploy a repair with higher untouched-test risk than source reuse. Appendix C gives protocol details, secondary baselines, and per-judge robustness summaries; full records remain in the artifact.

With source-history cell size $n_{s,c}$ and positive-label count $y^+_{s,c}$, source cells use $f_s(c) = (y^+_{s,c} + 0.5)/(n_{s,c} + 1)$; unseen cells use the global source-history label mean (0.5 only if the history is empty). The source-prior pseudo-count is $\tau = 10$. In discrete-state replays, the entire preassigned certification pool is the finite population, so its q is exact and $\epsilon_q = 0$; fitting and test groups are excluded. Inactive cells retain their target mass and contribute zero comparative loss. A nonpositive best bound returns source reuse. Rounded minimax-menu allocation is primary; the integer envelope is its solver-free surrogate.

Five prompt-grouped diagnostic folds use verdict-only states, which certify aggregate risk without a within-style claim; enriched contracts are separate sensitivity analyses. Deployment retains one random A/B order per prompt and splits labels equally between fitting and certification (Appendix Table 23). The displayed menu order, tuning, split, and $\delta = 0.05$ are fixed before certification outcomes.

4.2 When does source calibration become stale?

Figure 1 shows a clear contrast between the style conflict and the generator transitions. RM-Bench target refitting lowers held-out Brier risk by 0.0235, 0.0093, 0.0467, and 0.0135 for Qwen3-4B, Qwen2.5-7B, Llama-3.1-8B, and DeepSeek-V4-Flash, respectively; every prompt-cluster interval excludes zero. JETTS gains are much smaller, from -0.00081 to 0.00175 across the same judges.

All 16 90% intervals lie inside the frozen $[-0.01, 0.01]$ region, so every transition is practically equivalent; one Qwen2.5 transition is statistically detectable while still inside that margin. The four prompt-macro estimates are also practically equivalent, and the conclusions are unchanged at margins 0.005 and 0.02 (Appendix Table 12). Qwen3.8-27B follows the same pattern under the locked protocol: positive RM-Bench gain, 0.0059 [0.0032, 0.0086], but practical equivalence on all four JETTS transitions (Appendix C).

Table 1: Grouped five-fold source transfer on full RM-Bench. Both maps are evaluated on the same held-out target prompts. Gain is source risk minus target-refit risk; intervals use prompt-cluster bootstrap.

Judge	Source risk	Target refit	Gain	95% CI
Qwen3-4B	0.2680	0.2444	0.0235	[0.0175, 0.0294]
Qwen2.5-7B	0.2511	0.2418	0.0093	[0.0060, 0.0127]
Llama-3.1-8B	0.2963	0.2496	0.0467	[0.0393, 0.0542]
DeepSeek-V4-Flash	0.2349	0.2214	0.0135	[0.0091, 0.0179]
Qwen3.8-27B	0.1739	0.1681	0.0059	[0.0032, 0.0086]

Table 1 reports the absolute risks behind the gains. The source map is evaluated on held-out target prompts, so each comparison holds target examples and labels fixed. Effect size varies by nearly an order of magnitude, from 0.0059 for Qwen3.8-27B to 0.0467 for Llama-3.1-8B. Qwen3.8-27B has the lowest source-transfer Brier risk among the five judges, yet its positive refit gain shows that lower absolute judge risk does not eliminate calibration staleness under the controlled style conflict.

Target-unlabeled label-shift EM does not resolve the RM-Bench failure. Balanced AB/BA ordering fixes the true target prior at 0.5, whereas EM estimates 0.410, 0.563, and 0.340 for the three open canonical judges and increases Brier risk by 0.0055–0.0084. The changed state frequencies arise from a changed $P(C \mid Y)$, which violates the baseline's invariance condition.

The scalar reward-model result separates cue debiasing from maintenance (Appendix Table 6). A fixed length penalty improves absolute risk, yet target refitting retains a significant 0.0062 gain. Target-traffic LOWESS performs poorly in absolute risk. Removing a known style cue can help the raw score without establishing that its source reliability map transfers.

4.3 Finite-sample safety and deployment power

Controlled simulation checks the theorem independently of LLM modeling (Appendix Table 5). Disagreement allocation reduces radius by about 20%, with at least 99.4% coverage and at most 0.22% deployment at the exact zero-gain boundary. Benchmark replays instead ask how often selection raises risk on the untouched test split.

Table 2: Canonical RM-Bench decisions over 100 prompt-disjoint replays using the rounded minimax-menu allocation. Gain and test harm use untouched splits; test harm differs from the Theorem 1 event $\Delta_k \leq 0$.

B_{tot}	Judge	Naive test harm	Held-out gain	Cert. switch	Cert. test harm
50	Qwen3-4B	0.08	0.0001	0.01	0.00
	Qwen2.5-7B	0.21	0.0000	0.01	0.00
	Llama-3.1-8B	0.04	0.0022	0.08	0.00
	DeepSeek-V4-Flash	0.18	0.0001	0.01	0.00
400	Qwen3-4B	0.02	0.0054	0.22	0.00
	Qwen2.5-7B	0.10	0.0003	0.03	0.00
	Llama-3.1-8B	0.01	0.0230	0.57	0.00
	DeepSeek-V4-Flash	0.06	0.0009	0.05	0.00

Table 2 shows mostly reuse at $B_{\text{tot}} = 50$; at $B_{\text{tot}} = 400$ the gate realizes a 0.0230 Llama gain, with no observed held-out test-harm event in the eight minimax cells. This differs from the theorem event; Appendix Table 9 gives Wilson intervals.

When does menu-aware minimax matter? On the canonical menu, radii agree at displayed precision for $B_{\text{tot}} \in \{200, 400\}$ and deployment is comparable, so the envelope suffices (Appendix Table 25). Holding six candidates, 42 states, and $B = 420$ fixed, the controlled sweep moves disagreement from shared to candidate-private blocks; the envelope then needs 666 versus 420 labels ($1.59\times$) to match the minimax radius (Appendix Table 26). This core evidence for Theorem 2 localizes its gain to heterogeneous menus.

Power tracks the effect-to-radius ratio: the gate reuses at small budgets and switches most for Llama-3.1-8B, the largest RM-Bench effect. A registered output-style contract switches in 0.980/1.000 of replays at $B_{\text{tot}} = 200/400$; at $B_{\text{tot}} = 400$ it gains 0.1142 with no observed test harm (Appendix Table 24).

Despite positive OOF gain, the Qwen3.8-27B gate reuses through $B_{\text{tot}} = 400$; naive validation switches in 64–86% of replays with 12–21% test harm (Appendix Table 13).

4.4 External case and signal regimes

On RealDR, two gains are 0.00152 [0.00046, 0.00262] and 0.00198 [0.00054, 0.00343], while five intervals cover zero. Both range gates reuse in all 700 replays at $B_{\text{tot}} = 30$, which labels the complete 15-task window but retains the common uncorrected radius. Setting the census radius to zero instead switches in 0.823 of replays with no window harm; 0.240 incur harm on untouched tasks (Appendix Tables 14 and 15). The range-gate non-switch therefore reflects its conservative bound, whereas the latter harm is generalization beyond the certified window.

Table 3: Additional regimes at $B_{\text{tot}} = 200$: switch / held-out test-harm rates over seven judges and 100 grouped replays per judge. Certified uses single-repair disagreement allocation on the preassigned certification window.

Shift	Naive validation	Certified gate
LLMBar natural→adversarial	1.000 / 0.000	1.000 / 0.000
RewardBench easy→hard	0.453 / 0.261	0.000 / 0.000
LMArena non-code→code	0.721 / 0.203	0.084 / 0.013

Table 3 shows that the disagreement gate switches on LLMBar in 1.000 of replays versus 0.447 for LTT–IID Hoeffding, with zero observed harm for both; it suppresses RewardBench switching and reduces both switching and harm under LMArena. Appendix C gives full comparisons.

Cross-regime interpretation. The largest RM-Bench effects and LLMBar are stale and certifiable; JETTS remains practically transferable, while Qwen3.8 remains unresolved at $B_{\text{tot}} = 400$. RealDR's census removes sampling uncertainty for its finite window, but not generalization uncertainty on untouched tasks. Thus a non-switch says only that no registered repair clears the current radius; LMArena separately shows how unsupported states make auditing conservative.

Mechanism and scope. Label-preserving controls produce 17.8% average judge flips on MATH500 and 5.5–19.8% on HumanEval; state ablations separate verdict-only drift from enriched-contract support loss (Appendix Tables 18 and 32). Guarantees require a compact contract, random within-state labels, a fixed judge, and fit–certify independence.

5 Conclusion

A model update alone does not determine whether an LLM judge needs recalibration. The overlap sweep identifies when menu-aware auditing matters: the envelope needs up to $1.59\times$ the labels to match the worst-candidate radius under heterogeneous movements but is nearly identical on the canonical menu. The fixed-repair binary-Brier rate is optimal up to constants. Qwen3.8 shows that positive OOF evidence can remain below sampled-audit resolution; the RealDR census separates exact finite-window selection from generalization to untouched tasks.

Power depends on the contract, candidate disagreement, and budget: richer contracts localize drift but fragment support. Next is anytime-valid, label-adaptive auditing that preserves fit independence and post-selection safety.

AI USE STATEMENT

Generative AI tools were used to brainstorm and refine the research questions and hypotheses; discover and summarize relevant literature; develop the conceptual and theoretical framework; formulate, critique, and edit mathematical claims and proof drafts; design the methodology, experiments, and ablations; suggest experimental parameters; implement, debug, and orchestrate code; assist with dataset preprocessing and reformatting; interpret results; prepare figures and tables; and draft, translate, restructure, and edit the manuscript, including its title and references. Generative AI tools did not provide ground-truth labels or independently determine which results or claims to report. The authors reviewed all AI-assisted work: cited claims were checked against the original sources, mathematical statements and proofs were examined by the authors, code was executed and tested, and saved experimental artifacts and reported values were inspected. The authors made all final research and reporting decisions and take responsibility for the complete content of this work.

REPRODUCIBILITY STATEMENT

The appendix gives proofs, assumptions, configurations, split rules, and per-judge results. The artifact records estimator definitions, preprocessing, protocols, prompt hashes, judge outputs, and table and figure commands.

REFERENCES

Anastasios N. Angelopoulos, Stephen Bates, Clara Fannjiang, Michael I. Jordan, and Tijana Zrnic. Prediction-powered inference. *CoRR*, abs/2301.09633, 2023. doi: 10.48550/ARXIV.2301.09633. URL https://doi.org/10.48550/ARXIV.2301.09633.

Anastasios N. Angelopoulos, Stephen Bates, Emmanuel J. Candès, Michael I. Jordan, and Lihua Lei. Learn then Test: Calibrating predictive algorithms to achieve risk control. *The Annals of Applied Statistics*, 19(2):1641–1662, 2025.

Sher Badshah, Ali Emami, and Hassan Sajjad. SCOPE: Selective conformal optimized pairwise LLM judging. *CoRR*, abs/2602.13110, 2026. doi: 10.48550/ARXIV.2602.13110. URL https://doi.org/10.48550/ARXIV.2602.13110.

Anirudh Bharadwaj, Chaitanya Malaviya, Nitish Joshi, and Mark Yatskar. Flattery, fluff, and fog: Diagnosing and mitigating idiosyncratic biases in preference models. *CoRR*, abs/2506.05339, 2025. doi: 10.48550/ARXIV.2506.05339. URL https://doi.org/10.48550/ARXIV.2506.05339.

Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950.

Alexandra Carpentier and Rémi Munos. Toward optimal stratification for stratified monte-carlo integration. In *Proceedings of the 30th International Conference on Machine Learning (ICML)*, pp. 28–36, 2013.

Junjie Chen, Yuxi Dong, Haitao Li, Weihang Su, Yujia Zhou, Min Zhang, Yiqun Liu, and Qinyao Ai. Benchmarking LLM-as-a-judge for long-form output evaluation. *CoRR*, abs/2606.01629, 2026a. doi: 10.48550/ARXIV.2606.01629. URL https://doi.org/10.48550/ARXIV.2606.01629.

Lingjiao Chen, Matei Zaharia, and James Zou. How is ChatGPT's behavior changing over time? *CoRR*, abs/2307.09009, 2023. doi: 10.48550/ARXIV.2307.09009. URL https://doi.org/10.48550/ARXIV.2307.09009.

Mark Chen, Jerry Tworek, Heewoo Jun, et al. Evaluating large language models trained on code. *CoRR*, abs/2107.03374, 2021. doi: 10.48550/ARXIV.2107.03374. URL https://doi.org/10.48550/ARXIV.2107.03374.

Yiqun T. Chen, Sizhu Lu, Sijia Li, Moran Guo, and Shengyi Li. Efficient inference for noisy LLM-as-a-judge evaluation. *CoRR*, abs/2601.05420, 2026b. doi: 10.48550/ARXIV.2601.05420. URL https://doi.org/10.48550/ARXIV.2601.05420.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Banghua Zhu, Hao Zhang, Michael I. Jordan, Joseph E. Gonzalez, and Ion Stoica. Chatbot arena: An open platform for evaluating LLMs by human preference. In *Forty-first International Conference on Machine Learning (ICML)*, pp. 8359–8388. OpenReview.net, 2024. URL https://openreview.net/forum?id=3MW8GKNyzI.

William S. Cleveland. Robust locally weighted regression and smoothing scatterplots. *Journal of the American Statistical Association*, 74(368):829–836, 1979.

Hamid Dadkhahi, Firas Trabelsi, Parker Riley, Juraj Juraska, and Mehdi Mirzazadeh. Distribution-calibrated inference time compute for thinking LLM-as-a-judge. In *International Conference on Machine Learning (ICML)*, 2026.

DeepSeek-AI. DeepSeek V4 preview release. DeepSeek API Documentation, April 2026. URL https://api-docs.deepseek.com/news/news260424/.

Sunishchal Dev, Andrew Sloan, Joshua Kavner, Nicholas Kong, and Morgan Sandler. Judge reliability harness: Stress testing the reliability of LLM judges. *CoRR*, abs/2603.05399, 2026. doi: 10.48550/ARXIV.2603.05399. URL https://doi.org/10.48550/ARXIV.2603.05399.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B. Hashimoto. Length-controlled AlpacaEval: A simple way to debias automatic evaluators. *CoRR*, abs/2404.04475, 2024. doi: 10.48550/ARXIV.2404.04475. URL https://doi.org/10.48550/ARXIV.2404.04475.

Chen Feng, Minghe Shen, Ananth Balashankar, Carsten Gerner-Beuerle, and Miguel R. D. Rodrigues. Noisy but valid: Robust statistical evaluation of LLMs with imperfect judges. *CoRR*, abs/2601.20913, 2026. doi: 10.48550/ARXIV.2601.20913. URL https://doi.org/10.48550/ARXIV.2601.20913.

Benjamin Feuer, Lucas Rosenblatt, and Oussama Elachqar. Towards provably unbiased LLM judges via bias-bounded evaluation. *CoRR*, abs/2603.05485, 2026. doi: 10.48550/ARXIV.2603.05485. URL https://doi.org/10.48550/ARXIV.2603.05485.

Luke Guerdan, Justin Whitehouse, Kimberly Le Truong, Kenneth Holstein, and Zhiwei Steven Wu. Doubly-robust LLM-as-a-judge: Externally valid estimation with imperfect personas. *CoRR*, abs/2509.22957, 2025. doi: 10.48550/ARXIV.2509.22957. URL https://doi.org/10.48550/ARXIV.2509.22957.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *NeurIPS Datasets and Benchmarks*, 2021.

Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963.

Zeyu Huang, Zihan Qiu, Zili Wang, Edoardo M. Ponti, and Ivan Titov. Post-hoc reward calibration: A case study on length bias. In *International Conference on Learning Representations (ICLR)*, 2025.

Maxim Khomiakov and Jes Frellsen. Noise-response calibration: A causal intervention protocol for LLM-judges. *CoRR*, abs/2603.17172, 2026. doi: 10.48550/ARXIV.2603.17172. URL https://doi.org/10.48550/ARXIV.2603.17172.

Seungone Kim, Juyoung Suk, Shayne Longpre, Bill Yuchen Lin, Jamin Shin, Sean Welleck, Graham Neubig, Moontae Lee, Kyungjae Lee, and Minjoon Seo. Prometheus 2: An open source language model specialized in evaluating other language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 4334–4353, 2024. doi: 10.18653/V1/2024.EMNLP-MAIN.248.

Emma Yanyang Kong, JJ Tan, Ishan Gupta, Lars Olds, Claire Campbell, David Fagnan, Veli Balin, Rohan Gosain, Louis Garcia, and Minsu Jang. The lifecycle of LLM-as-a-judge for large-scale recommendation explanations. *CoRR*, abs/2608.18300, 2026. doi: 10.48550/ARXIV.2608.18300. URL https://doi.org/10.48550/ARXIV.2608.18300.

Meelis Kull, Telmo de Menezes e Silva Filho, and Peter A. Flach. Beta calibration: a well-founded and easily implemented improvement on logistic calibration for binary classifiers. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pp. 623–631, 2017.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, Lester James V. Miranda, Bill Yuchen Lin, Khyathi Raghavi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, Noah A. Smith, and Hannaneh Hajishirzi. Rewardbench: Evaluating reward models for language modeling. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 1755–1797, 2025. doi: 10.18653/V1/2025.FINDINGS-NAACL.96.

Max Lamparth, Daniel Fein, Andreas Haupt, Marcel Hussing, and Mykel J. Kochenderfer. Reward bias substitution: Single-axis bias mitigations redirect optimization pressure. *CoRR*, abs/2605.27996, 2026. doi: 10.48550/ARXIV.2605.27996. URL https://doi.org/10.48550/ARXIV.2605.27996.

Eddie Landesberg and Manjari Narayan. Causal judge evaluation: Calibrated surrogate metrics for LLM systems. *CoRR*, abs/2512.11150, 2025. doi: 10.48550/ARXIV.2512.11150. URL https://doi.org/10.48550/ARXIV.2512.11150.

Haitao Li, Junjie Chen, Qingyao Ai, Zhumin Chu, Yujia Zhou, Qian Dong, and Yiqun Liu. Calibraeval: Calibrating prediction distribution to mitigate selection bias in LLMs-as-judges. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 16537–16552, 2025. doi: 10.18653/V1/2025.ACL-LONG.808.

Yanran Li. Calibrate, don't curate: Label-efficient estimation from noisy LLM judges. *CoRR*, abs/2605.09702, 2026. doi: 10.48550/ARXIV.2605.09702. URL https://doi.org/10.48550/ARXIV.2605.09702.

Chris Yuhao Liu, Liang Zeng, Yuzhen Xiao, Jujie He, Jiacai Liu, Chaojie Wang, Rui Yan, Wei Shen, Fuxiang Zhang, Jiacheng Xu, Yang Liu, and Yahui Zhou. Skywork-Reward-V2: Scaling preference data curation via human–ai synergy. *CoRR*, abs/2507.01352, 2025a. doi: 10.48550/ARXIV.2507.01352. URL https://doi.org/10.48550/ARXIV.2507.01352.

Yantao Liu, Zijun Yao, Rui Min, Yixin Cao, Lei Hou, and Juanzi Li. RM-Bench: Benchmarking reward models of language models with subtlety and style. In *International Conference on Learning Representations (ICLR)*, 2025b.

Llama Team. The Llama 3 herd of models. *CoRR*, abs/2407.21783, 2024. doi: 10.48550/ARXIV.2407.21783. URL https://doi.org/10.48550/ARXIV.2407.21783.

Andreas Maurer and Massimiliano Pontil. Empirical bernstein bounds and sample-variance penalization. In *Conference on Learning Theory (COLT)*, 2009.

Kosuke Morikawa, Mst Moushumi Pervin, and Jae Kwang Kim. TS–Neyman: Posterior sampling for adaptive stratified estimation. *CoRR*, abs/2606.08418, 2026. doi: 10.48550/ARXIV.2606.08418. URL https://doi.org/10.48550/ARXIV.2606.08418.

Jerzy Neyman. On the two different aspects of the representative method: The method of stratified sampling and the method of purposive selection. *Journal of the Royal Statistical Society*, 97(4): 558–625, 1934.

Arjun Panickssery, Samuel R. Bowman, and Shi Feng. LLM evaluators recognize and favor their own generations. *CoRR*, abs/2404.13076, 2024. doi: 10.48550/ARXIV.2404.13076. URL https://doi.org/10.48550/ARXIV.2404.13076.

Sangdon Park, Osbert Bastani, James Weimer, and Insup Lee. Calibrated prediction with covariate shift via unsupervised domain adaptation. In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics (AISTATS)*, pp. 3219–3229, 2020.

Benjamin Pikus, Will LeVine, Tony Chen, and Sean Hendryx. A baseline analysis of reward models' ability to accurately analyze foundation models under distribution shift. *CoRR*, abs/2311.14743, 2023. doi: 10.48550/ARXIV.2311.14743. URL https://doi.org/10.48550/ARXIV.2311.14743.

John C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in Large Margin Classifiers*, pp. 61–74. MIT Press, 1999.

Teodora Popordanoska, Gorjan Radevski, Tinne Tuytelaars, and Matthew B. Blaschko. LaSCal: Label-shift calibration without target labels. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Qwen Team. Qwen3 technical report. *CoRR*, abs/2505.09388, 2025. doi: 10.48550/ARXIV.2505.09388. URL https://doi.org/10.48550/ARXIV.2505.09388.

Qwen Team. Qwen3.8-27B model card. Model release, August 2026. URL https://huggingface.co/Qwen/Qwen3.8-27B.

Bhaktipriya Radharapu, Eshika Saxena, Kenneth Li, Chenxi Whitehouse, Adina Williams, and Nicola Cancedda. Calibrating LLM judges: Linear probes for fast and reliable uncertainty estimation. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (ACL), Industry Track*, pp. 193–216, 2026.

Jongha Jon Ryu and Gregory W. Wornell. Gambling-based confidence sequences for bounded random vectors. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, pp. 42856–42869, 2024.

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.

Christoph Sawade, Niels Landwehr, and Tobias Scheffer. Active comparison of prediction models. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 1763–1771, 2012.

Janvijay Singh, Austin Xu, Yilun Zhou, Yefan Zhou, Dilek Hakkani-Tur, and Shafiq Joty. On the shelf life of fine-tuned LLM judges: Future proofing, backward compatibility, and question generalization. In *International Conference on Learning Representations (ICLR)*, 2026.

Zeyu Sun, Dogyoon Song, and Alfred O. Hero III. Minimum-risk recalibration of classifiers. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Sijun Tan, Siyuan Zhuang, Kyle Montgomery, William Yuan Tang, Alejandro Cuadron, Chenguang Wang, Raluca A. Popa, and Ion Stoica. Judgebench: A benchmark for evaluating LLM-based judges. In *International Conference on Learning Representations (ICLR)*, 2025.

Kelly Tang, Wei-Lin Chiang, and Anastasios N. Angelopoulos. Arena explorer: A topic modeling pipeline for LLM evals & analytics. Dataset release, 2025. URL https://huggingface.co/datasets/lmarena-ai/arena-human-preference-100k.

Rosanne J. Turner and Peter D. Grünwald. Safe sequential testing and effect estimation in stratified count data. In *Proceedings of the 26th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pp. 4880–4893, 2023.

Ximei Wang, Mingsheng Long, Jianmin Wang, and Michael I. Jordan. Transferable calibration with lower bias and variance in domain adaptation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Ian Waudby-Smith and Aaditya Ramdas. Estimating means of bounded random variables by betting. *CoRR*, abs/2010.09686, 2022. doi: 10.48550/ARXIV.2010.09686. URL https://doi.org/10.48550/ARXIV.2010.09686.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report. *CoRR*, abs/2412.15115, 2024. doi: 10.48550/ARXIV.2412.15115. URL https://doi.org/10.48550/ARXIV.2412.15115.

Zongyou Yang, Yinghan Hou, and Xiaokun Yang. When the judge changes, so does the measurement: Auditing LLM-as-judge reliability. *CoRR*, abs/2607.08535, 2026. doi: 10.48550/ARXIV.2607.08535. URL https://doi.org/10.48550/ARXIV.2607.08535.

Bianca Zadrozny and Charles Elkan. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 694–699, 2002. doi: 10.1145/775047.775151.

Matteo Zecchin, Sangwoo Park, and Osvaldo Simeone. Adaptive learn-then-test: Statistically valid and efficient hyperparameter selection. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025.

Zhiyuan Zeng, Jiatong Yu, Tianyu Gao, Yu Meng, Tanya Goyal, and Danqi Chen. Evaluating large language models at evaluating instruction following. In *International Conference on Learning Representations (ICLR)*, 2024.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-Bench and chatbot arena. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Yilun Zhou, Austin Xu, Peifeng Wang, Caiming Xiong, and Shafiq Joty. Evaluating judges as evaluators: The JETTS benchmark of LLM-as-judges as test-time scaling evaluators. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, pp. 79436–79471, 2025.

A Evidence Roles and Estimands

Table 4: Core evidence protocol. Rows play distinct inferential roles; only completed results enter the paper's claims.

Role	Data	Target label	Question
Positive case	RM-Bench	Benchmark preference	Does style–quality conflict render the source map stale?
Boundary case	JETTS complete	Objective verifier	Does generator identity alone imply repair?
External case	RealDR	Expert score	Do real generation-prompt or generator updates render a long-form score map stale?
Mechanism	MATH500; HumanEval	Objective verifier	Can presentation flip verdicts with fixed labels?
Deployment	RM-Bench replays	Benchmark preference	Can a scarce audit certify a lower-risk map?
Stress tests	LLMBar; RewardBench; LMArena non-code→code	Benchmark preference	Does the gate behave across drift and support regimes?

B Proofs and Formal Extensions

B.1 Proof of Proposition 1

Proof. For each state,

$$\widehat{\eta}_s(c) - \eta_t(c) = \{\eta_s(c) - \eta_t(c)\} + \{\widehat{\eta}_s(c) - \eta_s(c)\}.$$

The first term is b_c; the second has mean $a_s(c)$ and variance $v_s(c)$. Taking squared expectations statewise and summing with target weights q_c gives the result. □

B.2 WHY UNLABELED TARGET TRAFFIC IS INSUFFICIENT

Proposition 3 (No label-free certification without invariance)**.** *Even if the source map $\eta_s(c)$ and target masses q_c are known exactly, target calibration is not identifiable without target labels or an assumption linking η_t to η_s.*

Proof. Fix q_c and construct two target distributions with the same marginal $P(C = c) = q_c$. Set $\eta_0(c) = \eta_s(c)$. On any target-supported set $\mathcal{B}$, choose nonzero perturbations γ_c for which $\eta_s(c) + \gamma_c \in [0, 1]$ and set $\eta_1(c) = \eta_s(c) + \gamma_c$ on $\mathcal{B}$, leaving all other states unchanged. The two distributions induce identical unlabeled observations and different conditional label maps. No function of unlabeled target states can distinguish them. □

Proposition 4 (Label-shift special case)**.** *Suppose $P_t(C \mid Y) = P_s(C \mid Y)$ and the source and target class priors π_s, π_t lie in $(0, 1)$. With $\omega_\pi = \{\pi_t/(1-\pi_t)\}/\{\pi_s/(1-\pi_s)\}$,*

$$\eta_t(c) = \frac{\omega_\pi \eta_s(c)}{1 - \eta_s(c) + \omega_\pi \eta_s(c)}.$$

Hence label-free transport is possible when π_t is identifiable from the unlabeled target state masses and the class-conditional invariance holds.

Proof. Apply Bayes' rule in each environment and cancel the invariant likelihood ratio $P(C = c \mid Y = 1)/P(C = c \mid Y = 0)$. □

B.3 PROOF OF THEOREM 1

Proof. Conditional on the fitted maps and design, $\widehat{\Delta}$ is unbiased for Δ. For independent conditional labels, the summand in cell c has weighted range width $q_c w_c/n_c$; Hoeffding's inequality gives

$$P\left(|\widehat{\Delta} - \Delta| > r_\delta\right) \le \delta.$$

For uniform sampling without replacement, Hoeffding's finite-population reduction gives the same conservative bound for the realized window mean. On the complementary event, a positive lower confidence bound implies $\Delta > 0$. If $\Delta > 2r_\delta$, then $\widehat{\Delta} - r_\delta \ge \Delta - 2r_\delta > 0$, which proves the power claim.

Write $h_c = q_c w_c > 0$ on the active set; all sums in the remainder of this proof range over $c \in \mathcal{A}$. For any allocation with $\sum_c n_c = B$, Cauchy–Schwarz gives

$$\sum_c \frac{h_c^2}{n_c} \ge \frac{(\sum_c h_c)^2}{B}.$$

Without active bounds, equality holds at $n_c = Bh_c/W$. With $1 \le n_c \le N_c$, the objective is strictly convex and the KKT conditions give $n_c = \mathrm{clip}_{[1,N_c]}(\lambda h_c)$, where λ exhausts the budget. The unclipped substitution yields the displayed radius and sufficient budget condition. For integer n_c on this fixed active set, the marginal decrease from adding one label is $h_c^2/\{n_c(n_c+1)\}$; repeatedly taking the largest available decrease is optimal for this separable discrete-convex objective, including upper capacity constraints. For binary Brier loss, $w_c = 2|f_r(c) - f_s(c)|$ and $W = 2\Gamma$, recovering the disagreement-weighted formulas. Sampling without replacement from a finite traffic window obeys the same bound conservatively. □

B.4 PROOF OF THEOREM 2 AND COROLLARY 1

Proof of Theorem 2. All sums range over the registered set $\mathcal{A}_\cup$. For $\lambda \in \mathcal{S}_K$, set

$$F(n, \lambda) = \sum_k \lambda_k H_k(n) = \sum_c \frac{q_c^2 \sum_k \lambda_k w_{c,k}^2}{n_c}.$$

The allocation set is compact and convex, F is convex in n and affine in λ, and $\max_{\lambda \in \mathcal{S}_K} F(n, \lambda) = \max_k H_k(n)$. The minimax theorem therefore gives

$$\min_{n \in \mathcal{N}_B} \max_k H_k(n) = \max_{\lambda \in \mathcal{S}_K} \min_{n \in \mathcal{N}_B} F(n, \lambda).$$

For computation, define the concave dual

$$g(\lambda) = \min_{n \in \mathcal{N}_B} F(n, \lambda), \qquad \lambda \in \mathcal{S}_K.$$

For fixed λ, positive aggregate-coefficient cells in the inner solution obey

$$n_c(\lambda) = \mathrm{clip}_{[1, N_c]} \left(\alpha q_c \sqrt{\sum_k \lambda_k w_{c,k}^2} \right),$$

where α is the budget multiplier solving $\sum_c n_c(\lambda) = B$; the left-hand side is monotone in α, so the multiplier can be obtained by one-dimensional bisection. The outer numerical problem is therefore only K-dimensional. Zero aggregate coefficients are objective-neutral and receive residual budget only after positive-coefficient cells saturate; the resulting continuous allocation is rounded by the registered floor-and-fill rule.

At a saddle point (n^*, λ^*), the KKT conditions for the inner allocation give the displayed clipped form; without active bounds, n_c^* is proportional to $q_c \{\sum_k \lambda_k^* w_{c,k}^2\}^{1/2}$. A cell with zero aggregate coefficient is objective-neutral; after positive-coefficient cells reach their upper bounds, any required residual budget is assigned among these cells. At a dual optimum, choose this assignment to attain the primal minimax value; arbitrary choices among nonunique inner minimizers need not form a saddle point.

For floor-and-fill rounding, $\widetilde{n}_c \geq \lfloor n_c^* \rfloor \geq n_c^*/2$ because $n_c^* \geq 1$. Therefore, for every candidate,

$$H_k(\widetilde{n}) \leq 2 H_k(n^*),$$

which proves the $\sqrt{2}$ worst-radius factor. Theorem 1 applies to every fixed feasible integer allocation, so rounding changes efficiency but not certification validity.

For the envelope objective $E(n) = \sum_c q_c^2 (w_c^{\max})^2 / n_c$, pointwise

$$\max_k H_k(n) \leq E(n) \leq \sum_k H_k(n) \leq K \max_k H_k(n).$$

If n^{env} minimizes E and n^* minimizes $\max_k H_k$, then $\max_k H_k(n^{\mathrm{env}}) \leq K \max_k H_k(n^*)$; taking square roots proves the $\sqrt{K}$ radius factor. The integer envelope objective is separable, and the marginal-decrease argument in the proof of Theorem 1 gives its exact capacitated greedy solution. If one k^* attains $w_c^{\max}$ in every cell, then $E(n) = H_{k^*}(n)$ and the pointwise inequalities collapse to equality, proving the stated exactness regime. □

Proof of Corollary 1. A union bound and two-sided Hoeffding concentration give the event $|\widehat{\Delta}_k - \Delta_k| \leq r_k$ for every k with probability at least $1 - \delta$. On this event, $L_k \geq \Delta_k - 2r_k$ and $\Delta_k \geq L_k$. If the gate deploys,

$$\Delta_{\mathrm{dep}} \geq L_{\widehat{k}} = \max_k L_k \geq \max_k (\Delta_k - 2r_k),$$

and the positive lower bound also gives $\Delta_{\mathrm{dep}} > 0$. If the gate reuses, $\Delta_{\mathrm{dep}} = 0$ and $L_k \leq 0$ for every k; combining this with $L_k \geq \Delta_k - 2r_k$ gives $\Delta_k \leq 2r_k$. □

B.5 Loss-range extensions

For a normalized continuous verifier score $Y \in [0, 1]$, scalar squared loss has exact paired-loss width $2|a_c|$ for $a_c = f_r(c) - f_s(c)$. For unnormalized multiclass Brier loss and probability vectors p_s, p_r, let $a = p_r - p_s$; then

$$w_c = 2\{\max_j a_j - \min_j a_j\}.$$

Division of the loss by two divides the width by two. Normalized ordinal absolute loss satisfies $w_c \leq 2|f_r(c) - f_s(c)|$. For binary log loss with both maps clipped to $[\varepsilon, 1 - \varepsilon]$,

$$w_c = \left| \mathrm{logit}\, f_r(c) - \mathrm{logit}\, f_s(c) \right|.$$

Substitution of these registered widths into Theorem 1 gives the corresponding certificate.

B.6 Proof of Proposition 2

Proof. Set $f_s(c) = \{1 - a_c\}/2$ and $f_r(c) = \{1 + a_c\}/2$. Under the null, let $Y \mid C = c \sim \text{Bernoulli}(1/2)$, giving zero repair gain. Under the alternative, let

$$Y \mid C = c \sim \text{Bernoulli}\left(\frac{1}{2} + \text{sign}(a_c)\frac{\Delta}{2\Gamma}\right)$$

on active cells. The target masses and unlabeled observations agree under the two distributions, while the alternative repair gain is Δ. Each queried label contributes at most $\text{kl}(1/2+\Delta/(2\Gamma) \,\|\, 1/2)$ to transcript relative entropy, including under adaptive state selection. Data processing through the deploy event gives

$$B\,\text{kl}\left(\frac{1}{2} + \frac{\Delta}{2\Gamma} \,\middle\|\, \frac{1}{2}\right) \geq \text{kl}(1 - \beta \,\|\, \delta).$$

For $\Delta/(2\Gamma) \leq 1/4$, the Bernoulli relative entropy is at most Δ^2/Γ^2 by the corresponding χ^2 divergence. Rearrangement proves the stronger bound

$$B \geq \frac{\Gamma^2}{\Delta^2}\,\text{kl}(1 - \beta \,\|\, \delta),$$

which implies the displayed rate for fixed $\beta < 1$. □

B.7 Future-Population Certificate without a Global Mass Bound

Proposition 5 (Split-mass future-population certificate)**.** *Fix K fitted repairs and their label-allocation design before observing certification outcomes. Candidate k agrees with source reuse outside active states $\mathcal{A}_k$. Let $\mu_{c,k} = \mathbb{E}\{D_{c,k}(Y) \mid C = c\}$, $D_{c,k}(Y) \in [l_{c,k}, u_{c,k}]$, and $w_{c,k} = u_{c,k} - l_{c,k}$. For $\delta_y, \delta_q \in (0,1)$, write $N = \sum_k |\mathcal{A}_k| \geq 1$ and, from $n_c \geq 1$ independent certification labels in each active state, let $\widehat{\mu}_{c,k} = n_c^{-1} \sum_{i=1}^{n_c} D_{c,k}(Y_{ci})$ and define*

$$L_{c,k} = \max\left\{ l_{c,k},\ \widehat{\mu}_{c,k} - w_{c,k}\sqrt{\frac{\log(N/\delta_y)}{2n_c}} \right\}.$$

Set $L_k(c) = L_{c,k}$ on $\mathcal{A}_k$ and zero elsewhere. Let $C_1^q, \ldots, C_m^q \overset{\text{iid}}{\sim} q$ be state-only observations independent of all data used to fit repairs, choose support and allocation, and form the certification-label bounds. With $v_k = \max_c L_k(c) - \min_c L_k(c)$, define

$$\underline{\Delta}_k^{\text{pop}} = \frac{1}{m}\sum_{j=1}^{m} L_k(C_j^q) - v_k\sqrt{\frac{\log(K/\delta_q)}{2m}}.$$

Then, with probability at least $1 - \delta_y - \delta_q$, every candidate simultaneously satisfies $\Delta_k \geq \underline{\Delta}_k^{\text{pop}}$. Deploying the candidate with the largest positive bound therefore controls harmful deployment at level $\delta_y + \delta_q$ without knowing q or bounding $\|\widehat{q} - q\|_1$.

Proof. One-sided Hoeffding concentration and a union bound over the N active candidate–state pairs give $\mu_{c,k} \geq L_{c,k}$ simultaneously with probability at least $1 - \delta_y$. On this event, $\Delta_k = \sum_c q_c \mu_{c,k} \geq \mathbb{E}_q L_k(C)$. Conditional on all fitted maps, label data, and resulting functions L_k, the held-out states remain iid from q. A second one-sided Hoeffding bound and a union bound over K candidates give

$$\mathbb{E}_q L_k(C) \geq \frac{1}{m}\sum_{j=1}^{m} L_k(C_j^q) - v_k\sqrt{\frac{\log(K/\delta_q)}{2m}}$$

simultaneously with probability at least $1 - \delta_q$. Intersecting the two events proves the claim. □

B.8 Variance-Adaptive Disagreement Certificate

Proposition 6 (Variance-adaptive disagreement certificate)**.** *Fix K repairs and the shared disagreement-envelope allocation before opening independent target-conditional certification labels, with $n_c \geq 2$ in each retained active state. Candidates with empty active sets have zero gain and are omitted below. Let $\widehat{V}_{c,k}$ be the sample variance of $D_{c,k}$ and define*

$$b_{c,k} = \sqrt{\frac{2\widehat{V}_{c,k}\Lambda_k}{n_c}} + \frac{7w_{c,k}\Lambda_k}{3(n_c-1)}, \qquad \Lambda_k = \log\frac{2K|\mathcal{A}_k|}{\delta},$$

where $\mathcal{A}_k$ is candidate k's active-state set. Then, with probability at least $1-\delta$, every candidate simultaneously satisfies

$$\Delta_k \geq \widehat{\Delta}_k - \sum_{c\in\mathcal{A}_k} q_c b_{c,k}.$$

Selecting the candidate with the largest positive lower bound preserves the no-harm guarantee after candidate selection.

Proof. Condition on the fitted maps, target states, retained support, and allocation. For a fixed candidate and state, apply the empirical-Bernstein inequality of Maurer & Pontil (2009) to the reflected, range-normalized variable $\{u_{c,k} - D_{c,k}\}/w_{c,k}$ to obtain

$$\widehat{\mu}_{c,k} - \mu_{c,k} \leq b_{c,k}$$

with failure probability at most $\delta/(K|\mathcal{A}_k|)$. A union bound over every active state of every candidate makes all such events simultaneous. Multiplying the statewise lower bounds by q_c, summing over states, and then selecting any candidate with a positive resulting bound proves the claim. Candidate fitting and allocation remain independent of certification labels by construction, so no additional selection term is required. □

B.9 Multiple candidates and estimated target masses

For K candidates, apply Theorem 1 at level δ/K and take a union bound. In the general bounded-loss setting, use $w_c^{\max} = \max_k w_{c,k}$ and allocate in proportion to $q_c w_c^{\max}$. For binary Brier loss, $w_c^{\max} = 2\max_k |f_{r,k}(c) - f_s(c)|$, which recovers the disagreement envelope. Finally, if $|E\{D_{c,k}(Y) \mid C = c\}| \leq M$ for every state and candidate, replacing q with $\widehat{q}$ changes each gain by at most $M\|\widehat{q} - q\|_1$. Subtracting a valid upper bound $M\epsilon_q$ from each lower confidence bound therefore preserves simultaneous no-harm control.

C Additional Experimental Results

This appendix collects detailed tables that support the main experimental claims while keeping the paper body focused on the central deployment story.

C.1 Simulation and Scalar-Reward Controls

The Monte Carlo experiment isolates the finite-sample safety and power result from repair fitting. The scalar-reward experiment tests whether removing a known length cue eliminates the need for target recalibration.

Table 5: Monte Carlo check of Theorem 1 over 50,000 repetitions. The beneficial gain is 0.0528. Boundary columns give deployment rates under disagreement allocation; the exact null and nearby ± 0.005 alternatives directly probe the switching boundary. Coverage is at least 99.4% over all scenarios, budgets, and methods.

Budget	Radius		Switch, $\Delta = 0.0528$		Proposed deploy rate		
	Uniform	Proposed	Uniform	Proposed	$\Delta = 0.005$	$\Delta = 0$	$\Delta = -0.005$
25	0.0961	0.0789	0.105	0.197	0.0041	0.0022	0.0011
50	0.0697	0.0557	0.257	0.429	0.0034	0.0017	0.0007
100	0.0492	0.0394	0.578	0.835	0.0048	0.0015	0.0004
200	0.0350	0.0278	0.916	0.994	0.0070	0.0015	0.0001

Table 6: Full RM-Bench scalar-reward comparison. Source and target-refit risks use grouped five-fold evaluation with reward-margin states. The fixed length penalty and target-traffic LOWESS use no target labels. Intervals are 95% prompt-cluster bootstrap intervals for the refit gain.

Reward signal	Source risk	Target refit	Gain	95% CI
Raw Skywork Reward V2	0.1074	0.0992	0.0082	[0.0049, 0.0115]
Fixed length penalty	0.0926	0.0864	0.0062	[0.0034, 0.0091]
Target-unlabeled LOWESS	0.1786	0.1489	0.0297	[0.0230, 0.0363]

C.2 Exposure, Prompt Protocol, and Deployment Diagnostics

Table 7 records what public release chronology and model documentation can establish. A post-release benchmark designation only excludes direct use of the released items by that public checkpoint; it does not rule out overlap with upstream prompts or similar examples. The DeepSeek endpoint is date-stamped because its API alias is mutable. Table 8 separates the canonical pairwise protocol, the protocol-locked recent-model replication, and native specialist interfaces. The canonical and recent-model replication tracks share the same source/target rubric and grouped evaluation protocol; native-interface scores are calibrated and analyzed separately.

Table 7: **Exposure audit for completed central experiments.** "Post" means that the public checkpoint predates the benchmark release; it excludes direct exposure to released items, while allowing upstream-data overlap. "Possible" means chronology or data disclosure cannot exclude exposure. "Reported" means the model source explicitly reports benchmark evaluation.

Judge	RM-Bench	JETTS	RealDR
Llama-3.1-8B	Post	Post	Post
Qwen2.5-7B	Post	Post	Post
Prometheus-7B-v2	Post	Post	Post
Qwen3-4B	Possible	Possible	Post
Qwen3.8-27B	Possible	Possible	–
Skywork Reward V2	Reported/possible	Possible	Post
DeepSeek-V4-Flash API	Unknown/possible	Unknown/possible	Post/reported

Table 8: **Prompt and interface protocol.** Source and target use the same judge prompt within every experiment. Native-interface results are reported as replications and are never pooled with raw canonical scores.

Track	Judges	Interface
Canonical pairwise	Qwen3-4B, Qwen2.5-7B, Llama-3.1-8B, DeepSeek-V4-Flash	Shared rubric; JSON A/B/tie; temperature 0
Protocol-locked recent-model replication	Qwen3.8-27B	Canonical shared rubric; JSON A/B/tie; temperature 0; non-thinking
Scalar native	Skywork Reward V2	Official chat template; no system prompt; scalar reward
Specialist native	Prometheus-7B-v2	Official rubric/reference pairwise template
RealDR native	DeepSeek-V4-Flash	Exact benchmark vanilla prompt; JSON 0–10; non-thinking

Figure 2 and Table 9 provide the corresponding deployment-replay safety and power diagnostics.

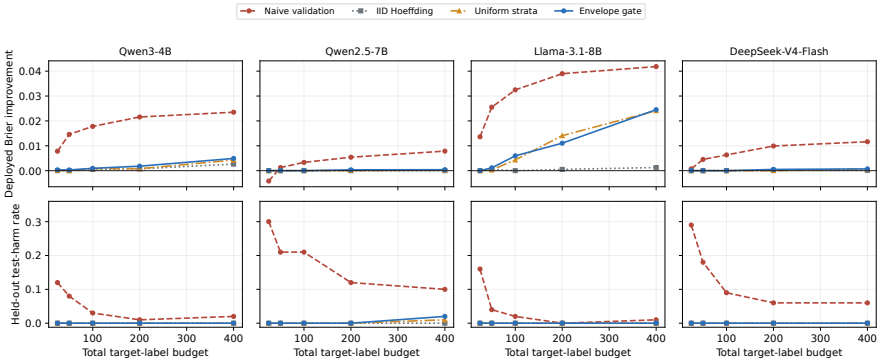

Figure 2: **Safety and power for the four canonical RM-Bench judges** over 100 prompt-disjoint replays. Top: held-out deployed Brier improvement; bottom: held-out test-harm frequency. The disagreement-envelope gate gains power for Llama while its observed held-out test harm remains at most 0.02; Table 9 reports the primary rounded-minimax counts.

Table 9: Held-out test-harm events for the rounded minimax-menu cells in Table 2. Entries give event count out of 100 replays and a 95% Wilson interval for the replay probability. They quantify finite-test behavior and are not coverage estimates for the true-gain event in Theorem 1.

B_{tot}	Judge	Naive count [95% CI]	Minimax count [95% CI]
50	Qwen3-4B	8 [0.041, 0.150]	0 [0.000, 0.037]
	Qwen2.5-7B	21 [0.142, 0.300]	0 [0.000, 0.037]
	Llama-3.1-8B	4 [0.016, 0.098]	0 [0.000, 0.037]
	DeepSeek-V4-Flash	18 [0.117, 0.267]	0 [0.000, 0.037]
400	Qwen3-4B	2 [0.006, 0.070]	0 [0.000, 0.037]
	Qwen2.5-7B	10 [0.055, 0.174]	0 [0.000, 0.037]
	Llama-3.1-8B	1 [0.002, 0.054]	0 [0.000, 0.037]
	DeepSeek-V4-Flash	6 [0.028, 0.125]	0 [0.000, 0.037]

C.3 Complete JETTS Equivalence Analysis

Complete JETTS inference uses the prompts that are jointly pair-eligible in both environments: 1,229 unique prompts across MATH, HumanEval, MBPP, and IFEval. The four prespecified generator transitions contain 2,490 shift–prompt instances and 9,960 source/target and A/B-order rows. Five deterministic prompt folds keep both orders and both generator regimes together. The main verdict-only analysis retains parse failure as a judge state and uses 10,000 prompt-cluster bootstrap

samples. Practical equivalence requires the 90% interval to lie inside the frozen $[-0.01, 0.01]$ Brier-gain interval; a positive 95% lower endpoint separately marks statistically detectable staleness.

Table 10: JETTS analysis scope. Common paired prompts have at least one correct and one incorrect response in both environments. Percentages use the 1,885 prompt groups appearing anywhere in the pair-construction file; this denominator is not a complete deployment stream. Excluded columns give Math/HumanEval/MBPP/IFEval counts not in the common paired set; the available file cannot separate missing outputs from all-correct or all-incorrect exclusions.

Transition	Common	% of 1,885	Excl. Math	Excl. HE	Excl. MBPP	Excl. IFEval
Llama 8B→70B	726	38.5	721	61	151	226
Qwen 32B→72B	639	33.9	653	93	224	276
Llama 8B→Qwen 32B	575	30.5	763	86	214	247
Llama 70B→Qwen 72B	550	29.2	745	91	214	285

Prompt-level macro gains range from approximately zero to 0.00066, and every judge is practically equivalent. Table 11 gives all 20 judge–transition cells from the canonical panel and recent-model replication. Qwen2.5 on the Llama-70B to Qwen-72B transition is additionally detectably stale, with gain 0.00175 and a positive 95% lower endpoint, while remaining inside the practical margin. The canonical outputs have zero parse failures for Qwen3, Qwen2.5, and DeepSeek and four for Llama; the Qwen3.8 replication has zero parse failures.

Table 11: **Complete JETTS fold inference for the canonical panel and protocol-locked recent-model replication.** Gain is source-transfer Brier risk minus target-refit risk; decisions use the frozen 0.01 margin.

Judge	Shift	Groups	Gain	95% CI	90% CI	Decision
DeepSeek-V4-Flash	Llama-70B to Qwen-72B	550	0.0010	[-0.0008, 0.0029]	[-0.0006, 0.0026]	practically equivalent
DeepSeek-V4-Flash	Llama-8B to Qwen-32B	575	-0.0001	[-0.0010, 0.0008]	[-0.0009, 0.0006]	practically equivalent
DeepSeek-V4-Flash	Llama 8B to 70B	726	0.0005	[-0.0011, 0.0022]	[-0.0009, 0.0019]	practically equivalent
DeepSeek-V4-Flash	Qwen 32B to 72B	639	0.0010	[-0.0012, 0.0034]	[-0.0009, 0.0030]	practically equivalent
Llama-3.1-8B	Llama-70B to Qwen-72B	550	-0.0008	[-0.0018, 0.0002]	[-0.0016, 0.0000]	practically equivalent
Llama-3.1-8B	Llama-8B to Qwen-32B	575	-0.0002	[-0.0007, 0.0003]	[-0.0006, 0.0003]	practically equivalent
Llama-3.1-8B	Llama 8B to 70B	726	0.0007	[-0.0010, 0.0024]	[-0.0007, 0.0021]	practically equivalent
Llama-3.1-8B	Qwen 32B to 72B	639	-0.0005	[-0.0010, -0.0000]	[-0.0009, -0.0001]	practically equivalent
Qwen2.5-7B	Llama-70B to Qwen-72B	550	0.0018	[0.0001, 0.0035]	[0.0004, 0.0032]	detectable but equivalent
Qwen2.5-7B	Llama-8B to Qwen-32B	575	0.0001	[-0.0005, 0.0007]	[-0.0004, 0.0006]	practically equivalent
Qwen2.5-7B	Llama 8B to 70B	726	0.0000	[-0.0016, 0.0017]	[-0.0013, 0.0014]	practically equivalent
Qwen2.5-7B	Qwen 32B to 72B	639	0.0001	[-0.0011, 0.0013]	[-0.0009, 0.0011]	practically equivalent
Qwen3-4B	Llama-70B to Qwen-72B	550	0.0003	[-0.0009, 0.0016]	[-0.0007, 0.0014]	practically equivalent
Qwen3-4B	Llama-8B to Qwen-32B	575	-0.0007	[-0.0019, 0.0005]	[-0.0017, 0.0003]	practically equivalent
Qwen3-4B	Llama 8B to 70B	726	-0.0002	[-0.0010, 0.0005]	[-0.0008, 0.0004]	practically equivalent
Qwen3-4B	Qwen 32B to 72B	639	0.0002	[-0.0007, 0.0012]	[-0.0005, 0.0010]	practically equivalent
Qwen3.8-27B	Llama-70B to Qwen-72B	550	-0.00026	[-0.00138, 0.00085]	[-0.00121, 0.00069]	practically equivalent
Qwen3.8-27B	Llama-8B to Qwen-32B	575	0.00018	[-0.00091, 0.00145]	[-0.00076, 0.00124]	practically equivalent
Qwen3.8-27B	Llama 8B to 70B	726	-0.00023	[-0.00141, 0.00099]	[-0.00120, 0.00078]	practically equivalent
Qwen3.8-27B	Qwen 32B to 72B	639	0.00098	[-0.00020, 0.00227]	[-0.00004, 0.00202]	practically equivalent

Table 12: JETTS practical-equivalence margin sensitivity for the canonical panel and protocol-locked recent-model replication. Intervals and point estimates are unchanged; only the equivalence margin is varied from the frozen primary value of 0.01.

Margin	Equivalent folds	Equivalent macros	Detectable and equiv.	Inconclusive folds
0.005	20/20	5/5	1	0
0.010	20/20	5/5	1	0
0.020	20/20	5/5	1	0

Protocol-locked recent-model replication. The prompt, folds, estimand, and decision thresholds were fixed before collecting Qwen3.8-27B outputs. The official checkpoint (ModelScope revision `1098534ab5d7`) uses `enable_thinking=False`; the full software and decoding manifest is retained in the artifact. Temperature 0 follows the locked protocol rather than the model-card sampling recommendation. RM-Bench gain is 0.00589 [0.00325, 0.00863]. Table 11 reports the four JETTS intervals; macro gain is -0.00002 $[-0.00072, 0.00073]$. Release chronology supports recent-model robustness, with no benchmark-exposure control.

Through $B_{\text{tot}} = 400$ the gate reuses source; the 0.0059 OOF gain is below certification power. Naive validation switches in 64–86% of replays and incurs 12–21% held-out test harm.

Table 13: Protocol-locked Qwen3.8-27B replay over 100 RM-Bench repetitions. Certified uses envelope allocation on the preassigned certification window; harm and gain use the untouched test set.

B_{tot}	Naive switch	Naive harm	Cert. switch	Cert. harm	Cert. gain
50	0.64	0.21	0.00	0.00	0.0000
200	0.74	0.17	0.00	0.00	0.0000
400	0.86	0.12	0.00	0.00	0.0000

Prometheus-7B-v2 uses its native feedback template and fails structured parsing on 1,171 of 9,960 rows (11.76%). Retaining parse failure as a judge state gives macro gain 0.00059; excluding invalid rows and rebuilding common source–target prompt groups gives 0.00096. Both analyses classify every transition and the prompt macro as practically equivalent. The specialist result supports the boundary finding without entering the canonical claim; complete rows are retained in the artifact.

C.4 REALDR EXPERT-SCORE EXTERNAL VALIDATION

Table 16 reports every prespecified generation-prompt and leave-one-generator-out fold. The judge produced 640/640 valid scores using 4.36M provider-reported tokens. Its score range is compressed: 535 of 640 reports receive 7 or 8, while expert scores span 3.6–9.9. The main state uses fixed score bins; isotonic and linear sensitivity analyses preserve the positive family-level macro direction. Restricting to the 29 complete two-annotator tasks also preserves positive prompt-family and generator-family macro gains. For the prompt-update folds, the source is the detailed high-quality instruction; the three targets use weakened requirements, authoritative packaging, and the raw user query.

Each replay fixes a task-level certification window disjoint from fit and test. The pooled Hoeffding radius uses the union of all task-mean loss intervals in that window, computed without expert outcomes; task means weight tasks equally. The finite-window EB variant retains this range-radius floor. Untouched-task risk remains a separate generalization diagnostic.

Table 14: Preassigned group pools and actual fit/certification use. RM-Bench rows use one label per prompt group. RealDR entries are tasks, with four expert-scored reports per task; its maximum budget labels all 15 certification-window tasks (60 reports).

Data	Fit pool	Cert. window	Test	Smaller use	Largest use
Canonical RM-Bench	232	232	199	25/25 ($B_{\text{tot}} = 50$)	200/200 ($B_{\text{tot}} = 400$)
Qwen3.8 RM-Bench	232	232	199	25/25 ($B_{\text{tot}} = 50$)	200/200 ($B_{\text{tot}} = 400$)
RealDR	15	15	10	2/2 tasks ($B_{\text{tot}} = 4$)	15/15 tasks ($B_{\text{tot}} = 30$)

At the largest task-disjoint deployment budget, each replay uses 15 target fit tasks and labels all 15 tasks in the certification window, corresponding to 60 expert-scored reports in each split. We nevertheless retain the common uncorrected range radius rather than using the census radius zero. Hoeffding and empirical-Bernstein gates retain source reuse in all 700 shift–replays. The exact-window gate instead selects the best positive complete-window gain: it switches in 82.3%, has zero window harm by construction, and has 24.0% held-out test harm on the ten untouched tasks. Thus the conservative non-switch is a property of the retained bound, while test harm is a distinct failure to generalize beyond the certified window. Deployment on new tasks requires stability across task windows or a separate future-population certificate.

Table 15: **RealDR census-window comparison at the maximum budget.** Both rules use the same frozen repair menu and all 15 certification-window tasks in each of 700 shift–replays. The conservative rule retains the uncorrected range radius; the exact-window gate sets the radius to zero and selects the candidate with the largest positive complete-window gain. Window and test gains are mean deployed improvements over source reuse. Window harm is zero for the exact gate by construction; test harm concerns the ten untouched tasks and therefore measures generalization beyond the certified window.

Rule	Window gain	Switch	Window harm	Test gain	Test harm
Uncorrected range	0.00000	0.000	0.000	0.00000	0.000
Exact window ($r = 0$)	0.00084	0.823	0.000	0.00045	0.240

Table 16: **Expert-scored long-report external validation.** A fixed DeepSeek-V4-Flash judge scores all 640 RealDR reports with the native 0–10 pointwise prompt. Source and target maps use fixed judge-score bins and are evaluated by five-fold task-grouped cross-fitting. Gain is source-transfer risk minus target-refit risk; intervals are 95% task-cluster bootstrap CIs. For the first three rows, the source is a detailed high-quality prompt and the targets weaken requirements, add authoritative packaging, or use the raw query. All seven prespecified folds are shown.

Shift	Source	Refit	Gain [95% CI]	Rel. gain
Detailed→weakened	0.00671	0.00519	**0.00152 [0.00046,0.00262]**	22.7%
Detailed→authoritative	0.00669	0.00704	−0.00035 [−0.00135,0.00047]	−5.2%
Detailed→raw query	0.00819	0.00752	0.00067 [−0.00035,0.00165]	8.2%
Others→Gemini	0.00451	0.00388	0.00063 [−0.00002,0.00131]	13.9%
Others→Grok	0.01196	0.00998	**0.00198 [0.00054,0.00343]**	16.6%
Others→Mita	0.00573	0.00523	0.00050 [−0.00013,0.00116]	8.8%
Others→Perplexity	0.00562	0.00532	0.00030 [−0.00004,0.00066]	5.3%

C.5 Source-Transfer Mechanism Detail

Table 17 reports the public model-release check. Replacing Qwen2.5-7B solver outputs with Qwen3-4B changes both objective labels and judge verdicts. Across five local judges, held-out source-map risk falls from 0.0084 to 0.0024 on MATH500 with a budget-200 target repair and from 0.0358 to 0.0016 on HumanEval with a budget-100 repair. This release experiment is less controlled than the label-preserving presentation checks and more closely matches the model-update event that initiates an audit.

Table 17: Qwen2.5-to-Qwen3 model-release update check. The source side uses Qwen2.5-7B solver outputs and the target side uses Qwen3-4B solver outputs on the same task sets. Risks are held-out target reliability-map risks averaged over five local judges under the paired-release protocol. Target audit uses the fixed target-only stratified estimator rather than retrospective best-method selection. These are target-refit diagnostic risks, not certified deployment decisions.

Split	Items	Label flip	Judge flip	Source	Target @100	Target @200
MATH500	500	13.4%	17.0%–26.2%	0.0084	0.0068	0.0024
HumanEval	164	37.8%	7.9%–32.9%	0.0358	0.0016	–

Table 18 separates two mechanisms in the controlled MATH500 presentation shift. Verdict-only cells have covered target support, so their increased risk reflects a changed judge–label relationship within the same coarse verdict states. Style-aware cells expose the deployment reality that target answers occupy presentation states absent from source calibration, which produces much larger source-transfer diagnostic mass. In paired same-item diagnostics, the target label remains fixed while judges flip on 17.8% of items on average in both presentation directions.

Table 19 adds a controlled code correctness variant. Each HumanEval candidate is paired with a comment-enriched version whose executable statements and unit-test label are unchanged. Coarse verdict-only transfer is nearly stable, while style-aware states place all target examples in new presentation support and raise the source-transfer risk ratio to about 1.09–1.10. The paired diagnostic shows zero label flips and 5.5% to 19.8% judge-verdict flips across the five local judges, so

Table 18: Controlled MATH500 presentation-shift mechanisms. Verdict-only cells keep target support covered; style-aware cells expose the target presentation as new support. Transfer mass is the target-mass-weighted statewise contribution to the source-transfer diagnostic; Top-cell share is the mean largest-cell fraction of this quantity over judges. Judge-flip rates are paired same-item diagnostics; human labels are unchanged in both directions.

Direction	Diagnostic	Uncovered	Ratio	Transfer mass	Top cell	Judge flip
Concise→confident	verdict-only	0.000	1.323	0.034	0.807 ± 0.087	0.178 ± 0.075
Concise→confident	style-aware	1.000	1.875	0.340	0.598 ± 0.042	0.178 ± 0.075
Confident→concise	verdict-only	0.000	1.079	0.018	0.838 ± 0.066	0.178 ± 0.075
Confident→concise	style-aware	1.000	1.776	0.347	0.589 ± 0.034	0.178 ± 0.075

code-style presentation is a smaller but independent objective-domain analogue of the MATH500 presentation effect.

Table 19: Controlled HumanEval code-style presentation variant. The paired rows share the same Python candidate and unit-test label; the comment-enriched version adds non-executable comments. Judge flip reports the range over five local judges; label flip is zero by construction.

Direction	Coarse source	Coarse target	Coarse ratio	Style ratio	Judge flip
Plain code→commented code	0.219	0.220	1.003	1.096	5.5–19.8%
Commented code→plain code	0.219	0.220	1.004	1.090	5.5–19.2%

The artifact additionally reports the final-answer-versus-rationale MATH control, observable style-cell movement, and difficulty, subject, and per-judge breakdowns.

C.6 Per-Judge and Panel Robustness

Tables 20–21 document panel composition and protocol diagnostics. Table 22 summarizes per-judge robustness.

Table 20: Judge panels used by each experimental split. Panel sizes are row-specific because some benchmarks provide released judge outputs while others are run with local structured-output judges.

Split	Panel	Members
MATH500 presentation	5 local	Qwen2.5-7B, Llama-3.1-8B, Mistral-7B, Selene-8B, Prometheus-7B-v2
LLMBar natural→adv.	7 local	Qwen2.5-7B, Llama-3-8B, Mistral-7B, Prometheus-7B-v2, Gemma-3-12B, Selene-8B, DeepSeek-V4-Flash
RewardBench easy→hard	7 local	Same seven local judges as LLMBar
LMArena non-code→code	7 local	Qwen2.5-7B, Llama-3-8B, Mistral-7B, Prometheus-7B-v2, Gemma-3-12B, Selene-8B, Qwen3-4B
JudgeBench generator	5 released	GPT-4o, Claude-3.5-Sonnet, Claude-3-Haiku, Llama-3.1-70B, Llama-3.1-8B arena-hard judges
JudgeBench length/style	5 released	GPT-4o and Claude-3.5-Sonnet arena-hard judges, Compass, Skywork, vanilla GPT-4o variants

Table 22 reports the per-judge consistency check for the main repair claims. The high-drift repair rows are positive for every compatible judge, while RewardBench target-cell repair is negative for every compatible judge. Restricting LLMBar, RewardBench, and the LMArena preference subset to their common six-judge panel preserves the qualitative regimes; the full panel-matched table and per-judge rows are retained in the artifact.

Table 21: Protocol diagnostics for the output-shift experiments. Rows report the calibration contract used in the main results, source and target split sizes per judge, invalid judge-parse rate, and target mass in unsupported or sparse source cells. Every contract includes the judge verdict. Style adds relative-length bin, answer-format transition, and the recorded bias/perturbation type; task+format adds the task slice and the latter two features. Sparse mass uses source-cell count at most five; Target cells is the mean number of occupied target states over judges.

Split	Contract	Judges	Source	Target	Parse err.	Uncovered	Sparse	Target cells
MATH concise→conf.	verdict+style	5	1000	1000	0.0007	1.0000	1.0000	2.4
MATH conf.→concise	verdict+style	5	1000	1000	0.0007	1.0000	1.0000	2.4
LLMBar natural→adv.	verdict+style	7	200	638	0.0002	0.1294	0.4655	65.1
RewardBench easy→hard	verdict+style	7	1610	390	0.0000	0.0029	0.0366	22.4
LMArena non-code→code	verdict+task+style	7	1298	302	0.0037	0.5516	0.8273	36.1
JudgeBench generator	verdict+task+style	5	700	540	0.0098	0.1037	0.5085	128.2
JudgeBench length	verdict+task+format	5	661	254	0.0013	0.0935	0.3303	67.0

Table 22: **Per-judge consistency of the main budget-200 repair claims.** Positive means source risk minus repair risk is above zero for that judge. The compact table reports the number of judges with positive gains and the range of judge-level gains; full judge-level rows are released in CSV.

Split	Repair	Pos./judges	Mean	Min	Median	Max	Weakest judge
MATH concise→conf.	target isotonic	5/5	0.0310	0.0007	0.0052	0.1403	Mistral-7B
MATH conf.→concise	target isotonic	5/5	0.0139	0.0013	0.0062	0.0518	Mistral-7B
LLMBar natural→adv.	target cells	7/7	0.1844	0.1381	0.1788	0.2471	DeepSeek-V4-Flash
RewardBench easy→hard	target cells	0/7	-0.0090	-0.0104	-0.0091	-0.0075	Llama-3-8B
LMArena non-code→code	beta-cell	7/7	0.0144	0.0016	0.0123	0.0504	Prometheus-7B-v2
LLMBar natural→adv.	gated target cells	7/7	0.1795	0.1333	0.1754	0.2441	DeepSeek-V4-Flash
RewardBench easy→hard	gated Platt+cell	0/7	-0.0001	-0.0005	-0.0001	0.0000	Llama-3-8B
LMArena non-code→code	gated Platt+cell	6/7	0.0084	-0.0020	0.0031	0.0373	Llama-3-8B
JudgeBench generator	target beta	5/5	0.0147	0.0088	0.0145	0.0203	GPT-4o
JudgeBench length	pooled	5/5	0.0093	0.0039	0.0059	0.0170	Claude-3.5

C.7 Decision Baselines and Sensitivity

Tables 23 and 24 vary the two design choices with the largest effect on power. At $B_{\text{tot}} = 400$, allocating 75% of labels to certification raises aggregate switch rate from 0.068 to 0.347 relative to a 25% certification share, with held-out test-harm rates at most 0.005. Style-aware contracts expose the controlled RM-Bench mechanism: at $B_{\text{tot}} = 50$ the support rule resets 28.9–59.6 active cells on average, while at $B_{\text{tot}} = 400$ the enriched contracts switch in every replay and gain 0.105–0.114 Brier risk. These are distinct registered contracts rather than target-label-selected representations. The style contract uses response-length ratio and formatting features, not the benchmark's preferred/rejected style-level labels. Length ratios use cutoffs 0.8 and 1.25; the perturbation field is constant in RM-Bench. Task–style interactions additionally use the task category. These controlled results do not establish the same gain on naturally updated outputs. The verdict-only certificate remains an aggregate-risk decision with no within-style claim.

Table 25 compares the primary rounded minimax design with the exactly integer-optimized envelope surrogate. Minimax weakly lowers the mean worst-candidate radius at each tested budget, but the gap is small: at $B_{\text{tot}} = 200$ switching rises from 0.110 to 0.120 and gain from 0.0034 to 0.0040. At $B_{\text{tot}} = 400$ the envelope has slightly higher switch and gain, with two held-out test-harm replays out of 400 versus none for minimax. Neither has an observed certification-window harm event. These results show little allocation advantage for this menu, not a general advantage over the envelope.

The controlled overlap sweep in Table 26 isolates the regime hidden by this menu. As candidate movements shift from shared toward candidate-private states, minimax can lower the worst radius; the envelope needs up to $1.59\times$ as many labels to match it. Specifically, for candidate k with scale $s_k \in \{.40, .26, .20, .16, .12, .08\}$ and floor $\varepsilon_0 = .02$, the constructed range width is $w_{c,k}(\kappa) = s_k(\kappa + \varepsilon_0)$ on six shared states, $s_k(1 - \kappa + \varepsilon_0)$ on candidate k's six private states, and zero on the other private states; all 42 states have mass $1/42$ and capacity 10,000. For each κ, "match labels" is the smallest integer envelope budget, found by exact greedy marginal allocation, whose worst-candidate radius does not exceed the rounded-minimax radius at $B = 420$. The resulting budget ratio need not vary monotonically with κ.

Table 23: Fit/certification split sensitivity: switch / held-out test-harm rates over 400 judge–replays using envelope allocation. Changing the fit fraction changes the preassigned pool sizes as well as label allocation; test groups stay fixed within each seed.

Fit/certification	$B_{\text{tot}} = 50$	$B_{\text{tot}} = 200$	$B_{\text{tot}} = 400$
25/75	0.025 / 0.000	0.223 / 0.000	0.347 / 0.005
50/50	0.018 / 0.000	0.110 / 0.000	0.230 / 0.005
75/25	0.007 / 0.000	0.030 / 0.000	0.068 / 0.000

Table 24: Registered calibration-contract sensitivity. Removed is the mean number of active cells reset to source reuse by the support guardrail at $B_{\text{tot}} = 50$; gain is the deployed Brier improvement at $B_{\text{tot}} = 400$. Switch and held-out test harm are fractions over 400 judge–replays. All rows use the disagreement-envelope allocation.

Contract	Source risk	$B_{\text{tot}} = 50$ switch / removed	$B_{\text{tot}} = 200$ switch	$B_{\text{tot}} = 400$ gain	$B_{\text{tot}} = 400$ switch / test harm
Verdict only	0.2633	0.018 / 0.0	0.110	0.0076	0.230 / 0.005
+ task	0.2351	0.007 / 0.0	0.210	0.0150	0.458 / 0.005
+ output style	0.2221	0.000 / 28.9	0.980	0.1142	1.000 / 0.000
+ task–style interactions	0.2390	0.013 / 59.6	0.882	0.1051	1.000 / 0.000

Table 25: Envelope versus rounded minimax-menu allocation on canonical RM-Bench. Each row aggregates four judges and 100 prompt-disjoint replays per judge. Worst radius is the mean of $\max_k r_k$ before certification labels are opened; gain and test harm use untouched groups.

B_{tot}	Allocation	Worst radius	Gain	Switch	Test harm
50	Envelope	0.1117	0.0004	0.018	0.000
50	Minimax menu	0.1114	0.0006	0.028	0.000
200	Envelope	0.0473	0.0034	0.110	0.000
200	Minimax menu	0.0473	0.0040	0.120	0.000
400	Envelope	0.0351	0.0076	0.230	0.005
400	Minimax menu	0.0351	0.0074	0.217	0.000

Table 26: **Mechanism check for Theorem 2.** Six repairs, 42 equal-mass states, and $B = 420$ are fixed. The sharedness parameter κ moves range width from candidate-private states toward six common states; a 0.02 floor keeps the state set fixed. Radius uses $\delta = 0.05$ simultaneous control. Match labels are the smallest envelope budget attaining the rounded-minimax radius at $B = 420$.

Shared κ	Envelope radius	Minimax radius	Match labels	Label factor
1.00	0.0050	0.0050	420	1.00×
0.75	0.0060	0.0053	524	1.25×
0.50	0.0069	0.0056	639	1.52×
0.25	0.0076	0.0061	666	1.59×
0.00	0.0084	0.0068	633	1.51×

Table 27 compares the range certificate with conservative empirical-Bernstein (EB) variants. Proposition 6 assumes independent conditional labels; finite-window replays instead sample without replacement and use the larger of the EB radius and the valid range radius. The pooled baseline uses the union of cellwise loss intervals, rather than the largest within-cell width. At $B_{\text{tot}} = 400$ both EB variants retain source, whereas the envelope range gate switches in 0.05–0.58 of replays. Because the range floor precludes a tighter radius, this check does not evaluate the potential efficiency of a dedicated finite-population variance-adaptive bound.

Table 28 measures sensitivity to estimated state masses. Independent state-only samples are drawn with replacement from the fixed certification window; this is not a validation on future traffic. Label and mass errors each receive 0.025 of the total 0.05 budget. The global L_1 correction never switches. Split-mass switching is 0.25%, 1.00%, and 2.25% at $m = 100$, 500, and 2,000, versus 23.0% with exact masses. Generic mass bounds therefore cost substantial power in this reference-window check; future-population validity requires the assumptions of the corresponding propositions separately.

Table 27: Variance-adaptive certification at $B_{\text{tot}} = 400$ on RM-Bench. Entries are switch / held-out test-harm rates over 100 replays. Range and stratified EB use envelope allocation; pooled EB uses uniform sampling. Finite-window EB replays retain the valid range-radius floor.

Judge	Pooled EB	Range (envelope)	Stratified EB
Qwen3-4B	0.000/0.000	0.230/0.000	0.000/0.000
Qwen2.5-7B	0.000/0.000	0.060/0.020	0.000/0.000
Llama-3.1-8B	0.000/0.000	0.580/0.000	0.000/0.000
DeepSeek-V4-Flash	0.000/0.000	0.050/0.000	0.000/0.000

Table 28: Estimated-mass sensitivity at $B_{\text{tot}} = 400$. State-only samples are drawn independently with replacement from the fixed certification window, not from newly observed future traffic. Split-mass uses m design states and m holdout states. Label and mass error each receive 0.025 of the total 0.05 error budget. Values aggregate four judges and 100 replays per judge using envelope allocation.

Mass treatment	Design m	Holdout m	Switch	Test harm	Gain
Exact finite window	0	0	0.230	0.005	0.0076
Global L_1 bound	100	0	0.000	0.000	0.0000
Split-mass certificate	100	100	0.003	0.000	0.0001
Global L_1 bound	500	0	0.000	0.000	0.0000
Split-mass certificate	500	500	0.010	0.000	0.0005
Global L_1 bound	2000	0	0.000	0.000	0.0000
Split-mass certificate	2000	2000	0.022	0.000	0.0010

Table 29 compares deployment rules under a common budget: naive validation, LTT–IID Hoeffding, pooled EB with a range floor, and the disagreement-stratified gate. Table 30 summarizes retrospective repair-family coverage across controlled and benchmark shifts; protocol-valid deployment comparisons remain in Table 29 and the main replay tables. Table 31 reports uncertainty for the retained repair claims. Complete candidate matrices, oracle diagnostics, and scalar prediction-powered/doubly robust results with their distinct aggregate estimand remain in the artifact.

Table 29: Baselines at $B_{\text{tot}} = 200$ over seven judges and 100 grouped replays per judge. Risk is deployed test Brier risk. Test harm is the unconditional frequency of selecting a repair with higher test risk than reuse. Certification targets the separate preassigned window; pooled EB uses a conservative range-radius floor.

Shift	Gate / allocation	Risk	Switch	Test harm
LLMBar natural→adversarial	Naive empirical	0.2409	1.000	0.000
	LTT–IID Hoeffding	0.2893	0.447	0.000
	Pooled EB (range floor)	0.3052	0.296	0.000
	Disagreement stratified	0.2409	1.000	0.000
RewardBench easy→hard	Naive empirical	0.0115	0.453	0.261
	LTT–IID Hoeffding	0.0115	0.000	0.000
	Pooled EB (range floor)	0.0115	0.000	0.000
	Disagreement stratified	0.0115	0.000	0.000
LMArena non-code→code	Naive empirical	0.2471	0.721	0.203
	LTT–IID Hoeffding	0.2529	0.003	0.000
	Pooled EB (range floor)	0.2529	0.000	0.000
	Disagreement stratified	0.2517	0.084	0.013

Table 30: Target risks at $B_{\text{tot}} = 200$; lower is better within each row. MATH rows report conditional-reliability map error; all other rows report held-out prediction Brier risk, including irreducible label variance. Absolute values across these metrics are not comparable. Pooled combines source and target labels cellwise. Scalar and Scalar-cell report the within-family minima over Platt/isotonic/beta and Platt+cell/beta-cell, respectively; Source-prior and Decision likewise summarize the available prior and split-specific selection families. Family minima summarize retrospective method coverage; deployment uses the prespecified certified selector. Dashes mark unavailable methods, and Oracle is analysis-only.

Split	Panel	Source	Target cells	Pooled	Scalar	Scalar-cell	Source-prior	Decision	Oracle
MATH concise→conf.	5 local	0.0334	0.0032	0.0150	0.0025	–	–	–	0.0000
MATH conf.→concise	5 local	0.0160	0.0024	0.0125	0.0020	–	–	–	0.0000
LLMBar natural→adv.	7 local	0.3417	0.1574	0.1940	0.2391	0.1795	0.1852	0.1620	0.1423
RewardBench easy→hard	7 local	0.0119	0.0209	0.0127	0.0127	0.0129	0.0120	0.0118	0.0111
LMArena non-code→code	7 local	0.2583	0.2547	0.2549	0.2513	0.2439	0.2445	0.2488	0.2389
JudgeBench generator	5 released	0.2620	0.2566	0.2587	0.2472	0.2473	0.2565	0.2573	0.2409
JudgeBench length	5 released	0.2362	0.2313	0.2269	0.2338	0.2276	0.2311	0.2313	0.2038

Table 31: Bootstrap confidence intervals for main repair claims at $B_{\text{tot}} = 200$. Gain is source risk minus repair risk, computed within judge and bootstrapped over the selected judge panel. Intervals are descriptive panel-resampling summaries, not inference over a population of judges. MATH rows use reliability-map error; other rows use prediction Brier risk. Positive gain means the repair improves over source reuse.

Split	Repair	Judges	Source	Repair	Gain	95% CI
MATH concise→conf.	target isotonic	5	0.0334	0.0025	0.0310	[0.0023, 0.0859]
MATH conf.→concise	target isotonic	5	0.0160	0.0020	0.0139	[0.0032, 0.0332]
LLMBar natural→adv.	target cells	7	0.3417	0.1574	0.1844	[0.1584, 0.2119]
RewardBench easy→hard	target cells	7	0.0119	0.0209	-0.0090	[-0.0097, -0.0083]
LMArena non-code→code	beta-cell	7	0.2583	0.2439	0.0144	[0.0054, 0.0273]
LLMBar natural→adv.	gated target cells	7	0.3414	0.1620	0.1795	[0.1527, 0.2079]
RewardBench easy→hard	gated Platt+cell	7	0.0117	0.0118	-0.0001	[-0.0003, -0.0000]
LMArena non-code→code	gated Platt+cell	7	0.2572	0.2488	0.0084	[0.0013, 0.0187]
JudgeBench generator	target beta	5	0.2620	0.2472	0.0147	[0.0109, 0.0186]
JudgeBench length	pooled	5	0.2362	0.2269	0.0093	[0.0044, 0.0143]

C.8 State-Contract Diagnostics

Table 32 varies the finite calibration state used for source-transfer diagnostics. Each row defines a different reliability contract: verdict-only states support an aggregate map, while enriched states can localize presentation effects at the cost of fragmented source support. When uncovered mass is one, the comparison is support repair against the registered global source fallback, not within-support conditional recalibration. The LMArena non-code-to-code split illustrates the resulting support regime: the registered enriched contract has 0.552 uncovered target mass, while the gated repair improves risk from 0.2572 to 0.2488 when it switches (Table 31). Full JudgeBench, per-judge, drift-cell, and exploratory allocation tables are retained in the artifact.

Table 32: Calibration-contract granularity and source support. Risk ratio is target risk divided by source risk for the transferred source map; uncovered mass is target traffic in states absent from source calibration. The enriched column is prespecified, while the final column shows the support cost of the finest explored contract.

Split	Verdict-only ratio	Verdict-only uncovered	Enriched contract	Enriched ratio	Enriched uncovered	Finest explored uncovered
MATH concise→conf.	1.323	0.000	verdict + style	1.875	1.000	1.000
MATH conf.→concise	1.079	0.000	verdict + style	1.777	1.000	1.000
LLMBar natural→adv.	2.324	$< .001$	verdict + style	2.486	0.129	1.000
RewardBench easy→hard	0.835	0.000	verdict + style	0.790	0.003	1.000
LMArena non-code→code	0.990	0.000	verdict + style	1.144	0.552	0.552
JudgeBench generator	1.116	0.000	verdict + style	1.214	0.007	0.104
JudgeBench length	0.912	0.000	task + format	1.249	0.094	1.000