LEARNING TO TEST: SEQUENTIAL MONITORING OF GENERATIVE MODELS AND AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

As AI models scale and are deployed with increasing autonomy, their opaque decision-making can pose risks to users and providers, motivating guardrails and continuous task monitoring for compliance with safety and deployment protocols. However, monitoring quality can fluctuate depending on the model and problem at hand, and generally lacks formal assurances. We recast deployment monitoring as a statistical testing problem, viewed in particular through the lens of *sequential hypothesis testing*, or 'testing by betting'. Leveraging the framework's error-control qualities, we equip monitors with verifiable false alarm controls such that benign model behavior triggers interventions only at a prespecified, tolerable rate. Challenged by the ambiguity of defining complex black-box monitoring problems in statistical terms, we investigate data-driven test designs with *learned* reference, betting, or evidence-normalization components. Empirical evaluations span diverse models and monitoring targets, including language generation, chain-of-thought reasoning, multi-turn interactions, and agentic tool use. Our results affirm that learned tests exhibit error-controlling and adaptive properties, while highlighting the nuances of translating raw monitoring scores into statistically valid evidence.

1 INTRODUCTION

Generative models such as large language models (LLMs) are increasingly deployed to solve complex tasks through extended trajectories of generation, reasoning, interaction, and tool use (Wang et al., 2024a; Chowa et al., 2026). The promise of such autonomous agents depends on reliable post-deployment behavior and adherence to specified protocols, yet continued safety research and benchmark development underscore the remaining challenges (Hammond et al., 2025; Zou et al., 2023; Cobben et al., 2026; Potamitis et al., 2025). Under continuous deployment monitoring, the resulting trajectories, or *traces*, expose intermediate steps that can be assessed by black-box scorers such as process reward models, safeguard classifiers, and LLM or agent judges (Zhang et al., 2025b; Inan et al., 2023; Shi et al., 2026). The obtained scores are informative but imperfect proxies whose quality can vary substantially across scorers, generators, and tasks (Korbak et al., 2025; Guan et al., 2025; Baker et al., 2025). Moreover, an error-indicating score does not itself specify when to intervene or how often a benign trace may be flagged. False alarms may terminate useful behavior or trigger costly fallback and review, whereas delayed alarms allow failures to propagate.

A statistical perspective on monitoring can help translate raw scores into alarm decisions with explicit error controls (Dolin et al., 2026; Rao et al., 2026). Because reasoning and agentic behavior unfold sequentially, *sequential hypothesis testing* offers a natural formulation: evidence can be inspected continuously to support a stop-or-continue decision with time-uniform false alarm control (Ramdas et al., 2023; Ramdas & Wang, 2025), as illustrated in Fig. 1. Applying this framework to generative traces, however, reveals two important nuances. First, the semantic monitoring target is complex and does not uniquely determine a statistical null, leaving the user to select score references and intervention thresholds. A practical data-driven design should reduce this burden and the reliance on expert knowledge. Second, within a trace, each stepwise score provides a *history-dependent* view of the same evolving trajectory, precluding the use of convenient *i.i.d.* or independence assumptions to establish statistically valid evidence factors. We identify this history-conditional requirement as the central technical obstacle to turning black-box scores into proper sequential testing evidence.

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.

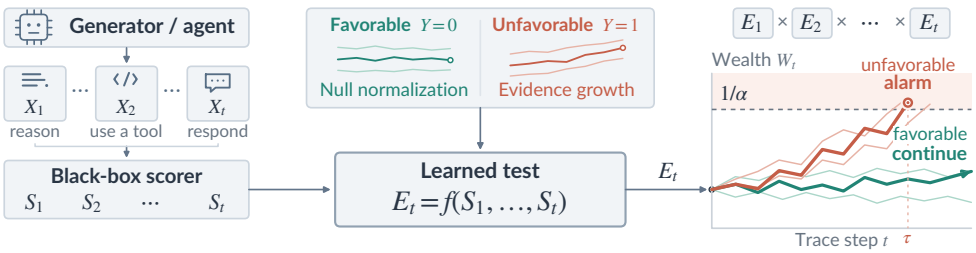


Figure 1: **Sequential monitoring with black-box scores.** A generator or agent model produces an evolving trace of states, which a black-box scorer maps to scores. Favorable and unfavorable calibration traces reflecting the monitoring target are used to learn local evidence factors E_t, promoting growth under unfavorable behavior. At deployment, these factors accumulate into wealth $W_t = E_1 \times \cdots \times E_t$ and crossing a threshold $(1/\alpha)$ triggers a monitor alarm, else the trace continues.

Close prior works such as Sadhuka et al. (2026) and Schirmer et al. (2026b) sidestep this issue by treating accumulated scores as a heuristic statistic, and instead calibrate the alarm threshold using representative holdout data. We pursue the complementary question of whether the necessary local structural constraints can be enforced, and indeed *learned*, from such a calibration set. Local evidence has reusable semantics and supports modular composition without recalibrating every fixed aggregate, while being likewise computationally efficient. We investigate both a structured conditional-mean and a more flexible two-component design, characterize the conditions under which they are structurally valid, and broadly evaluate them across diverse models and monitoring targets relative to calibrated decision rules. Our results show favorable empirical error-control and adaptive monitoring behavior, suggesting that statistical monitoring need not stop at threshold calibration but can itself *learn to test*.

In summary, our contributions include:

- Formalizing black-box statistical monitoring across completed traces and within an evolving trace, isolating history-conditional normalization as the key obstacle in the latter setting (§ 3);
- Developing structured conditional-mean and more flexible two-component test designs for learning local evidence from raw score data (§ 4);
- Comparing learned evidence to calibrated decision rules across settings spanning mathematical reasoning, generation prefixes, agentic tool use, red-teaming interactions, and chess traces (§ 5).

2 Problem Setting and Background

Monitoring objects and target. Our framework has three key objects. The *generator* is a model working through a solution or an agent carrying out a task. Given context C_i, it produces a trace of observable states or actions $X_{i,t}$. The *scorer* maps information available at step t to a bounded score $S_{i,t} \in [0,1]$, with larger values indicating less favorable assessments. The statistical *monitor* accumulates evidence from these scores to decide when to raise an alarm. Here, $i \in \{1, \ldots, n\}$ indexes traces and $t \in \{1, \ldots, T_i\}$ their within-trace steps. A terminal label $Y_i \in \{0, 1\}$ records whether the completed trace is favorable ($Y_i = 0$) or unfavorable ($Y_i = 1$) for the target property. The monitor receives only these scores, allowing arbitrary pairings of black-box generators and scorers.

The monitor's goal is to detect unfavorable target outcomes $Y_i = 1$ as early and reliably as possible, while keeping the probability of false alarms on favorable outcomes $Y_i = 0$ below a prespecified tolerance. The target may encode factual correctness, harmfulness, malicious intent, or other measurable forms of protocol compliance, so $Y_i = 1$ might denote an incorrect or harmful response. At deployment, however, neither Y_i nor any per-step ground truth is observed. Thus, the monitor must base its decisions solely on the scores $S_{i,t}$, which are imperfect, trace-dependent proxies for Y_i. To learn how score sequences relate to the target property, we only assume access to a labeled pre-deployment dataset $\mathcal{D}_{\text{cal}}$[1] of pairs $(S_{i,1:T_i}, Y_i)$ for monitor fitting and calibration.

Monitoring regimes and active horizon. We consider two distinct sequential monitoring regimes: across-trace monitoring and within-trace monitoring. In across-trace monitoring, each completed

[1]We define $\mathcal{D}_{\text{cal}}{}^{y} := \{(S_{i,1:T_i}, Y_i) \in \mathcal{D}_{\text{cal}} : Y_i = y\},\ y \in \{0,1\}$ as the label-specific subset.

trace i is assigned a single summary score S_i^{tr}, such as the mean of its stepwise scores. Monitoring then proceeds over trace indices $i = 1, \ldots, n$, treating each variable-length trace as one statistical unit. In within-trace monitoring, the monitor instead observes the evolving sequence $S_{i,1:t}$ and may stop at any step $t \leq T_i$. Since our design is data-driven, monitoring histories longer than those supported by $\mathcal{D}_{\mathrm{cal}}$ require extrapolation assumptions, which we do not consider here. We denote by T_{cal} the longest trace length supported by $\mathcal{D}_{\mathrm{cal}}$ and define the active trace horizon $H_i := \min\{T_i, T_{\mathrm{cal}}\}$. Traces longer than T_{cal} are monitored only through step T_{cal}, a limitation shared by prior work (Sadhuka et al., 2026). This is less of a practical concern if $\mathcal{D}_{\mathrm{cal}}$ adequately covers trace-length variation. Throughout, we suppress the trace index i when focusing on a single trace, but retain it when distinguishing traces or treating i as the sequential index in across-trace monitoring.

Sequential hypothesis testing. The monitor treats deployment observations as drawn from an unknown distribution P and tests the generally composite *null* hypothesis $\mathcal{H}_0 : P \in \mathcal{P}_0$ against the *alternative* $\mathcal{H}_1 : P \in \mathcal{P}_1$. The restrictions defining these families depend on the monitoring design. For arbitrary $P_0 \in \mathcal{P}_0$ and $P_1 \in \mathcal{P}_1$, we use $\mathbb{E}_0, \mathbb{P}_0, p_0$ and $\mathbb{E}_1, \mathbb{P}_1, p_1$ for the corresponding expectations, probabilities, and densities. All quantities learned from $\mathcal{D}_{\mathrm{cal}}$ are fixed before deployment and represented by the initial information $\mathcal{F}_{i,0}$. Conditional on this initial information, the monitor's *filtration* $\mathcal{F}_{i,t} := \mathcal{F}_{i,0} \vee \sigma(S_{i,1:t})$ represents all information available to the monitor through deployment step t (see § B.1 for details). Suppressing the index i for a single trace, a local evidence factor E_t, or *e-value*, and its cumulative wealth W_t, or *e-process*, satisfy for every $P_0 \in \mathcal{P}_0$ that

$$E_t \geq 0, \qquad \mathbb{E}_0[E_t \mid \mathcal{F}_{t-1}] \leq 1, \qquad \text{and} \qquad W_0 = 1, \quad W_t = \prod_{j=1}^{t} E_j. \tag{1}$$

In the betting interpretation, W_t is the bettor's wealth against $\mathcal{H}_0$ and E_t its next multiplicative return (Ramdas et al., 2023). The conditional expectation constraint makes this game fair or unfavorable under the null, preventing systematic expected wealth growth; we refer to this critical constraint as *evidence normalization*. Although E_t may depend on the current score, any bet, reference, or normalizer used to construct it must be $\mathcal{F}_{t-1}$-measurable, and hence computed *without* observing S_t.

Under Eq. 1, W_t is a nonnegative test supermartingale, and use of Ville's inequality (Ville, 1939) therefore yields a Type-I error guarantee, or false-alarm condition

$$\mathbb{P}_0(\tau < \infty) \leq \alpha \quad \text{with} \quad \tau = \inf\{t \geq 1 : W_t \geq \frac{1}{\alpha}\}, \tag{2}$$

for stopping time τ and tolerated error rate $\alpha \in (0, 1)$. Thus, crossing $1/\alpha$ triggers an alarm, and *anytime validity* allows continuous inspection and history-dependent stopping without increasing false-alarm probabilities beyond α (Ramdas & Wang, 2025). As the standard Ville threshold can be conservative, terminal randomization (RandVille) replaces it by U/α, where $U \sim \mathrm{Unif}(0, 1)$ is independent, to improve rejection probability while preserving validity (Ramdas & Manole, 2026).

Evaluation criteria. Let τ_i be the first alarm time of trace i, with $\tau_i = \infty$ if no alarm occurs. Then monitor assessment measures false alarm control through the *false positive rate* (FPR) on null traces, and detection capability through *power* on alternative traces, formally

$$\mathrm{FPR} := \mathbb{P}_0(\tau_i < \infty) = \mathbb{P}(\tau_i < \infty \mid Y_i = 0), \; \mathrm{Power} := \mathbb{P}_1(\tau_i < \infty) = \mathbb{P}(\tau_i < \infty \mid Y_i = 1). \tag{3}$$

Connecting back to our target setting, we identify null traces as favorable ($Y_i = 0$) and alternatives as unfavorable ($Y_i = 1$). When test conditions hold, Eq. 2 guarantees that $\mathrm{FPR} \leq \alpha$. FPR alone is insufficient, since a monitor that never alarms trivially satisfies the bound, hence power captures its corresponding ability to detect unfavorable traces. In practice, finite-horizon evaluation replaces the event in Eq. 3 by $\{\tau_i \leq H_i\} \subseteq \{\tau_i < \infty\}$, a subset retaining the same null normalization constraint. To measure detection speed, we can define the censored runtime $R_i := \min\{\tau_i, H_i\}$ and also measure the *average run length* $\mathrm{ARL} := \mathbb{E}_1[R_i]$ and its normalized counterpart $\mathrm{nARL} := \mathbb{E}_1[R_i/H_i]$. Missed alternatives contribute their full active horizon, and lower values indicate preferred earlier detection when interpreted jointly with power and false alarms.

3 From Scores to Sequential Evidence

We next contrast the easier setting of across-trace monitoring with the more challenging within-trace monitoring problem, and explain how prior work addresses this issue through threshold calibration.

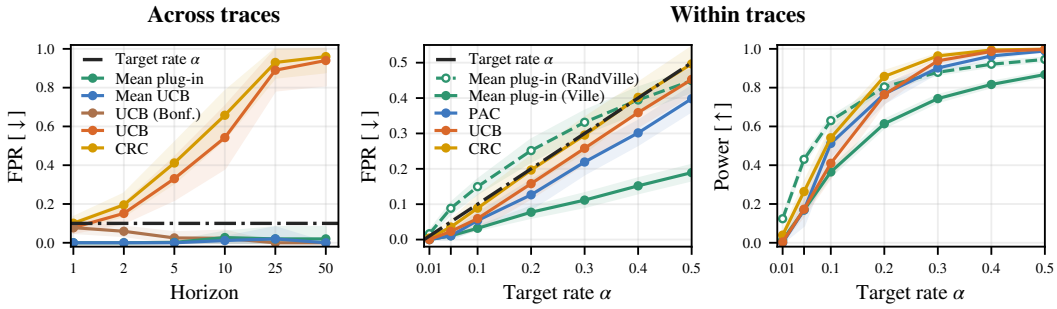


Figure 2: **Across- and within-trace monitoring.** *Left:* Across traces, mean tests using marginal plug-in or upper-confidence estimates stably maintain FPR below the target $\alpha = 0.1$, while per-trace risk control (CRC/UCB) accumulates errors as the task horizon grows. *Center, Right:* Within traces, a marginal reference violates the FPR target under thresholding (RandVille), exposing unaddressed history-dependent null drift. Calibrated thresholds remain below target. Results use MATH traces generated by Mistral-7B and scored by Qwen2.5-PRM, curves show means ± 1 SD across ten trials.

Across-trace monitoring. In this setting the monitor operates over a stream of completed traces, each summarized by a score $S_i^{\mathrm{tr}} \in [0, 1]$, such as its trace mean. Let $\mathcal{F}_i$ contain the fitted monitor and the first i trace scores. If a given reference null mean $\mu_0 \in (0, 1]$ satisfies $\mathbb{E}_0[S_i^{\mathrm{tr}} \mid \mathcal{F}_{i-1}] \leq \mu_0$ for every i, then any predictable bet $\lambda_i \in [0, 1/\mu_0]$ yields a nonnegative factor $E_i^{\mathrm{tr}} = 1 + \lambda_i(S_i^{\mathrm{tr}} - \mu_0)$ with $\mathbb{E}_0[E_i^{\mathrm{tr}} \mid \mathcal{F}_{i-1}] \leq 1$. Hence, the wealth $W_i^{\mathrm{tr}} = \prod_{j=1}^{i} E_j^{\mathrm{tr}}$ forms an e-process by Eq. 1 (Waudby-Smith & Ramdas, 2024). If traces are independently sampled from a common null population, a valid assumption for separately evaluated problems or tasks, the conditional mean reduces to the marginal mean $\mathbb{E}_0[S_i^{\mathrm{tr}} \mid \mathcal{F}_{i-1}] = \mathbb{E}_0[S_i^{\mathrm{tr}}]$, so a marginal upper bounding reference suffices. This makes a data-driven construction comparatively straightforward: the marginal calibration mean $\hat{\mu}_0 = |\mathcal{D}_{\mathrm{cal}}^0|^{-1} \sum_{i \in \mathcal{D}_{\mathrm{cal}}^0} S_i^{\mathrm{tr}}$ provides a natural empirical plug-in reference, while an estimated upper confidence bound can yield a valid e-process on its stated coverage event (see § B.2). Despite their simplicity, bounded-mean tests can achieve strong power (Clerico, 2025; Waudby-Smith et al., 2025). Furthermore, the monitor's anytime property is particularly attractive here. As illustrated in Fig. 2, a sequential test controls false alarms even as the stream of tasks grows without a predetermined horizon. In contrast, threshold calibration following Schirmer et al. (2026b) provides only atomic, or per-trace, risk control. False alarms therefore accumulate, causing the stream-level FPR to grow unless recalibration occurs whenever the horizon increases, or conservative multiple-testing corrections such as Bonferroni are applied, at a loss in power (Sedgwick, 2012).

Conditionality in within-trace monitoring. Within a trace, the marginal simplification no longer applies. A candidate mean factor $E_t = 1 + \lambda_t(S_t - \hat{\mu}_{0,t})$ requires its predictable reference to upper-bound the *history-conditional* null mean $\mathbb{E}_0[S_t \mid \mathcal{F}_{t-1}] \leq \hat{\mu}_{0,t}$ for every t. More generally, every local evidence factor must be conditionally normalized. Stepwise marginal calibration is insufficient as $\mathbb{E}_0[S_t] \leq \hat{\mu}_{0,t} \not\Rightarrow \mathbb{E}_0[S_t \mid \mathcal{F}_{t-1}] \leq \hat{\mu}_{0,t}$, nor is independence plausible: the scorer observes the preceding trace when assigning S_t, while each reasoning or action step shapes those that follow. A reference may therefore be conservative on average yet underestimate null scores after particular histories, producing insufficient normalization and positive null drift in W_t (a simple example is given in § B.3). History-adaptive bets can concentrate wagers on precisely these histories, increasing alternative growth *and* false alarms together, as shown in Fig. 2. Reformulating the hypothesis alone does not resolve this problem (see App. A). Unlike settings with a-priori known null properties, our latent target property is observed only through complex, imperfect scores, and provides no canonical null behavior from which to derive a uniquely appropriate test. Data-driven designs must therefore either *learn* an appropriate history-conditional normalizer, or shift the calibration burden elsewhere.

Calibrating alarm thresholds. Risk-control and *probably approximately correct* (PAC) baselines take the latter route by calibrating the alarm rule directly (Sadhuka et al., 2026; Schirmer et al., 2026b). Fix a stepwise alarm statistic A_t (in the simplest case, $A_t = S_t$) and define the trace-level alarm event through $M_i = \max_{t \leq H_i} A_t$. The null set $\mathcal{D}_{\mathrm{cal}}^0$ can then calibrate a cutoff $\hat{\tau}$ against the distribution of M_i, avoiding any requirement for stepwise normalization. Schirmer et al. (2026b) study *conformal risk control* (CRC; Angelopoulos et al. (2024)) and risk upper confidence bounds

(UCB), which provide in-expectation or high-probability control on the false alarm risk. Similarly, Sadhuka et al. (2026) learn A_t as a likelihood-ratio statistic and PAC-calibrate its maximum to obtain a high-probability guarantee; see § B.4 for details. Threshold calibration is convenient as it can absorb approximation error and temporal dependence directly into the alarm event, and is a valid solution when the scorer, statistic, horizon, and intervention policy are fixed before calibration.

Calibrating sequential evidence. Why then pursue the harder structural route? Threshold calibration permits an arbitrary statistic A_t, but its guarantee is tied to the calibrated event, and changing the evidence aggregate or monitoring horizon generally requires recalibration. Evidence that is structurally valid under a common deployment filtration instead permits flexible composition and adaptation, potentially improving power without recalibrating each alarm rule; Fig. 4 provides a motivating example. Accumulated wealth also has an evidential interpretation at every time and remains valid under optional stopping, rather than acquiring meaning only at a calibrated cutoff. Conceptually, validity and power can be separated during calibration: favorable traces permit null normalization, while unfavorable traces optimize for evidence growth. We highlight that these benefits formally require evidence normalization to hold for the *population-level*, history-conditional null distribution, a property that finite calibration histories cannot certify without additional assumptions. We state the required assumptions for structural validity explicitly in Prop. 1, and treat learned factors on real data as empirical approximations, not certified e-processes. Our claim is not that evidence calibration dominates threshold calibration, but that it targets a richer and potentially more reusable object.

4 Learning Sequential Tests

Learning sequential evidence poses a constrained optimization problem as we seek wealth growth without violating local null constraints. At a fixed horizon t, the target can be written abstractly as

$$\max_{E_1,\dots,E_t} \mathbb{E}_1[\log W_t] \quad \text{s.t.} \quad E_j \geq 0, \;\; \mathbb{E}_0[E_j \mid \mathcal{F}_{j-1}] \leq 1 \quad \text{for all } j \leq t \text{ and } P_0 \in \mathcal{P}_0. \tag{4}$$

Following growth-rate optimality principles (Waudby-Smith & Ramdas, 2024; Grünwald et al., 2024), we use the expected log-wealth as a power-oriented objective, but on its own apparent growth can exploit positive null drift. Unlike related learned-test settings where known null symmetries yield normalization by construction (Pandeva et al., 2024; Richter et al., 2025), our conditional normalizer must itself be inferred from $\mathcal{D}_{\text{cal}}$. We pursue two data-driven parametrizations of Eq. 4, detailed next.

4.1 Learning a Conditional Mean Test

Our first design retains the bounded-mean structure from § 3 but instantiates both its betting rate λ_θ and null reference mean $\hat{\mu}_\phi$ as learnable components with parameters θ, ϕ, that is

$$E_t = 1 + \lambda_t(S_t - \hat{\mu}_{0,t}) \quad \text{with} \quad \lambda_t = \lambda_\theta(h_{t-1}, \hat{\mu}_{0,t}), \;\; \hat{\mu}_{0,t} = \hat{\mu}_\phi(h_{t-1}), \tag{5}$$

where $\hat{\mu}_{0,t} \in (0,1]$ and $\lambda_t \in [0, 1/\hat{\mu}_{0,t}]$ as before to ensure nonnegativity. The state $h_{t-1} = h_\eta(S_{1:t-1})$ with parameters η is itself a learned, compressed representation of the predictable score history, satisfying $\sigma(h_{t-1}) \subseteq \mathcal{F}_{t-1}$ rather than representing the complete filtration. The reference model targets the conditional null mean $\hat{\mu}_{0,t} \approx \mathbb{E}_0[S_t \mid h_{t-1}]$ to enforce normalization, while the betting factor λ_t learns a response to drive per-factor evidence growth (in the alternatives).

Model design. Rather than fitting a separate estimator at every trace step, we desire a lightweight, length-agnostic sequence model. A natural choice here is a *gated recurrent unit* (GRU, Chung et al. (2014)), whose single hidden state propagates a predictable summary of the preceding scores forward across steps. The calibration set is leveraged two-fold: the GRU encoder state h_η and reference head $\hat{\mu}_\phi$ are trained jointly on $\mathcal{D}^0_{\text{cal}}$ by next-score mean-squared error and then frozen. The betting head λ_θ subsequently maximizes terminal log-wealth on $\mathcal{D}^1_{\text{cal}}$, with an additional null drift penalty within coarse time bins. Hence the GRU state supplies predictable next-score representations, the reference estimates the necessary normalizer on that basis, and the bet drives evidence growth. Jointly, the model empirically approximates E_t as a conditionally normalized factor, and displays favorable behavior (Fig. 3). The exact objectives and validity conditions are given in § B.5.

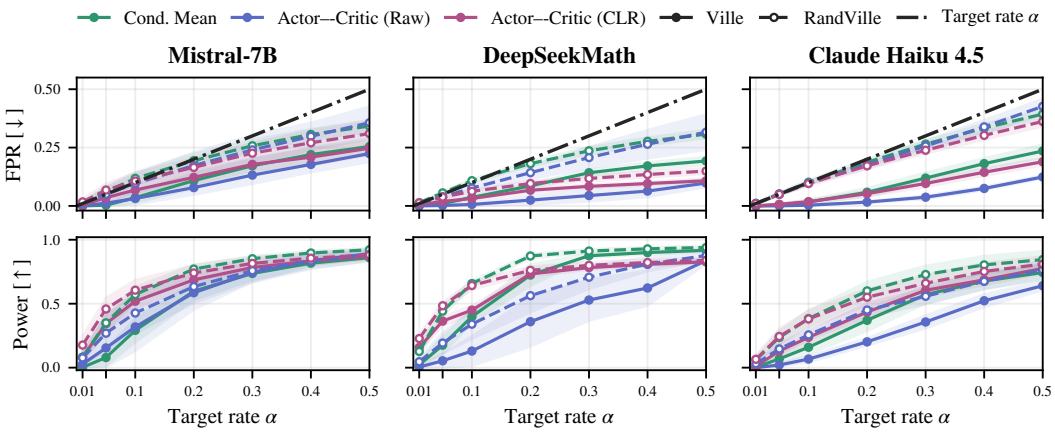


Figure 3: **Learned within-trace monitoring.** Across reasoning traces generated by Mistral-7B, DeepSeekMath, and Claude Haiku 4.5 and scored by Qwen2.5-PRM on MATH, learned tests generally track the target FPR, with modest deviations at low α under the more sensitive RandVille threshold. The structured conditional mean combines accurate empirical FPR with strong power, while actor–critic learning with a CLR warm start consistently improves upon raw-score initialization, suggesting a more informative history representation. Curves show means ± 1 SD across ten trials.

4.2 Learning An Actor–Critic Test

The mean construction in § 4.1 restricts E_t to be affine in the current score. To learn a richer factor shape, we consider the interplay of two role-separated components: an *actor* $a_\theta(h_{t-1}, S_t)$ observes the current score and proposes an evidence potential, while a *critic* $v_\phi(h_{t-1})$ sees only the predictable state and estimates its null log-normalizer. The resulting factor is

$$E_t = \exp(a_t - v_t - \kappa) \quad \text{with} \quad a_t = a_\theta(h_{t-1}, S_t),\ v_t = v_\phi(h_{t-1}), \tag{6}$$

where $\kappa \geq 0$ is an additional deployment margin. This form draws on exponential martingales, which offset a current-score potential by its log moment-generating function (Howard et al., 2020) and regret-based betting, which compares realized wealth to the optimal hindsight bet (Orabona & Jun, 2023). Our terminology loosely mirrors the role separation of actor–critic reinforcement learning (Arulkumaran et al., 2017): the actor proposes alternative-oriented evidence; the critic estimates the normalizing null 'policy'. Their difference is an advantage-like log-evidence increment toward $\mathcal{H}_1$.

Model design. As in the mean test, a shared GRU encoder constructs h_{t-1}, followed by separate actor and critic heads. Suppressing trace-balancing details, the objectives take the population form

$$\mathcal{L}_{\text{actor}} = -\mathbb{E}_1[a_t - v_t] + \beta \cdot \mathcal{C}_0, \qquad \mathcal{L}_{\text{critic}} = \mathbb{E}_0[\exp(a_t - v_t) + v_t]\,. \tag{7}$$

The actor's first term is the stepwise counterpart of the log-growth target in Eq. 4, while $\mathcal{C}_0$ with $\beta > 0$ additionally penalizes gains obtained through increased null drift. The critic loss targets the log-normalizer specific to the exponential design. For a fixed actor and an unrestricted critic, its pointwise population optimum is $v^\star(h_{t-1}) = \log \mathbb{E}_0[\exp(a_t) \mid h_{t-1}]$, which gives $\mathbb{E}_0[\exp(a_t - v^\star(h_{t-1})) \mid h_{t-1}] = 1$ and hence targets the desired null identity. Training follows a similar protocol as § 4.1, with the recurrent state and critic trained on null traces in a first stage, and alternating power-seeking actor updates with critic 'repair' steps in a second stage. After training, a final drift correction step determines κ and subtracts it from every active log-factor. § B.6 gives the exact losses and schedule.

Likelihood-ratio interpretation. Alongside end-to-end training from raw scores via Eq. 7, we consider a conditional likelihood-ratio (CLR) warm start. Binary cross-entropy losses train the critic to estimate alternative log-odds before observing S_t, and the actor to estimate them afterwards. At their population optima, Bayes' rule cancels the shared history-dependent odds, yielding the form

$$E_t^\star = \exp(a_t^* - v_t^*) = \frac{p_1(S_t \mid h_{t-1})}{p_0(S_t \mid h_{t-1})}, \qquad W_t^\star = \prod_{j=1}^{t} E_j^\star = \frac{p_1(S_{1:t})}{p_0(S_{1:t})}. \tag{8}$$

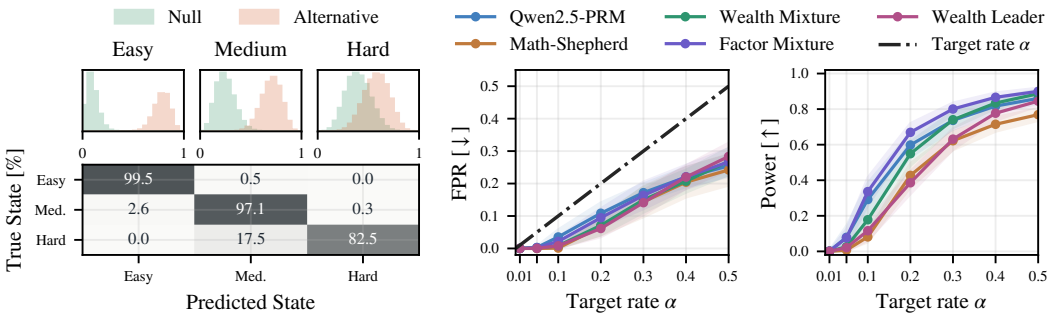


Figure 4: **State recovery and evidence composition.** *Left:* On a synthetic task with three predictable difficulty states (§ C.1), the GRU encoder h_η modeling the score history $h_{t-1} = h_\eta(S_{1:t-1})$ recovers the underlying state structure with high accuracy. *Right:* On Mistral-7B reasoning traces, Qwen2.5-PRM and Math-Shepherd scores provide complementary evidence streams. Composing their learned evidence at deployment through wealth mixtures, factor mixtures, or predictable routing can improve power over either scorer alone (§ B.8). Curves show means ± 1 SD across ten trials.

The actor–critic design thus recovers the log-optimal CLR under idealized optimization in response to Eq. 4 (Ramdas & Wang, 2025), motivating a classification warm start as a principled initialization. Interestingly, the resulting wealth W_t^* recovers the same full-trace likelihood-ratio used by Sadhuka et al. (2026) as an alarm statistic. Whereas they learn the ratio separately at every step (and hence store t separate models), our recurrent model elegantly factorizes it into estimated local increments.

Validity and composition. More formally, we propose the following sufficient conditions under which the learned actor–critic factors form a certified e-process, rather than only an approximation.

Proposition 1 (Sufficient conditional normalization) *Fix the learned encoder, actor, and critic before deployment. Suppose that, for every $P_0 \in \mathcal{P}_0$ and every supported t, the following hold P_0-almost surely: (i) h_{t-1} is $\mathcal{F}_{t-1}$-measurable and sufficient for the null next-score law,* i.e.

$$P_0(S_t \le s \mid \mathcal{F}_{t-1}) = P_0(S_t \le s \mid h_{t-1}) \quad \textit{for every } s \in [0,1];$$

and (ii) the critic provides a conditional log-normalizer, i.e. $v_t + \kappa \ge \log \mathbb{E}_0[\exp(a_t) \mid h_{t-1}]$. *Then $E_t = \exp(a_t - v_t - \kappa)$ satisfies Eq. 1 over its horizon, and W_t is a nonnegative test supermartingale.*

Condition *(i)* ensures that the conditional null distribution of S_t given the full history $\mathcal{F}_{t-1}$ depends on that history only through h_{t-1}. Condition *(ii)* then upper-bounds the corresponding conditional log-normalizer. Although the critic objective targets this condition, finite-data optimization certifies neither condition, and our learned factors therefore remain empirical. Prop. 1 nevertheless identifies a model-based route to structural validity. Under analogous conditions and a common filtration, fixed mixtures of wealth processes and predictable mixtures of local factors also preserve validity, as detailed in § B.8. We illustrate the resulting flexibility for deployment-time composition in Fig. 4.

5 Experiments

Experimental design. Our experiments address three leading questions: *(i)* how sequential testing changes when moving from across-trace to within-trace monitoring, *(ii)* whether learned history-dependent normalization can close the resulting gap and enable useful evidence composition, and *(iii)* how monitoring behavior transfers across tasks, models, and target properties. Our core study mirrors Schirmer et al. (2026b); Sadhuka et al. (2026) on sequential monitoring of MATH reasoning traces (Hendrycks et al., 2021), with Qwen2.5-Math-PRM-7B (Zhang et al., 2025b) providing the primary process-reward scores, and Math-Shepherd (Wang et al., 2024b) the paired scores for composition. We evaluate Mistral-7B-Instruct-v0.3 (Mistral, 2023), DeepSeekMath-7B-Instruct (Shao et al., 2024), and Claude Haiku 4.5 (Anthropic, 2025), which span solution accuracies of ca. 13%, 44%, and 94%.

Our broader cross-domain monitoring evaluation further covers harmful red-teaming interactions from HH-RLHF (Ganguli et al., 2022), scored by Llama-Guard-3-8B (Inan et al., 2023); failure to obtain a White win in historical chess games (LiChess, 2026), scored by Stockfish (Stockfish, 2026);

Table 1: **Cross-domain monitoring.** Empirical FPR and power for our learned within-trace monitors under RandVille thresholding for interaction, generation, and agentic tasks. Across diverse targets and trace structures, FPR remains at or below target in most cases; power varies with task difficulty and scorer separation. Values are means ± SD across ten trials; shaded entries exceed $\alpha = 0.2$.

Domain	Cond. Mean		Actor–Critic (Raw)		Actor–Critic (CLR)	
	FPR [↓]	Power [↑]	FPR [↓]	Power [↑]	FPR [↓]	Power [↑]
Interaction Tasks						
Red Teaming	0.20± 0.01	0.46± 0.05	0.17± 0.01	0.42± 0.06	0.17± 0.02	0.48± 0.03
Chess	0.21± 0.04	0.78± 0.04	0.11± 0.03	0.51± 0.07	0.08± 0.03	0.36± 0.10
Generation Tasks						
MATH	0.20± 0.01	0.52± 0.03	0.18± 0.02	0.37± 0.06	0.14± 0.01	0.45± 0.02
Coding (MBPP)	0.20± 0.01	0.41± 0.03	0.18± 0.02	0.46± 0.08	0.15± 0.01	0.73± 0.03
Agentic Tasks						
ALFWorld	0.19± 0.02	1.00± 0.00	0.16± 0.04	0.89± 0.09	0.20± 0.02	0.96± 0.01
DBBench	0.19± 0.01	0.52± 0.02	0.15± 0.01	0.33± 0.05	0.17± 0.01	0.73± 0.01
KnowledgeGraph	0.25± 0.05	0.85± 0.01	0.21± 0.08	0.70± 0.21	0.22± 0.08	0.69± 0.24
WebShop	0.20± 0.01	0.34± 0.02	0.16± 0.02	0.32± 0.07	0.17± 0.01	0.39± 0.02

incorrect OLMo-3-7B-Instruct MATH generations (Olmo Team, 2025) monitored using linear probes; Qwen3.5-4B code completions (Qwen Team, 2026a) monitored for hacking responses on MBPP (Austin et al., 2021); and unsuccessful Qwen3.6-27B tool-call trajectories (Qwen Team, 2026b) across four AgentBench environments (Liu et al., 2024). For each source we assign roughly 60% of traces to monitor fitting ($\mathcal{D}_{\text{cal}}$) and 40% to deployment, and report empirical FPR and power as means with ± 1 SD over ten trials. Source construction and implementation details are in App. C, with complete results including average run length (nARL) and ablations in App. D.

Across- and within-trace monitoring. Our results in Fig. 2 use the same Mistral–Qwen score source to isolate the effect of changing the statistical unit. Across traces, a reduction to a single summary score implies internal history no longer plays a leading role. A mean test with marginal plug-in reference $\hat{\mu}_0$ consequently maintains stable empirical FPR as the stream horizon grows ($H_i = 1, \ldots, 50$), illustrating its stream-level anytime property. By contrast, repeatedly applying per-trace CRC or UCB decisions accumulates false alarms across the stream. A fixed-horizon Bonferroni correction can prevent this, but at a cost in power (see Fig. 6). Within a reasoning trace, the same marginal reference no longer approximates the required history-conditional null mean. Conservative Ville crossing can conceal this mismatch, whereas RandVille exposes those FPR violations. PAC, UCB, and CRC remain effective as they calibrate the complete fixed alarm event through the trace-level maximum, thereby absorbing history-dependent errors.

Learned within-trace monitoring. Fig. 3 shows that empirical FPR remains close to target across all three reasoning sources, even under the more sensitive RandVille threshold. The structured conditional mean test from § 4.1 is particularly competitive in power, suggesting that a more constrained factor design and the comparatively easier task of predicting the next null score can capture substantial history dependence. CLR initialization also consistently improves actor–critic power over raw-score initialization, without systematic increases in FPR. One explanation is that the pre- versus post-score classification objective aligns the history state with the incremental information supplied by next scores, giving subsequent power optimization a better starting point. Our experiments do not separate representation from optimization effects, but the synthetic recovery result in Fig. 4 shows that a compact recurrent model can recover known predictable states under controlled assumptions. This suggests a learned state could be sufficient on real (but constrained) data as required by Prop. 1.

Our composition experiment illustrates the potential benefit of reusable sequential evidence. A *wealth mixture* averages two cumulative wealth processes, a *factor mixture* averages their current evidence factors before compounding, and a *wealth leader* assigns the next step to the component with greater preceding wealth. On Mistral reasoning traces, the learned Qwen and Math-Shepherd scores contain complementary information, allowing factor mixing to achieve greater power than either component alone. The PAC comparison in Fig. 16 distinguishes composing normalized evidence from reusing uncalibrated thresholds. Borrowed component thresholds can be too conservative and lose power, or anti-conservative and exceed the target FPR. Fresh calibration repairs the selected aggregate but must be repeated whenever a component or routing policy changes.

Cross-domain monitoring. Across diverse traces and targets, Tab. 1 shows favorable empirical FPR behavior. Even under the more sensitive RandVille threshold, most learned monitors remain at or below the target $\alpha = 0.20$. Power varies with scorer separation, trace length, and class balance (Fig. 17). For instance, ALFWorld exhibits strong score separation and correspondingly high power, whereas red teaming and MATH probing provide weaker separation and scorer accuracy. KnowledgeGraph is the notable FPR exception, as its rising null scores over time make history-dependent normalization difficult to estimate reliably. Overall, the results suggest that learned sequential monitors operate effectively across settings, and that no single design uniformly dominates.

6 Related Work

Sequential testing has become a practical tool for deployment monitoring of machine learning systems. Initial work tracks changes in labeled predictive risk, with subsequent extensions addressing unlabeled or gradual shifts (Podkopaev & Ramdas, 2022; Amoukou et al., 2024; Schirmer et al., 2026a; Timans et al., 2025). Related formulations connect betting to conformal procedures (Zecchin & Simeone, 2025; Podkopaev et al., 2024), exchangebility and changepoint detection (Prinster et al., 2025; Shaer et al., 2026; Shin et al., 2023), and other audits (Chugg et al., 2023a; Gauthier et al., 2026; Velasco et al., 2026). The perspective is also increasingly applied to generative models and agents. Most closely, Sadhuka et al. (2026); Schirmer et al. (2026b) address monitoring of reasoning traces with threshold calibration, and Huang et al. (2025) aggregate evidence to falsify agent-proposed experiment protocols. Broader work also studies generated-text detection (Chen & Wang, 2025; Huang et al., 2026), self-consistency (Ota et al., 2026; Dhillon et al., 2026), and performance evaluation (Park et al., 2026; Zhou et al., 2026). Several methods design their statistic around a known sampling or generation mechanism, so evidence construction and monitored target are inherently tied. Moreover, these problems often operate across filtrations where independence or exchangeability simplify normalization. We instead observe external black-box scores inside dependent traces, leaving crucial design choices unresolved. Regarding the learning of test properties, most relevant to us are Pandeva et al. (2024), who learn neural betting by maximizing log-wealth but obtain normalization from paired, antisymmetric observations; and Richter et al. (2025) who transfer this design to language-model auditing. Related works in a similar spirit include Shekhar & Ramdas (2023); He & Sutherland (2026); Taga et al. (2026). Our data-driven setting lacks any analogous transformations or symmetries, so conditional validity requires learned null normalization instead.

7 Discussion

We ask whether black-box scores from evolving generative traces can be converted into reusable sequential evidence. Distinguishing across- and within-trace monitoring, we develop structured, data-driven sequential monitors that exhibit favorable error control and detection properties but remain empirical approximations on real data. An outer PAC or threshold calibration step reintroduces decision-specific limitations, and does not resolve the local certification question. More fundamentally, every data-driven reference remains tied to its calibration population, and formal support ends at the active horizon. Changes to the generator, scorer, or labels can alter the null distribution, and stronger semantic anchoring of scores improves the target proxy but does not itself establish statistical control (see App. A). Finally, false alarm control alone supplies neither a lower bound on power nor guaranteed detectability of unfavorable traces. A hypothesis inversion may be useful for a singular model release decision rather than continuous monitoring, but tackles a different operational setting.

Several directions could help narrow these gaps. Auditable state representations, generalization bounds or hypothesis abstractions may help turn learned normalization into a verifiable statistical object (Chugg et al., 2023b; Larsson et al., 2026). Related theory could aid in characterizing power, log-wealth growth, and stopping times (Clerico, 2025; Waudby-Smith et al., 2025), or extend beyond means toward other targets (Wang et al., 2026a). White-box model access, multiple scorers, or prior information could alleviate any black-box information ceiling and yield richer signals, provided they adhere to merging and filtration rules (Vovk & Wang, 2024; Choe & Ramdas, 2026). At deployment scale, multiple traces or monitored properties may call for less-constrained controls (Wang & Ramdas, 2022; Xu & Ramdas, 2024), while model-assisted or synthetic data collection can reduce the calibration burden (Zrnic & Candès, 2024; Angelopoulos et al., 2023). Our broader objective with this direction is not merely a monitor that detects well, but one whose statistical contract can be inspected, certified, and safely reused as deployment systems evolve.

Ethics statement

This work aims to support more reliable monitoring and controlled intervention in generative model and agent deployments. Such monitoring nevertheless carries operational risks, as false alarms may interrupt useful behavior or trigger unnecessary review, while missed alarms may allow undesirable behavior to continue. Any statistical assurances concern the monitored score process under the stated calibration and deployment conditions, and do not by themselves establish semantic correctness or system safety. Performance may also reflect biases or misspecification in the datasets, labels, scorers, generators, and deployment population. Practical use should therefore include deployment-specific validation, human oversight, and interventions proportionate to the consequences of error.

Reproducibility statement

The methods and experimental protocol are described in § 4 and § 5. Assumptions and mathematical details, task construction and other implementation details, and additional results are provided in App. B, App. C, and App. D. All experiments use publicly available datasets and publicly accessible pretrained models, either open-source or accessible via API endpoints. Source code is not included with this submission, but we intend to release it upon acceptance subject to institutional approval and applicable policies.

AI use statement

Generative AI tools were employed to assist with developing conceptual and theoretical frameworks, providing feedback on research methodology and experimental design, supporting qualitative and thematic analysis, and as implementation aids for creating or editing code, scientific figures, and drafting and editing manuscript text. They were not used to formulate mathematical claims, supply critical ingredients and write mathematical proofs without human oversight. They were also not used to translate text, augment or reformat datasets, or interpret experimental results. The authors reviewed and verified all AI-assisted material and retain responsibility for the paper's claims, implementation, and conclusions.

References

Salim I Amoukou, Tom Bewley, Saumitra Mishra, Freddy Lecue, Daniele Magazzeni, and Manuela Veloso. Sequential harmful shift detection without labels. *Advances in Neural Information Processing Systems*, 2024. 9

Anastasios Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. Conformal risk control. *International Conference on Learning Representations*, 2024. 4, 20

Anastasios N Angelopoulos, Stephen Bates, Clara Fannjiang, Michael I Jordan, and Tijana Zrnic. Prediction-powered inference. *Science*, 2023. 9

Anthropic. Claude Haiku 4.5: System card, 2025. https://assets.anthropic.com/m/99128ddd009bdcb/Claude-Haiku-4-5-System-Card.pdf. 7, 26

Kai Arulkumaran, Marc Peter Deisenroth, Miles Brundage, and Anil Anthony Bharath. Deep reinforcement learning: A brief survey. *IEEE Signal Processing Magazine*, 2017. 6

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021. 8, 27

Sher Badshah, Ali Emami, and Hassan Sajjad. Scope: Selective conformal optimized pairwise llm judging. *International Conference on Machine Learning*, 2026. 17

Bowen Baker, Joost Huizinga, Leo Gao, Zehao Dou, Melody Y Guan, Aleksander Madry, Wojciech Zaremba, Jakub Pachocki, and David Farhi. Monitoring reasoning models for misbehavior and the risks of promoting obfuscation. *arXiv Preprint (arXiv:2503.11926)*, 2025. 1

Can Chen and Jun-Kun Wang. Online detection of llm-generated texts via sequential hypothesis testing by betting. *International Conference on Machine Learning*, 2025. 9

John J Cherian, Isaac Gibbs, and Emmanuel J Candès. Large language model validity via enhanced conformal prediction methods. *Advances in Neural Information Processing Systems*, 2024. 17

Yo Joong Choe and Aaditya Ramdas. Combining evidence across filtrations. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 2026. 9

Sadia Sultana Chowa, Riasad Alvi, Subhey Sadi Rahman, Md Abdur Rahman, Mohaimenul Azam Khan Raiaan, Md Rafiqul Islam, Mukhtar Hussain, and Sami Azam. From language to action: a review of large language models as autonomous agents and tool users. *Artificial Intelligence Review*, 2026. 1

Ben Chugg, Santiago Cortes-Gomez, Bryan Wilder, and Aaditya Ramdas. Auditing fairness by betting. *Advances in Neural Information Processing Systems*, 2023a. 9

Ben Chugg, Hongjian Wang, and Aaditya Ramdas. A unified recipe for deriving (time-uniform) pac-bayes bounds. *Journal of Machine Learning Research*, 2023b. 9

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *Deep Learning and Representation Learning Workshop at NeurIPS*, 2014. 5

Eugenio Clerico. On the optimality of coin-betting for mean estimation. *International Journal of Approximate Reasoning*, 2025. 4, 9

Pepijn Cobben, Xuanqiang Angelo Huang, Thao Amelia Pham, Isabel Dahlgren, Terry Jingchen Zhang, and Zhijing Jin. Gt-harmbench: Benchmarking ai safety risks through the lens of game theory. *arXiv preprint arXiv:2602.12316*, 2026. 1

Hen Davidov, Shai Feldman, Gilad Freidkin, and Yaniv Romano. Calibrated predictive lower bounds on time-to-unsafe-sampling in llms. *International Conference on Artificial Intelligence and Statistics*, 2026. 17

Guneet S Dhillon, Javier Gonzalez, Teodora Pandeva, and Alicia Curth. E-scores for (in) correctness assessment of generative model outputs. *International Conference on Artificial Intelligence and Statistics*, 2026. 9

Pavel Dolin, Weizhi Li, Gautam Dasarathy, and Visar Berisha. Statistically valid post-deployment monitoring should be standard for ai-based digital health. *Advances in Neural Information Processing Systems*, 2026. 1

Shai Feldman and Yaniv Romano. How many iterations to jailbreak? dynamic budget allocation for multi-turn llm evaluation. *arXiv preprint arXiv:2605.06605*, 2026. 17

Naihe Feng, Yi Sui, Shiyi Hou, Ga Wu, and Jesse C Cresswell. Conformal agent error attribution. *arXiv preprint arXiv:2605.06788*, 2026. 17

Deep Ganguli, Liane Lovitt, Jackson Kernion, Amanda Askell, Yuntao Bai, Saurav Kadavath, Ben Mann, Ethan Perez, Nicholas Schiefer, Kamal Ndousse, et al. Red teaming language models to reduce harms: Methods, scaling behaviors, and lessons learned. *arXiv Preprint (arXiv:2209.07858)*, 2022. 7, 26

Etienne Gauthier, Francis Bach, and Michael I Jordan. Betting on equilibrium: Monitoring strategic behavior in multi-agent systems. *International Conference on Machine Learning*, 2026. 9

Peter Grünwald, Rianne de Heide, and Wouter Koolen. Safe testing. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 2024. 5

Melody Y Guan, Miles Wang, Micah Carroll, Zehao Dou, Annie Y Wei, Marcus Williams, Benjamin Arnav, Joost Huizinga, Ian D Kivlichan, Amelia Glaese, et al. Monitoring monitorability. *International Conference on Machine Learning*, 2025. 1

Yu Gui, Ying Jin, and Zhimei Ren. Conformal alignment: Knowing when to trust foundation models with guarantees. *Advances in Neural Information Processing Systems*, 2024. 17

Lewis Hammond, Alan Chan, Jesse Clifton, Jason Hoelscher-Obermaier, Akbir Khan, Euan McLean, Chandler Smith, Wolfram Barfuss, Jakob Foerster, Tomáš Gavenčiak, The Anh Han, Edward Hughes, Vojtěch Kovařík, Jan Kulveit, Joel Z. Leibo, Caspar Oesterheld, Christian Schroeder de Witt, Nisarg Shah, Michael Wellman, Paolo Bova, Theodor Cimpeanu, Carson Ezell, Quentin Feuillade-Montixi, Matija Franklin, Esben Kran, Igor Krawczuk, Max Lamparth, Niklas Lauffer, Alexander Meinke, Sumeet Motwani, Anka Reuel, Vincent Conitzer, Michael Dennis, Iason Gabriel, Adam Gleave, Gillian Hadfield, Nika Haghtalab, Atoosa Kasirzadeh, Sébastien Krier, Kate Larson, Joel Lehman, David C. Parkes, Georgios Piliouras, and Iyad Rahwan. Multi-agent risks from advanced ai. *Tech Report, Cooperative AI Foundation*, 2025. 1

Zheng He and Danica J Sutherland. Sequential kernel-based conditional independence testing via adaptive betting. *arXiv preprint arXiv:2606.18993*, 2026. 9

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. *NeurIPS*, 2021. 7, 26, 27

Eliahu Horwitz and Yedid Hoshen. Conffusion: Confidence intervals for diffusion models. *arXiv preprint arXiv:2211.09795*, 2022. 17

Steven R Howard, Aaditya Ramdas, Jon McAuliffe, and Jasjeet Sekhon. Time-uniform chernoff bounds via nonnegative supermartingales. *Probability Surveys*, 2020. 6

Baihe Huang, Eric Xu, Kannan Ramchandran, Jiantao Jiao, and Michael I Jordan. Towards anytime-valid statistical watermarking. *arXiv preprint arXiv:2602.17608*, 2026. 9

Kexin Huang, Ying Jin, Ryan Li, Michael Y Li, Emmanuel Candes, and Jure Leskovec. Automated hypothesis validation with agentic sequential falsifications. *International Conference on Machine Learning*, 2025. 9

Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, et al. Llama guard: Llm-based input-output safeguard for human-ai conversations. *arXiv Preprint (arXiv:2312.06674)*, 2023. 1, 7, 26

Tomek Korbak, Mikita Balesni, Elizabeth Barnes, Yoshua Bengio, Joe Benton, Joseph Bloom, Mark Chen, Alan Cooney, Allan Dafoe, Anca Dragan, et al. Chain of thought monitorability: A new and fragile opportunity for ai safety. *arXiv preprint arXiv:2507.11473*, 2025. 1

Martin Larsson, Aaditya Ramdas, and Johannes Ruf. Testing hypotheses generated by constraints. *Mathematics of Operations Research*, 2026. 9

Chungpa Lee, Thomas Zeng, Jongwon Jeong, Jy-yong Sohn, and Kangwook Lee. How to correctly report llm-as-a-judge evaluations. *International Conference on Machine Learning*, 2026. 17

LiChess. LiChess Open Database, 2026. https://database.lichess.org/ [Accessed 24-09-2026]. 7, 26

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, Shudan Zhang, Xiang Deng, Aohan Zeng, Zhengxiao Du, Chenhui Zhang, Sheng Shen, Tianjun Zhang, Yu Su, Huan Sun, Minlie Huang, Yuxiao Dong, and Jie Tang. AgentBench: Evaluating LLMs as agents. *International Conference on Learning Representations*, 2024. 8, 27

Gabriel Loaiza-Ganem, Kevin Zhang, Wei Cui, Marc T Law, and Kin Kwan Leung. Conf-gen: Conformal uncertainty quantification for generative models. *International Conference on Machine Learning*, 2026. 17

Mistral. Mistral-7B-Instruct-v0.3: Model card, 2023. https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3. 7, 26

Christopher Mohri and Tatsunori Hashimoto. Language models with conformal factuality guarantees. *International Conference on Machine Learning*, 2024. 17

Olmo Team. Olmo 3. *arXiv Preprint (arXiv:2512.13961)*, 2025. 8

Francesco Orabona and Kwang-Sung Jun. Tight concentrations and confidence sequences from the regret of universal portfolio. *IEEE Transactions on Information Theory*, 2023. 6

Hirofumi Ota, Naoto Iwase, Yuki Ichihara, Junpei Komiyama, and Masaaki Imaizumi. Cite: Anytime-valid statistical inference in llm self-consistency. *arXiv preprint arXiv:2605.05873*, 2026. 9

Teodora Pandeva, Patrick Forré, Aaditya Ramdas, and Shubhanshu Shekhar. Deep anytime-valid hypothesis testing. *International Conference on Artificial Intelligence and Statistics*, 2024. 5, 9

Sangwoo Park, Matteo Zecchin, and Osvaldo Simeone. Adaptive prediction-powered autoeval with reliability and efficiency guarantees. *Advances in Neural Information Processing Systems*, 2026. 9

Aleksandr Podkopaev and Aaditya Ramdas. Tracking the risk of a deployed model and detecting harmful distribution shifts. *International Conference on Learning Representations*, 2022. 9

Aleksandr Podkopaev, Dong Xu, and Kuang chih Lee. Adaptive conformal inference by betting. *International Conference on Machine Learning*, 2024. 9

Nearchos Potamitis, Vansh Ramani, Har Ashish Arora, Dhairya Kuchhal, Lars Klein, and Akhil Arora. Reasonbench: Benchmarking the (in) stability of llm reasoning. *arXiv preprint arXiv:2512.07795*, 2025. 1

Drew Prinster, Xing Han, Anqi Liu, and Suchi Saria. Watch: Adaptive monitoring for ai deployments via weighted-conformal martingales. *International Conference on Machine Learning*, 2025. 9

Victor Quach, Adam Fisch, Tal Schuster, Adam Yala, Jae Ho Sohn, Tommi Jaakkola, and Regina Barzilay. Conformal language modeling. *International Conference on Learning Representations*, 2024. 17

Qwen Team. Qwen3.5: Towards native multimodal agents, 2026a. https://qwen.ai/blog?id=qwen3.5. 8

Qwen Team. Qwen3.6-27B: Flagship-level coding in a 27B dense model, 2026b. https://qwen.ai/blog?id=qwen3.6-27b. 8, 27

Aaditya Ramdas and Tudor Manole. Randomized and exchangeable improvements of markov's, chebyshev's and chernoff's inequalities. *Statistical Science*, 2026. 3

Aaditya Ramdas and Ruodu Wang. *Hypothesis testing with e-values*. Foundations and Trends® in Statistics, 2025. 1, 3, 7

Aaditya Ramdas, Peter Grünwald, Vladimir Vovk, and Glenn Shafer. Game-theoretic statistics and safe anytime-valid inference. *Statistical Science*, 2023. 1, 3

Anita Rao, Drew Keller, Neha Kalra, Ryan Steed, Kweku Kwegyir-Aggrey, Kevin Klyman, Diane Staheli, and Stevie Bergman. Challenges to the monitoring of deployed ai systems. *Tech Report, Center for AI Standards and Innovation*, 2026. 1

Leo Richter, Xuanli He, Pasquale Minervini, and Matt Kusner. An auditing test to detect behavioral shift in language models. *International Conference on Learning Representations*, 2025. 5, 9

Kai Ruan, Zihe Huang, Ziqi Zhou, Qianshan Wei, Jinghao Lin, Xuan Wang, and Hao Sun. Doomed from the start: Early abort of LLM agent episodes via a recall-controlled probe cascade. *arXiv preprint arXiv:2607.06503*, 2026. 27

Maxon Rubin-toles, Maya Gambhir, Keshav Ramji, Aaron Roth, and Surbhi Goel. Conformal language model reasoning with coherent factuality. *International Conference on Learning Representations*, 2025. 17

Shuvom Sadhuka, Drew Prinster, Clara Fannjiang, Gabriele Scalia, Bonnie Berger, Aviv Regev, and Hanchen. E-valuator: Reliable agent verifiers with sequential hypothesis testing. *Hypothesis Testing Workshop at ICML*, 2026. 2, 3, 4, 5, 7, 9, 17, 20, 21, 25, 29

Mona Schirmer, Metod Jazbec, Christian Andersson Naesseth, and Eric Nalisnick. Monitoring risks in test-time adaptation. *Advances in Neural Information Processing Systems*, 2026a. 9

Mona Schirmer, Metod Jazbec, Alexander Timans, Christian Naesseth, Maja Waldron, and Eric Nalisnick. Online safety monitoring for LLMs. *Hypothesis Testing Workshop at ICML*, 2026b. 2, 4, 7, 9, 20

Philip Sedgwick. Multiple significance tests: the bonferroni correction. *British Medical Journal*, 2012. 4

Shalev Shaer, Yarin Bar, Drew Prinster, and Yaniv Romano. Testing for distribution shifts with conditional conformal test martingales. *International Conference on Machine Learning*, 2026. 9

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv Preprint (arXiv:2402.03300)*, 2024. 7, 26

Shubhanshu Shekhar and Aaditya Ramdas. Nonparametric two-sample testing by betting. *IEEE Transactions on Information Theory*, 2023. 9

Huanxin Sheng, Xinyi Liu, Hangfeng He, Jieyu Zhao, and Jian Kang. Analyzing uncertainty of llm-as-a-judge: Interval evaluations with conformal prediction. *Conference on Empirical Methods in Natural Language Processing*, 2025. 17

Wentao Shi, Yu Wang, Yuyang Zhao, Yuxin Chen, Fuli Feng, Xueyuan Hao, Xi Su, Qi Gu, Hui Su, Xunliang Cai, et al. Aj-bench: Benchmarking agent-as-a-judge for environment-aware evaluation. *Findings of the Association for Computational Linguistics: ACL 2026*, 2026. 1

Jaehyeok Shin, Aaditya Ramdas, and Alessandro Rinaldo. E-detectors: A nonparametric framework for sequential change detection. *The New England Journal of Statistics in Data Science*, 2023. 9

Stockfish. Stockfish, 2026. https://github.com/official-stockfish/Stockfish [Accessed 24-09-2026]. 7, 26

Ege Onur Taga, Samet Oymak, and Shubhanshu Shekhar. Learning to bet for horizon-aware anytime-valid testing. *International Conference on Machine Learning*, 2026. 9

Jacopo Teneggi, Matthew Tivnan, Web Stayman, and Jeremias Sulam. How to trust your diffusion model: A convex optimization approach to conformal risk control. *International Conference on Machine Learning*, 2023. 17

Alexander Timans, Rajeev Verma, Eric Nalisnick, and Christian A Naesseth. On continuous monitoring of risk violations under unknown shift. *Conference on Uncertainty in Artificial Intelligence*, 2025. 9

Ander Artola Velasco, Stratis Tsirtsis, and Manuel Gomez Rodriguez. Auditing pay-per-token in large language models. *International Conference on Artificial Intelligence and Statistics*, 2026. 9

Jean Ville. *Etude critique de la notion de collectif*. Gauthier-Villars Paris, 1939. 3

Vladimir Vovk and Ruodu Wang. E-values: Calibration, combination and applications. *The Annals of Statistics*, 2021. 25

Vladimir Vovk and Ruodu Wang. Merging sequential e-values via martingales. *Electronic Journal of Statistics*, 2024. 9, 25

Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, et al. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 2024a. 1

Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. Math-shepherd: Verify and reinforce llms step-by-step without human annotations. *Proceedings of Association for Computational Linguistics*, 2024b. 7, 26

Qiuqi Wang, Ruodu Wang, and Johanna Ziegel. E-backtesting. *Management Science*, 2026a. 9

Ruodu Wang and Aaditya Ramdas. False discovery rate control with e-values. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 2022. 9

Xi Wang, Anushri Suresh, Alvin Zhang, Rishi More, William Jurayj, Benjamin Van Durme, Mehrdad Farajtabar, Daniel Khashabi, and Eric Nalisnick. Conformal thinking: Risk control for reasoning on a compute budget. *International Conference on Machine Learning*, 2026b. 17

Ian Waudby-Smith and Aaditya Ramdas. Estimating means of bounded random variables by betting. *Journal of the Royal Statistical Society: Series B*, 2024. 4, 5, 22, 29

Ian Waudby-Smith, Ricardo Sandoval, and Michael I Jordan. Universal log-optimality for general classes of e-processes and sequential hypothesis tests. *arXiv preprint arXiv:2504.02818*, 2025. 4, 9

Menghua Wu, Cai Zhou, Stephen Bates, and Tommi Jaakkola. Thought calibration: Efficient and confident test-time scaling. *Conference on Empirical Methods in Natural Language Processing*, 2025. 17

Ziyu Xu and Aaditya Ramdas. Online multiple testing with e-values. *International Conference on Artificial Intelligence and Statistics*, 2024. 9

Yasin Abbasi Yadkori, Ilja Kuzborskij, David Stutz, András György, Adam Fisch, Arnaud Doucet, Iuliya Beloshapka, Wei-Hung Weng, Yao-Yuan Yang, Csaba Szepesvári, et al. Mitigating llm hallucinations via conformal abstention. *arXiv preprint arXiv:2405.01563*, 2024. 17

Runyang You, Hongru Cai, Caiqi Zhang, Qiancheng Xu, Meng Liu, Tiezheng Yu, Yongqi Li, and Wenjie Li. Agent-as-a-judge. *arXiv preprint arXiv:2601.05111*, 2026. 17

Weiqiu You, Anton Xue, Shreya Havaldar, Delip Rao, Helen Jin, Chris Callison-Burch, and Eric Wong. Probabilistic soundness guarantees in llm reasoning chains. *Conference on Empirical Methods in Natural Language Processing*, 2025. 17

Matteo Zecchin and Osvaldo Simeone. Adaptive Learn-then-Test: Statistically Valid and Efficient Hyperparameter Selection. *International Conference on Machine Learning*, 2025. 9

Boxuan Zhang, Jianing Zhu, Zeru Shi, Dongfang Liu, and Ruixiang Tang. Agentforesight: Online auditing for early failure prediction in multi-agent systems. *arXiv preprint arXiv:2605.08715*, 2026. 17

Hanchen Zhang, Xiao Liu, Bowen Lv, Xueqiao Sun, Bohao Jing, Iat Long Iong, Zhenyu Hou, Zehan Qi, Hanyu Lai, Yifan Xu, Rui Lu, Hongning Wang, Jie Tang, and Yuxiao Dong. AgentRL: Scaling agentic reinforcement learning with a multi-turn, multi-task framework. *arXiv preprint arXiv:2510.04206*, 2025a. 27

Zhenru Zhang, Chujie Zheng, Yangzhen Wu, Beichen Zhang, Runji Lin, Bowen Yu, Dayiheng Liu, Jingren Zhou, and Junyang Lin. The lessons of developing process reward models in mathematical reasoning. *Findings of the Association for Computational Linguistics*, 2025b. 1, 7, 26

Zhijian Zhou, Zesheng Ye, Zhaorun Chen, Bo Li, and Feng Liu. CELEUS: Certifiable and efficient LLM evaluation via e-processes. *Hypothesis Testing Workshop at ICML*, 2026. 9

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, et al. Representation engineering: A top-down approach to ai transparency. *arXiv Preprint (arXiv:2310.01405)*, 2023. 1

Tijana Zrnic and Emmanuel J Candès. Active statistical inference. *International Conference on Machine Learning*, 2024. 9

—APPENDIX—
LEARNING TO TEST: SEQUENTIAL MONITORING OF GENERATIVE MODELS AND AGENTS

CONTENTS

1 Introduction **1**

2 Problem Setting and Background **2**

3 From Scores to Sequential Evidence **3**

4 Learning Sequential Tests **5**

4.1 Learning A Conditional Mean Test 5

4.2 Learning An Actor–Critic Test 6

5 Experiments **7**

6 Related Work **9**

7 Discussion **9**

A Additional Background and Discussion **17**

B Mathematical Details **18**

B.1 Definitions and Terminology 18

B.2 Across-Trace Mean Test 19

B.3 Example: Why Marginal Normalization Is Insufficient 20

B.4 Alarm Threshold Calibration 20

B.5 Learning a Conditional Mean Test 21

B.6 Learning an Actor–Critic Test 22

B.7 Likelihood-Ratio Interpretation 24

B.8 Evidence Composition 25

C Experimental and Implementation Details **26**

C.1 Data and Monitoring Tasks 26

C.2 Monitor Implementations 29

C.3 Training and Evaluation Protocol 30

D Additional Experimental Results **31**

D.1 Across-Trace Monitoring 31

D.2 Within-Trace Reasoning 34

D.3 Focused Actor–Critic Correction Diagnostics 41

D.4 Sufficiency and Composition 41

D.5 Other Monitoring Domains 44

A Additional Background and Discussion

Other statistical monitoring of generative models and agents. Conformal prediction and risk control offer a complementary route by calibrating prediction sets, abstention rules, or alarm events directly. Applications include risk-controlled early exit from reasoning chains (Wu et al., 2025; Wang et al., 2026b); correctness, factuality, and abstention guarantees (Rubin-toles et al., 2025; Mohri & Hashimoto, 2024; Cherian et al., 2024; Quach et al., 2024; Yadkori et al., 2024); survival-style modeling of sampling or evaluation budgets (Feldman & Romano, 2026; Davidov et al., 2026); and alignment, judge, or agent-error calibration (Gui et al., 2024; Sheng et al., 2025; Badshah et al., 2026; Feng et al., 2026). Related conformal methods address generative tasks beyond text (Teneggi et al., 2023; Horwitz & Hoshen, 2022; Loaiza-Ganem et al., 2026). These methods can provide useful uncertainty quantification or calibrated decision regions (e.g., prediction sets, abstention rules, or alarm events) for an output and decision rule specified before calibration. Our goal is instead reusable local evidence that can be accumulated and composed before an alarm policy is fixed, approached via the sequential testing route. A separate line of work builds problem-specific probabilistic models, for instance of reasoning soundness or of judge error (You et al., 2025; Lee et al., 2026), and recent Agent-as-a-Judge and learned agent-monitoring systems propose a range of bespoke detectors (You et al., 2026; Zhang et al., 2026). These are typically heuristic, as they report a score without accompanying error guarantees.

Calibration and deployment populations. The data-driven monitor is fitted from $\mathcal{D}_{\mathrm{cal}}$ and then held fixed during deployment. Calibration-to-deployment stability is a shared requirement of learned evidence, CRC, risk-UCB, PAC thresholds, and fitted likelihood-ratio methods. For their respective statements to transfer, future traces must follow the same relevant population as the corresponding calibration subset. In particular, null calibration traces and future favorable traces must be exchangeable for CRC, risk-UCB, and PAC threshold calibration, while an across-trace upper-mean reference additionally requires the deployment trace summaries to satisfy the population mean bound used in its construction. The necessary stability concerns the complete monitoring pipeline: the generator and its decoding or action policy, task or prompt distribution, scorer and score sign convention, trace-construction protocol, terminal-label definition, trace-level aggregation, and active-horizon convention. A change to any of these objects can change either the null score law or the calibrated alarm event, even when the underlying property of interest remains unchanged. Each statistical statement therefore attaches to a specific configuration of these components, and a change to any of them requires fresh calibration.

Conditioning on $\mathcal{D}_{\mathrm{cal}}$ makes fitted quantities deterministic for deployment-time analysis, but does not establish that calibration and deployment share a population. The statistical guarantees in this paper therefore concern the induced labeled score population under the stated stability assumptions. Whether these guarantees say anything about the underlying property of interest further depends on the quality of the terminal labels and on how faithfully the scorer measures it.

Active horizons and extrapolation. Let T_{cal} denote the largest declared step supported by the fitted monitor and $H_i = \min\{T_i, T_{\mathrm{cal}}\}$ the active horizon of trace i. A shorter trace ends at its observed terminal time T_i; a longer trace is monitored formally only through T_{cal}. For an evidence process, ceasing accumulation after support is equivalent to setting subsequent factors $(E)_{t>T_{\mathrm{cal}}} = 1$, so the stopped wealth remains unchanged. For a calibrated decision rule, the fitted event is similarly restricted to the maximum statistic over $t \le H_i$. Scores or fitted model outputs may still be displayed beyond this horizon as diagnostics, but they do not enter the formal rejection event.

This support restriction is distinct from optional stopping. A conditionally valid e-process may be stopped at any data-dependent time within its supported filtration without invalidating Ville's bound. A recurrent reference can still be evaluated on longer histories and may extrapolate smoothly when the underlying dynamics remain stable, but outputs beyond T_{cal} are model predictions rather than formally supported evidence. The same distinction applies to fitted likelihood-ratio estimates (Sadhuka et al., 2026). Extending the active horizon, changing a post-support fallback, or altering the trace-length distribution therefore requires a new validity argument or fresh calibration.

Reformulating the monitoring hypothesis? The monitoring target is an underlying property of interest, seen only through terminal labels and imperfect scores. Unlike settings where the null is

pinned down by exchangeability, a known randomization design, or a simulator one can sample from, nothing here singles out a conditional distribution for the next score given the past, so the test design is underdetermined. Weakening the null to a bound on the running average does not help: a wealth process built as a product of per-step multipliers still requires each multiplier to have conditional null expectation at most one, even when only cumulative 'badness' is the claim.

A relative monitor may instead decide to test

$$\mathbb{E}_0[S_{i,t}^{\mathrm{dep}} - S_{i,t}^{\mathrm{base}} \mid \mathcal{F}_{t-1}] \leq 0,$$

which replaces an absolute favorable-behavior claim by non-degradation of the deployed model scores $S_{i,t}^{\mathrm{dep}}$ relative to those of a chosen base model $S_{i,t}^{\mathrm{base}}$. Validity still requires the paired difference to satisfy this conditional inequality, and what can be claimed on rejection is correspondingly relative: evidence that the deployed model has degraded against the base model, not that its behaviour is poor in absolute terms. A probability-integral transform can yield a simple marginal reference when the null score distribution is known and continuous, while rank or exchangeability methods can exploit a stable reference sample. In the present setting, however, these approaches require the reference distribution and exchangeability assumptions to remain valid after the observed prefix. Marginal calibration of the transformed score is not the same as calibration conditional on every history.

Similarly, the population prefix likelihood ratio

$$\frac{p_1(S_{1:t})}{p_0(S_{1:t})}$$

is an exact e-process when the two densities are the true prefix densities under the null and the alternative, but replacing either density by a fitted model generally removes exact normalization. One may then certify the local fitted ratios, or calibrate the maximum fitted statistic as a fixed alarm event. The latter is the PAC route described in the main text. These alternatives therefore either introduce a new structural assumption, change the scientific claim, or transfer calibration from the evidence factors to the final decision.

B Mathematical Details

B.1 Definitions and Terminology

Fix a null distribution $P_0 \in \mathcal{P}_0$ on an underlying probability space $(\Omega, \mathcal{A}, P_0)$. For trace i, let $U_{i,t}$ denote the random information admitted to the monitor's filtration at step t, and let $\mathcal{F}_{i,0}$ contain all information fixed or revealed before the first monitored score, including the fitted monitor, calibration-derived quantities, initial context, and any pre-drawn randomization. The corresponding deployment filtration is

$$\mathcal{F}_{i,t} := \mathcal{F}_{i,0} \vee \sigma(U_{i,1}, \ldots, U_{i,t}), \qquad t \geq 1,$$

where $\sigma(U_{i,1}, \ldots, U_{i,t})$ is the smallest sigma-field with respect to which the revealed variables are measurable. Hence $(\mathcal{F}_{i,t})_{t\geq 0}$ is increasing: $\mathcal{F}_{i,0} \subseteq \mathcal{F}_{i,1} \subseteq \cdots$. This construction imposes no independence assumption. The score-only interface in the main text takes $U_{i,t} = S_{i,t}$; a filtration for the complete generated trace could instead take $U_{i,t} = (X_{i,t}, S_{i,t})$, where $X_{i,t}$ represents observed states or actions. Conditioning on $\mathcal{F}_{i,0}$ fixes all pre-deployment quantities while retaining them in the formal information set.

Fixing i and writing $\mathcal{F}_t := \mathcal{F}_{i,t}$, a real-valued process $(Z_t)_{t\geq 0}$ is *adapted* to $(\mathcal{F}_t)_{t\geq 0}$ if Z_t is $\mathcal{F}_t$-measurable for every t. It is a P_0-*martingale* if it is adapted, integrable, and

$$\mathbb{E}_0[Z_t \mid \mathcal{F}_{t-1}] = Z_{t-1} \qquad P_0\text{-almost surely for every } t \geq 1.$$

Replacing equality by $\leq$ defines a P_0-supermartingale. A nonnegative supermartingale $(W_t)_{t\geq 0}$ with $W_0 = 1$ is a test supermartingale; in particular, $\mathbb{E}_0[W_t] \leq 1$ for every t.

A sequence $(\lambda_t)_{t\geq 1}$ is *predictable* if λ_t is $\mathcal{F}_{t-1}$-measurable for every t, so its value must be fixed before the step-t information is revealed. A random time $\tau : \Omega \to \{0, 1, \ldots\} \cup \{\infty\}$ is a stopping time if $\{\tau \leq t\} \in \mathcal{F}_t$ for every t. Thus the decision whether to stop at step t may use information through step t, but cannot depend on future observations. If (Z_t) is a supermartingale, then the

stopped process $(Z_{t\wedge\tau})_{t\geq 0}$ is also a supermartingale. Consequently, a test supermartingale satisfies $\mathbb{E}_0[W_{t\wedge\tau}] \leq 1$ and Ville's inequality

$$\mathbb{P}_0\left(\sup_{t\geq 0} W_t \geq \frac{1}{\alpha}\right) \leq \alpha, \qquad \alpha \in (0,1).$$

This is the formal basis for the anytime-valid crossing rule in Eq. 2.

The factor construction in Eq. 1 is a sufficient local condition for this behavior. If E_t is nonnegative and $\mathcal{F}_t$-measurable with $\mathbb{E}_0[E_t \mid \mathcal{F}_{t-1}] \leq 1$, then $W_t = \prod_{j=1}^{t} E_j$ obeys

$$\mathbb{E}_0[W_t \mid \mathcal{F}_{t-1}] = W_{t-1}\mathbb{E}_0[E_t \mid \mathcal{F}_{t-1}] \leq W_{t-1}.$$

Therefore $(W_t)_{t\geq 0}$ is a test supermartingale. The factor E_t may depend on the newly observed step-t information, whereas every bet, reference, allocation, or normalizer used to construct it must be predictable. Since P_0 was arbitrary, a composite-null guarantee requires these inequalities to hold under every $P_0 \in \mathcal{P}_0$. We suppress the trace index i in the within-trace derivations below; the across-trace construction instead uses trace index i as its sequential index.

B.2 Across-Trace Mean Test

Let $S_i^{\mathrm{tr}} \in [0,1]$ be the scalar summary of completed trace i, and define the across-trace filtration

$$\mathcal{F}_i := \mathcal{F}_0 \vee \sigma(S_1^{\mathrm{tr}}, \ldots, S_i^{\mathrm{tr}}),$$

where $\mathcal{F}_0$ contains the fitted monitor and all pre-deployment information. Suppose a reference null mean $\mu_0 \in (0,1]$ satisfies

$$\mathbb{E}_0[S_i^{\mathrm{tr}} \mid \mathcal{F}_{i-1}] \leq \mu_0 \qquad \text{for every } i$$

under every $P_0 \in \mathcal{P}_0$. Let λ_i be $\mathcal{F}_{i-1}$-measurable with $0 \leq \lambda_i \leq 1/\mu_0$, and set

$$E_i^{\mathrm{tr}} := 1 + \lambda_i(S_i^{\mathrm{tr}} - \mu_0), \qquad W_i^{\mathrm{tr}} := \prod_{j=1}^{i} E_j^{\mathrm{tr}}, \qquad W_0^{\mathrm{tr}} := 1.$$

The bet constraint gives $E_i^{\mathrm{tr}} \geq 1 - \lambda_i\mu_0 \geq 0$. Predictability and the null upper bound further give

$$\mathbb{E}_0[E_i^{\mathrm{tr}} \mid \mathcal{F}_{i-1}] = 1 + \lambda_i\left(\mathbb{E}_0[S_i^{\mathrm{tr}} \mid \mathcal{F}_{i-1}] - \mu_0\right) \leq 1.$$

Consequently,

$$\mathbb{E}_0[W_i^{\mathrm{tr}} \mid \mathcal{F}_{i-1}] = W_{i-1}^{\mathrm{tr}}\mathbb{E}_0[E_i^{\mathrm{tr}} \mid \mathcal{F}_{i-1}] \leq W_{i-1}^{\mathrm{tr}},$$

so $(W_i^{\mathrm{tr}})_{i\geq 0}$ is a nonnegative test supermartingale and Ville's inequality yields anytime false-alarm control. If completed traces are independently sampled from a common null population with mean μ_0^{pop}, then

$$\mathbb{E}_0[S_i^{\mathrm{tr}} \mid \mathcal{F}_{i-1}] = \mathbb{E}_0[S_i^{\mathrm{tr}}] = \mu_0^{\mathrm{pop}},$$

and we can construct the reference null mean μ_0 using any common marginal upper bound to μ_0^{pop}. Dependence among the scores within a completed trace affects the distribution of S_i^{tr}, but no longer enters the across-trace conditioning once the trace has been summarized.

An upper bounding high-probability reference. The calibration mean used in the main text is

$$\widehat{\mu}_0 := \frac{1}{|\mathcal{D}_{\mathrm{cal}}^0|}\sum_{i\in\mathcal{D}_{\mathrm{cal}}^0} S_i^{\mathrm{tr}}.$$

It is a natural empirical plug-in reference, but it is not generally an upper bound on μ_0^{pop} and does not by itself establish an e-process. Suppose instead that an independent null calibration sample produces an upper confidence reference $\widehat{\mu}_0^{\mathrm{UCB}} = \widehat{\mu}_0^{\mathrm{UCB}}(\mathcal{D}_{\mathrm{cal}}^0)$ satisfying

$$\mathbb{P}_{\mathcal{D}_{\mathrm{cal}}^0}\left(\mu_0^{\mathrm{pop}} \leq \widehat{\mu}_0^{\mathrm{UCB}}\right) \geq 1 - \delta.$$

Here and moving forward, distributions explicitly fitted from calibration data are indexed as $P_{\mathcal{D}_{\mathrm{cal}}}$, $P_{\mathcal{D}_{\mathrm{cal}}^0}$, or $P_{\mathcal{D}_{\mathrm{cal}}^1}$. On this coverage event, using $\widehat{\mu}_0^{\mathrm{UCB}}$ as the reference in the preceding construction gives, conditionally on $\mathcal{D}_{\mathrm{cal}}^0$,

$$\mathbb{P}_{\mathcal{D}_{\mathrm{cal}}^0}\left(\mathbb{P}_0\left(\sup_{i\geq 1} W_i^{\mathrm{tr}} \geq \frac{1}{\alpha} \;\middle|\; \mathcal{D}_{\mathrm{cal}}^0\right) \leq \alpha\right) \geq 1 - \delta.$$

The reference-coverage error δ is separate from Ville's crossing level α. Writing $\mathbb{P}_{\mathcal{D}^0_{\mathrm{cal}},0}$ for the joint calibration and null-deployment law, marginalizing over both draws gives the conservative bound

$$\mathbb{P}_{\mathcal{D}^0_{\mathrm{cal}},0}\left(\sup_{i\geq 1} W_i^{\mathrm{tr}} \geq \frac{1}{\alpha}\right) \leq \alpha + \delta - \alpha\delta.$$

A sample mean, empirical quantile, or fitted point prediction without a corresponding upper-mean coverage statement does not support this conclusion.

Atomic versus stream-level false alarms. Applying a calibrated threshold independently to each completed trace controls the risk of *one trace-level decision*, not the probability of any alarm over an indefinitely growing stream. If each of N independent null traces has alarm probability p, then

$$\mathbb{P}_0(\text{at least one alarm in } N \text{ traces}) = 1 - (1-p)^N.$$

Without independence, the union bound gives an upper bound of Np. A Bonferroni construction can target stream-level error α over a separately declared horizon N by calibrating each trace at level α/N, but changing N changes the rule. This contrasts with a structurally valid across-trace e-process, whose Ville crossing probability remains bounded by α over an unbounded stream.

B.3 Example: Why Marginal Normalization Is Insufficient

The distinction between marginal and conditional normalization can be seen directly within the mean-factor family. Let $S_1 \sim \mathrm{Bernoulli}(1/2)$ and set $S_t = S_1$ for all $t \geq 2$. For $t \geq 2$, take the marginally correct reference $\widehat{\mu}_{0,t} = 1/2$ and predictable bet $\lambda_t = 2$, giving

$$E_1 = 1, \qquad E_t = 1 + 2(S_t - 1/2) = 2S_t.$$

For every $t \geq 2$, $\mathbb{E}_0[S_t] = \widehat{\mu}_{0,t}$ and $\mathbb{E}_0[E_t] = 1$. Nevertheless, after observing $S_1 = 1$,

$$\mathbb{E}_0[E_t \mid \mathcal{F}_{t-1}] = 2, \qquad t \geq 2,$$

and the wealth equals $W_t = 2^{t-1}$ on an event of probability $1/2$. For any $\alpha < 1/2$, a sufficiently long trace therefore crosses $1/\alpha$ with null probability $1/2 > \alpha$. Thus a reference can match every marginal score mean while failing the history-conditional upper-bound requirement, allowing apparently normalized factors to compound into an invalid process.

B.4 Alarm Threshold Calibration

Following the fixed-rule calibration used by the closest baselines (Sadhuka et al., 2026; Schirmer et al., 2026b), let $A_{i,t}$ be a statistic fixed before threshold calibration and define the null trace-level maximum

$$M_i := \max_{t \leq H_i} A_{i,t}.$$

For a cutoff c, the corresponding false-alarm risk is $R(c) := \mathbb{P}_0(M_i > c)$. This construction calibrates the complete ever-crossing event and imposes no requirement that $A_{i,t}$ be an e-value.

Conformal risk control and upper confidence bounds. Let $M_1, \ldots, M_{n_0}$ be the maxima of n_0 exchangeable null calibration traces and define

$$\widehat{R}_{n_0}(c) := \frac{1}{n_0} \sum_{i=1}^{n_0} \mathbb{1}\{M_i > c\}.$$

CRC (Angelopoulos et al., 2024) selects the least conservative candidate cutoff $\widehat{\tau}_{\mathrm{CRC}}$ whose finite-sample-corrected empirical risk satisfies

$$\frac{n_0}{n_0+1}\widehat{R}_{n_0}(c) + \frac{1}{n_0+1} \leq \alpha.$$

Under exchangeability and the nested threshold family, this yields the in-expectation guarantee

$$\mathbb{E}_{\mathcal{D}^0_{\mathrm{cal}}}[R(\widehat{\tau}_{\mathrm{CRC}})] \leq \alpha.$$

A risk-UCB procedure instead evaluates an upper confidence bound $U(c; \mathcal{D}^0_{\mathrm{cal}}, \delta)$ for $R(c)$ and selects the least conservative cutoff $\widehat{\tau}_{\mathrm{UCB}}$ satisfying $U(c; \mathcal{D}^0_{\mathrm{cal}}, \delta) \leq \alpha$. Its guarantee is

$$\mathbb{P}_{\mathcal{D}^0_{\mathrm{cal}}}(R(\widehat{\tau}_{\mathrm{UCB}}) \leq \alpha) \geq 1 - \delta.$$

PAC calibration. Sadhuka et al. (2026) first fits a separate full-prefix statistic $A_{i,t}$ from labeled calibration traces. When the population prefix likelihood ratio is known exactly, it is an e-process under the corresponding null and alternative laws. A fitted classifier-based ratio does not inherit that property automatically. The PAC variant therefore fixes the fitted statistic, computes M_i on an independent null threshold-calibration split $\mathcal{D}^0_{\text{thr}}$, and chooses a one-sided tolerance bound $\widehat{\tau}_{\text{PAC}}$. Conditional on the statistic-fitting split $\mathcal{D}_{\text{fit}}$, the resulting statement has the form

$$\mathbb{P}_{\mathcal{D}^0_{\text{thr}}}\left(\mathbb{P}_0\left(M_i > \widehat{\tau}_{\text{PAC}} \mid \mathcal{D}_{\text{fit}}, \mathcal{D}^0_{\text{thr}}\right) \le \alpha\right) \ge 1 - \delta.$$

The CRC, risk-UCB, and PAC statements all concern a fixed trace-level event. If the statistic, active-horizon convention, aggregation, routing rule, or intervention event changes, then so does M_i, and the new event requires calibration.

B.5 Learning a Conditional Mean Test

Validity of the mean factor. Condition on the fitted pre-deployment quantities in $\mathcal{F}_0$. For $S_t \in [0,1]$, let $\widehat{\mu}_{0,t} \in (0,1]$ and $\lambda_t \in [0, 1/\widehat{\mu}_{0,t}]$ be $\mathcal{F}_{t-1}$-measurable, and suppose the reference satisfies

$$\widehat{\mu}_{0,t} \ge \mathbb{E}_0[S_t \mid \mathcal{F}_{t-1}] \qquad \text{almost surely under every } P_0 \in \mathcal{P}_0.$$

Define

$$E_t := 1 + \lambda_t(S_t - \widehat{\mu}_{0,t}) \qquad \text{and} \qquad W_t := \prod_{j=1}^{t} E_j.$$

Because the smallest possible current score is zero,

$$E_t \ge 1 - \lambda_t \widehat{\mu}_{0,t} \ge 0.$$

Moreover, predictability of the reference and bet gives

$$\mathbb{E}_0[E_t \mid \mathcal{F}_{t-1}] = 1 + \lambda_t\left(\mathbb{E}_0[S_t \mid \mathcal{F}_{t-1}] - \widehat{\mu}_{0,t}\right) \le 1,$$

and hence

$$\begin{aligned}\mathbb{E}_0[W_t \mid \mathcal{F}_{t-1}] &= W_{t-1}\mathbb{E}_0[E_t \mid \mathcal{F}_{t-1}] \\ &\le W_{t-1}.\end{aligned}$$

Thus (W_t) is a nonnegative test supermartingale over the supported horizon. Setting subsequent factors to one as in § B.8 gives the null crossing guarantee in Eq. 2. This is a population statement under each $P_0 \in \mathcal{P}_0$, conditional on the realized calibration data. A learned point predictor, marginal calibration, or a bound conditional only on a lossy state h_{t-1} does not establish the displayed full-filtration inequality.

Reference and betting training. The implemented reference applies one causal rule across variable-length traces. Suppressing the start indicator, its GRU state is

$$h_{i,t-1} = h_\eta(S_{i,1:t-1}, t/T_{\text{cal}}),$$

which satisfies $\sigma(h_{i,t-1}) \subseteq \mathcal{F}_{i,t-1}$. The current score, future scores, and eventual trace length are excluded. Let $\mathcal{D}^y_{\text{learn}} \subseteq \mathcal{D}^y_{\text{cal}}$ denote the label-y traces in the method-specific learning split. The state map and reference head are fitted jointly on $\mathcal{D}^0_{\text{learn}}$ by transition-averaged squared error,

$$\mathcal{L}_{\text{ref}}(\eta, \phi) := \frac{1}{N_0} \sum_{i \in \mathcal{D}^0_{\text{learn}}} \sum_{t=1}^{H_i} \left\{S_{i,t} - \widehat{\mu}_\phi(h_{i,t-1})\right\}^2, \qquad N_0 := \sum_{i \in \mathcal{D}^0_{\text{learn}}} H_i.$$

At the population optimum over an unrestricted function class, squared-error regression recovers $\mathbb{E}_0[S_t \mid h_{t-1}]$. This equals the required full-history conditional mean only if the state preserves the null next-score mean. Even then, a finite-sample point estimate need not be an upper bound. A certified construction would require a predictable one-sided envelope $b_t(h_{t-1})$ such that, simultaneously over supported histories and times,

$$\mathbb{E}_0[S_t \mid \mathcal{F}_{t-1}] \le \widehat{\mu}_{0,t} := \min\{1, \widehat{\mu}_\phi(h_{t-1}) + b_t(h_{t-1})\}$$

on a stated calibration event. The real-data implementation uses the point predictor without such an envelope, so its Ville-family replay remains an empirical diagnostic.

After fitting, the state and reference are frozen. The betting head

$$\lambda_t = \lambda_\theta(h_{t-1}, \widehat{\mu}_{0,t})$$

uses a bounded output layer to enforce

$$0 \le \lambda_t \le \frac{c}{\widehat{\mu}_{0,t}}, \quad c \in (0,1),$$

where c denotes an optional conservative constant to limit aggressive betting at the tolerance limits (Waudby-Smith & Ramdas, 2024), and takes the default value $c = 1$. The increment $\Delta_{i,t} := E_{i,t} - 1 = \lambda_{i,t}(S_{i,t} - \widehat{\mu}_{0,i,t})$ then satisfies $E_{i,t} \ge 1 - c$. Partition normalized time into a fixed set of bins b. For each null trace occupying bin b, let $\overline{\Delta}_{i,b}$ be its mean increment in that bin, and let $\overline{\Delta}_b$ average these trace means. The betting head minimizes

$$\mathcal{L}_{\mathrm{bet}}(\theta) := -\frac{1}{|\mathcal{D}^1_{\mathrm{learn}}|} \sum_{i \in \mathcal{D}^1_{\mathrm{learn}}} \sum_{t=1}^{H_i} \log E_{i,t} + \gamma \sum_b [\overline{\Delta}_b]^2_+,$$

where $\gamma \ge 0$ is fixed before fitting. The first term maximizes mean alternative terminal log-wealth. The second replaces the history-conditional normalization requirement by a small number of grouped null-moment checks. Since $E_t = 1 + \Delta_t$, requiring a nonpositive grouped mean increment is equivalent to requiring a grouped mean factor of at most one, or a nonpositive grouped log-normalizer. It is therefore a deliberately coarse approximation: histories with different next-score laws may cancel within the same time bin. We found this penalty sufficiently effective as a practical stabilizer in the reported experiments, but it is auxiliary rather than the source of normalization. The learned reference supplies the intended null baseline, and the penalty cannot repair an invalid reference or establish the required full-history inequality. If the upper-reference premise does hold, any predictable bet in the displayed range remains valid regardless of how alternative traces were used to tune it.

B.6 Learning an Actor–Critic Test

Sufficient conditional normalization. We first restate Prop. 1 before proving the claimed null constraint. Condition on the fitted encoder, actor, and critic in $\mathcal{F}_0$.

Proposition 1 (Sufficient conditional normalization) *Fix the learned encoder, actor, and critic before deployment. Suppose that, for every $P_0 \in \mathcal{P}_0$ and every supported t, the following hold P_0-almost surely: (i) h_{t-1} is $\mathcal{F}_{t-1}$-measurable and sufficient for the null next-score law,* i.e.

$$P_0(S_t \le s \mid \mathcal{F}_{t-1}) = P_0(S_t \le s \mid h_{t-1}) \quad \textit{for every } s \in [0,1];$$

and (ii) the critic provides a conditional log-normalizer, i.e. $v_t + \kappa \ge \log \mathbb{E}_0[\exp(a_t) \mid h_{t-1}]$. *Then $E_t = \exp(a_t - v_t - \kappa)$ satisfies Eq. 1 over its horizon, and W_t is a nonnegative test supermartingale.*

Proof. The state and critic are predictable, while the actor and factor are $\mathcal{F}_t$-measurable; exponential factors are nonnegative. Condition *(i)* gives equality of the full-history and state-conditional expectations for every integrable measurable function of S_t, including $s \mapsto \exp(a_\theta(h_{t-1}, s))$. Therefore

$$\begin{aligned}
\mathbb{E}_0[E_t \mid \mathcal{F}_{t-1}] &= \exp(-v_t - \kappa)\mathbb{E}_0[\exp(a_t) \mid \mathcal{F}_{t-1}] \\
&= \exp(-v_t - \kappa)\mathbb{E}_0[\exp(a_t) \mid h_{t-1}] \\
&\le \exp(-v_t - \kappa)\exp(v_t + \kappa) = 1.
\end{aligned}$$

Consequently,

$$\mathbb{E}_0[W_t \mid \mathcal{F}_{t-1}] = W_{t-1}\mathbb{E}_0[E_t \mid \mathcal{F}_{t-1}] \le W_{t-1},$$

which proves the claim for every $P_0 \in \mathcal{P}_0$. The finite-horizon neutralization in § B.8 extends the stopped wealth without accumulating unsupported factors. For a fixed actor, the full next-score sufficiency condition can be weakened to invariance of the single conditional moment $\mathbb{E}_0[\exp(a_t) \mid \cdot]$. The stronger condition in the proposition is actor-independent and matches the main-text statement.

Population log-normalizer target. For a fixed actor and history state h, consider the population critic risk

$$\mathcal{R}_h(v) := \mathbb{E}_0[\exp(a_\theta(h, S_t) - v) + v \mid h] .$$

Its first and second derivatives are

$$\mathcal{R}'_h(v) = 1 - \mathbb{E}_0[\exp(a_\theta(h, S_t) - v) \mid h], \qquad \mathcal{R}''_h(v) = \mathbb{E}_0[\exp(a_\theta(h, S_t) - v) \mid h] > 0.$$

The unique population minimizer is therefore

$$\arg\min_{v\in\mathbb{R}} \mathcal{R}_h(v) = \log \mathbb{E}_0[\exp(a_\theta(h, S_t)) \mid h],$$

which makes $\mathbb{E}_0[\exp(a_t - v^*(h)) \mid h] = 1$. The additive v term is essential: minimizing only $\exp(a_t - v)$ would send the critic upward without bound, while the displayed risk has the desired finite optimum. This population calculation explains why the critic, rather than an actor-side penalty, is the primary normalization mechanism.

The empirical problem is harder than the population expression suggests. A continuous recurrent state observed in a real trace is generally paired with only one realized next score, providing approximately one Monte Carlo draw from the corresponding conditional law. The implemented outer average therefore combines many different states, and the shared critic must pool information across them to estimate the conditional log-normalizer. The population joint risk retains the pointwise optimum for an unrestricted critic, but finite conditional sampling, the model class, and state sufficiency determine whether the fitted critic recovers it. A low held-out mean factor is useful empirical evidence, but itself not a history-uniform upper bound.

Population and finite-data losses. The abstract population objectives in Eq. 7 are

$$\mathcal{L}_{\text{actor}} = -\mathbb{E}_1[a_t - v_t] + \beta\mathcal{C}_0, \qquad \mathcal{L}_{\text{critic}} = \mathbb{E}_0[\exp(a_t - v_t) + v_t].$$

The implementation instantiates $\mathcal{C}_0$ as a trace-balanced null mean factor. For the resulting finite-data objectives, let $a_{i,t} := a_\theta(h_{i,t-1}, S_{i,t})$ and $v_{i,t} := v_\phi(h_{i,t-1})$, and let stopgrad denote a detached quantity. The final actor objective consists of alternative growth, *Actor Loss 1*, and an auxiliary null-factor guard, *Actor Loss 2*. Defining $\mathcal{D}^0_{\text{learn}}, \mathcal{D}^1_{\text{learn}}$ as the null- and alternative-facing training subsets of $\mathcal{D}_{\text{cal}}$ used for learned component fitting, we have the losses

$$\mathcal{L}_{\text{A1}} := -\frac{1}{|\mathcal{D}^1_{\text{learn}}|} \sum_{i\in\mathcal{D}^1_{\text{learn}}} \frac{1}{H_i} \sum_{t=1}^{H_i} (a_{i,t} - v_{i,t}),$$

$$\mathcal{L}_{\text{A2}} := \frac{1}{|\mathcal{D}^0_{\text{learn}}|} \sum_{i\in\mathcal{D}^0_{\text{learn}}} \frac{1}{H_i} \sum_{t=1}^{H_i} \exp(a_{i,t} - \text{stopgrad}(v_{i,t})),$$

$$\mathcal{L}_{\text{actor}} := \mathcal{L}_{\text{A1}} + \beta\mathcal{L}_{\text{A2}}.$$

The trace means prevent longer traces from determining the update scale. $\mathcal{L}_{\text{A1}}$ seeks power, while $\mathcal{L}_{\text{A2}}$ discourages an indiscriminate actor shift that raises null factors. Because it averages over the fitted null histories rather than conditioning on each state, Actor Loss 2 is only a coarse marginal proxy for the null constraint. It is an extra guardrail, not the force that estimates the log-normalizer.

With the actor fixed, the implemented critic objective is *Critic Loss 1*, the trace-balanced empirical analogue of the population risk:

$$\mathcal{L}_{\text{C1}} := \frac{1}{|\mathcal{D}^0_{\text{learn}}|} \sum_{i\in\mathcal{D}^0_{\text{learn}}} \frac{1}{H_i} \sum_{t=1}^{H_i} \{\exp(\text{stopgrad}(a_{i,t}) - v_{i,t}) + v_{i,t}\} .$$

The actor is detached during Critic Loss 1, and the critic is detached during both actor losses, preserving the role separation. We also considered one-sided actor penalties within normalized-time bins and within fixed clusters of frozen GRU hidden states. These alternatives pin empirical null factors in coarse predictable groups and connect to the binned guard used by the mean test, but they can merge histories with different next-score laws. They were less natural than directly fitting the critic's population log-normalizer and are not part of the final locked objective.

Post-fit deployment correction. To turn a fitted critic into a certificate, one would need a simultaneous one-sided statement over all deployment-reachable histories and supported times, for example

$$\log \mathbb{E}_0[\exp(a_\theta(h_{t-1}, S_t)) \mid \mathcal{F}_{t-1}] \le v_\phi(h_{t-1}) + b_t$$

on an explicitly stated calibration event, with b_t predictable and the fitted actor and critic fixed before deployment. The empirical correction variant is instead applied only after fitting and checkpoint selection. It searches a fixed grid using held-out null traces, *e.g.* $\{0, \dots, 0.10\}$. For each candidate κ, let $\widehat{\mathrm{FPR}}_\alpha(\kappa)$ be the validation FPR of the corrected Ville replay and define

$$d(\kappa) := \max_{\alpha \in \mathcal{A}_{\mathrm{sel}}} \left\{ \widehat{\mathrm{FPR}}_\alpha(\kappa) - (\alpha + \varepsilon_{\mathrm{FPR}}) \right\},$$

where $\mathcal{A}_{\mathrm{sel}}$ is the predeclared grid, and $\varepsilon_{\mathrm{FPR}}$ is the fixed tolerance. The procedure chooses the smallest κ with $d(\kappa) \le 0$. If no candidate is feasible, it chooses the lexicographically smallest pair $(d(\kappa), \kappa)$ and records a least-worst-excess fallback. The selected value is subtracted from every active log-factor of the frozen process. The model is neither retrained nor rescored, and one selected κ is reused across reported deployment values of α.

If the uniform population inequality

$$\log \mathbb{E}_0[\exp(a_\theta(h_{t-1}, S_t) - v_\phi(h_{t-1})) \mid \mathcal{F}_{t-1}] \le \kappa$$

were known, the correction would be valid. The procedure is therefore a pragmatic global drift correction, but not a conditional-validity certificate. PAC calibration may still be placed around the corrected maximum statistic, in which case the guarantee is the event-level statement of § B.4.

B.7 Likelihood-Ratio Interpretation

Matched classifier odds. The conditional-likelihood-ratio warm start has a direct population interpretation. Under the effective training law induced by the shared class and trace weights, binary cross-entropy gives the post-score actor logit

$$a^\star(h, s) = \log \frac{\mathbb{P}(Y = 1 \mid h, S_t = s)}{\mathbb{P}(Y = 0 \mid h, S_t = s)}$$

and the pre-score critic logit

$$v^\star(h) = \log \frac{\mathbb{P}(Y = 1 \mid h)}{\mathbb{P}(Y = 0 \mid h)}.$$

Bayes' rule expands the first odds as

$$\frac{\mathbb{P}(Y = 1 \mid h, S_t = s)}{\mathbb{P}(Y = 0 \mid h, S_t = s)} = \frac{p_1(s \mid h)\mathbb{P}(Y = 1 \mid h)}{p_0(s \mid h)\mathbb{P}(Y = 0 \mid h)}.$$

Because the actor and critic use the same effective class weighting, the history-dependent odds cancel upon subtraction:

$$a^\star(h, s) - v^\star(h) = \log \frac{p_1(s \mid h)}{p_0(s \mid h)}.$$

Under the usual support condition, exponentiating gives the conditional likelihood-ratio factor

$$E_t^\star = \frac{p_1(S_t \mid h_{t-1})}{p_0(S_t \mid h_{t-1})}, \qquad \mathbb{E}_0[E_t^\star \mid h_{t-1}] = \int \frac{p_1(s \mid h_{t-1})}{p_0(s \mid h_{t-1})} p_0(s \mid h_{t-1})\, ds = 1.$$

If h_{t-1} preserves the relevant next-score laws under both P_0 and P_1, the chain rule yields

$$W_t^\star = \prod_{j=1}^{t} E_j^\star = \prod_{j=1}^{t} \frac{p_1(S_j \mid S_{1:j-1})}{p_0(S_j \mid S_{1:j-1})} = \frac{p_1(S_{1:t})}{p_0(S_{1:t})}.$$

Tilted-null interpretation. The same connection follows from exact critic normalization. For $\kappa = 0$, suppose

$$v_\phi(h) = \log \mathbb{E}_0[\exp(a_\theta(h, S_t)) \mid h].$$

Then

$$q_\theta(s \mid h) := \exp(a_\theta(h, s) - v_\phi(h))p_0(s \mid h)$$

integrates to one, and the actor–critic factor is

$$E_t = \frac{q_\theta(S_t \mid h_{t-1})}{p_0(S_t \mid h_{t-1})}.$$

Alternative log-growth therefore fits a normalized null tilt toward the alternative conditional law. At the unrestricted population optimum we have $q_\theta(\cdot \mid h) = p_1(\cdot \mid h)$, again giving the CLR.

This is the same population full-prefix likelihood ratio targeted by Sadhuka et al. (2026), who estimate that ratio afresh at each supported prefix, whereas the recurrent actor–critic shares parameters over time and factorizes it into local increments. The latter representation aligns naturally with local evidence normalization, but its approximation errors can potentially compound. If $\log E_j = \log E_j^\star + \delta_j$, then

$$\log W_t - \log W_t^\star = \sum_{j=1}^{t} \delta_j$$

is the compounded error term. Small systematic local errors can therefore accumulate across a long trace. After Stage 2 power and repair updates, the learned actor-minus-critic value is best viewed as a log-factor initialized by the likelihood-ratio structure, not as an exact posterior-odds decomposition.

B.8 Evidence Composition

Finite-horizon validity. Suppose the local factors of trace i satisfy

$$E_{i,t} \geq 0, \qquad \mathbb{E}_0[E_{i,t} \mid \mathcal{F}_{i,t-1}] \leq 1 \qquad \text{for } t \leq H_i,$$

where $H_i = \min\{T_i, T_{\text{cal}}\}$ is the supported stopping horizon. Define the stopped factors and wealth by

$$\widetilde{E}_{i,t} := \begin{cases} E_{i,t}, & t \leq H_i, \\ 1, & t > H_i, \end{cases} \qquad \widetilde{W}_{i,t} := \prod_{j=1}^{t} \widetilde{E}_{i,j} = W_{i,\min\{t,H_i\}}.$$

Because H_i is a stopping time and T_{cal} is fixed before deployment, the event $\{t \leq H_i\}$ is $\mathcal{F}_{i,t-1}$-measurable. Therefore

$$\begin{aligned} \mathbb{E}_0[\widetilde{E}_{i,t} \mid \mathcal{F}_{i,t-1}] &= \mathbb{1}\{t \leq H_i\}\mathbb{E}_0[E_{i,t} \mid \mathcal{F}_{i,t-1}] + \mathbb{1}\{t > H_i\} \\ &\leq 1. \end{aligned}$$

Thus $\widetilde{W}_{i,t}$ is a test supermartingale. Neutral factors preserve the evidence accumulated through the supported horizon without making a claim about later scores. Components with different supported horizons must either be composed through their minimum common horizon or individually neutralized after support.

Composition rules. Standard e-value combination and sequential merging results motivate two useful constructions (Vovk & Wang, 2021; 2024). Let $W_t^{(1)}, \ldots, W_t^{(K)}$ be nonnegative test supermartingales with $W_0^{(k)} = 1$ under the same null and with respect to a common filtration $\mathcal{F}_t$. For fixed, or $\mathcal{F}_0$-measurable, weights $\pi_k \geq 0$ satisfying $\sum_{k=1}^{K} \pi_k = 1$, define

$$W_t^{\text{mix}} := \sum_{k=1}^{K} \pi_k W_t^{(k)}.$$

This process starts at one and satisfies

$$\mathbb{E}_0[W_t^{\text{mix}} \mid \mathcal{F}_{t-1}] \leq \sum_{k=1}^{K} \pi_k W_{t-1}^{(k)} = W_{t-1}^{\text{mix}}.$$

Thus a fixed convex mixture of valid wealth processes is valid. Alternatively, suppose each component supplies a conditionally valid factor $E_t^{(k)}$, and let $q_{k,t} \geq 0$ be $\mathcal{F}_{t-1}$-measurable allocations satisfying $\sum_{k=1}^{K} q_{k,t} = 1$. Then

$$E_t^{\text{route}} := \sum_{k=1}^{K} q_{k,t} E_t^{(k)}$$

obeys

$$\mathbb{E}_0[E_t^{\text{route}} \mid \mathcal{F}_{t-1}] \leq \sum_{k=1}^{K} q_{k,t} = 1,$$

so $W_t^{\text{route}} = \prod_{j=1}^{t} E_j^{\text{route}}$ is a test supermartingale. Predictable self-financing allocation and one-component routing are special cases. Allocations may depend on preceding scores, states, or component wealth, but not on the current factors or scores being routed.

The common-filtration requirement is substantive. A component proved valid only under a coarser information set need not remain valid after conditioning on the enlarged history used by another component or by the router. No independence among components is required once each conditional guarantee holds under the common filtration. Fixed wealth mixtures and predictable factor mixtures are different processes, and validity is not generally preserved by arbitrary products or post-score selection of the largest current factor. Most importantly, composition preserves validity but does not create it: each input must already satisfy its conditional null requirement. The fitted real-data components in this paper are empirical approximations, so their composition results remain empirical approximations.

C Experimental and Implementation Details

C.1 Data and Monitoring Tasks

MATH reasoning-trace monitoring. We evaluate within-trace correctness monitoring on three sets of MATH solutions. For each of the 5,000 problems in the official test split (Hendrycks et al., 2021), Mistral-7B-Instruct-v0.3 (Mistral, 2023), DeepSeekMath-7B-Instruct (Shao et al., 2024), and Claude Haiku 4.5 (Anthropic, 2025) each generate a reasoned solution and boxed final answer. `math_verify` assigns the terminal correctness label; missing, unparseable, and truncated final answers are treated as incorrect. After each reasoning step, Qwen2.5-Math-PRM-7B (Zhang et al., 2025b) scores the complete causal prefix. We use the alternative-oriented score $1 - \texttt{judge_probability}$, so larger values indicate greater evidence of eventual incorrectness. Aligned Math-Shepherd scores (Wang et al., 2024b) provide the second stream for composition. Terminal labels are available for development and evaluation but not to the deployed monitor.

Red-teaming interaction monitoring. We use conversations from the Anthropic HH-RLHF red-teaming release (Ganguli et al., 2022), restricted to the 52B context-distillation model. A conversation is removed if its original and sanitized turn counts differ, leaving 17,644 conversations. The 7,370 conversations with red-team rating at least three are alternatives, so the terminal label denotes a successful harmful interaction rather than the safety of every message. The monitoring unit is a completed assistant response, scored by its Llama Guard 3 unsafe probability (Inan et al., 2023); larger values indicate stronger evidence of a harmful interaction. User-message checkpoints are excluded so the score tracks assistant behavior. Because scoring occurs after each response, this is a post-response monitoring task.

Chess outcome monitoring. We use 1,111 historical Lichess games comprising 69,023 recorded plies (LiChess, 2026). Each monitoring step is the position after one historical move, evaluated by Stockfish (Stockfish, 2026). The stored White-win probability is converted to the score $S_t = 1 - P_t(\text{White win})$, so larger scores indicate that the game is moving away from a White victory. White wins form the favorable null; Black wins and draws are alternatives. This corresponds to monitoring an agent playing for White.

MATH correctness monitoring. We use the 5,000-problem official MATH test split (Hendrycks et al., 2021). OLMo-3-7B-Instruct greedily generates one solution per problem from a prompt requesting reasoning and a final `\boxed{}` expression, with at most 4,096 new tokens. We save the residual state at the final prompt token, running prefix-mean states at 64, 128, ..., 512 generated tokens and every 128 tokens through 2,048, and the complete-solution mean residual state. An answer-equivalence procedure compares the extracted final answer with the reference; unparseable or truncated generations are incorrect. A standardized logistic-regression probe predicts eventual correctness from residual representations pooled across all 22 landmarks. It is fitted on 2,000 MATH training problems and validated on a disjoint 500, selecting the layer and regularization by pooled validation AUROC, with accuracy, smaller C, and earlier layer as tie-breakers. The selected probe is frozen before application to the official test set, and the monitoring score is $1 - p(\text{correct})$.

MBPP hacking monitoring. We use the sanitized MBPP program-synthesis benchmark (Austin et al., 2021) with a Qwen3.5-4B checkpoint after 14 fine-tuning steps. Each task is presented under a neutral programming prompt with two revealed tests and the remaining tests withheld for evaluation. For each held-out prompt, the model produces 64 completions (one greedy and 63 sampled) using temperature 0.7, top-$p = 0.95$, and at most 1,536 new tokens. We save post-attention residual activations at prompt end, at 64, 128, 192, 256, 384, and 512 generated tokens, and at the final token. Eligible completions contain a parsed reasoning trace and code block and pass the revealed tests. An *observed hack* then fails a withheld test, whereas a *generalizing pass* passes both test sets. This yields 4,623 eligible completions from 85 held-out tasks, including 623 observed hacks. A standardized logistic regression uses the running mean residual stream over the response prefix, with the prompt-token residual mean at prompt end. Hyperparameters are selected by validation AUROC over the pooled fixed landmarks using completion-equal training weights. The resulting score is $p(\text{hack})$.

AgentBench failure monitoring. We evaluate four AgentBench tool-use environments: Database (DBBench), ALFWorld, Knowledge Graph (KG), and WebShop, together with the corresponding AgentRL training tasks (Liu et al., 2024; Zhang et al., 2025a). We exclude Operating System (OS) because its 504 traces contain only 106 failures. For each remaining task, Qwen3.6-27B (Qwen Team, 2026b) runs once in instruct (non-thinking) mode. The terminal environment outcome labels the trajectory; for WebShop, only reward exactly one is a success, and partial rewards are failures. A monitoring step is a completed assistant turn, normally an action or tool call. At its closing `<|im_end|>` token, after the action but before the next environment response, we extract the 5,120-dimensional residual-stream activation. This follows hidden-state probing of failures in tool-using agents (Ruan et al., 2026).

For each environment, a standardized logistic-regression probe predicts the terminal success label at every turn. Each probe uses 512 training trajectories and minimizes L2-regularized binary cross-entropy with inverse-trajectory-length weights and equal class contributions. We search the full-attention layers of Qwen3.6-27B, every fourth block in its 3:1 Gated DeltaNet/full-attention stack $(4, 8, 12, \ldots, 64)$. Regularization is selected by trajectory-grouped five-fold cross-validation, and the layer by turn-wise AUROC averaged over active turns on a disjoint, outcome-balanced set of 128 trajectories. The exported score is $1 - p(\text{success})$. Evaluation combines canonical AgentBench tasks with unused training tasks after excluding all probe-fitting and layer-selection trajectories:

Environment	Trajectories	Monitored turns	Mean turns/task	Turn cap	Failures (%)
DBBench	4,394	21,739	4.95	15	1,784 (40.6)
ALFWorld	2,619	37,952	14.49	20	1,096 (41.8)
KG	724	8,983	12.41	15	522 (72.1)
WebShop	10,560	79,649	7.54	20	7,441 (70.5)

The downstream split preserves the distinction between training and canonical standard/test tasks. All standard/test trajectories enter deployment; seeded, outcome-stratified training-task samples complete its configured size. The remaining training trajectories supply the calibration and auxiliary partitions, so no canonical standard/test trajectory fits a probe-dependent monitor component or threshold.

Synthetic sufficient-state construction. The controlled experiment for Prop. 1 generates two conditionally independent scorer processes $m \in \{1, 2\}$ with predictable states $Z_t^{(m)} \in \{E, M, D\}$, initial state $Z_1^{(m)} = E$, and horizon $H = 12$. For outcome $Y \in \{0, 1\}$,

$$S_t^{(m)} \mid Z_t^{(m)} = z, Y = y \sim \mathrm{Beta}(\kappa_z \mu_{y,z,m},\ \kappa_z(1 - \mu_{y,z,m}))\,,$$

with state rows (E, M, D) and scorer columns $(1, 2)$ given by

$$[\mu_{0,z,m}] = \begin{pmatrix} 0.10 & 0.12 \\ 0.25 & 0.23 \\ 0.40 & 0.37 \end{pmatrix}, \qquad [\mu_{1,z,m}] = \begin{pmatrix} 0.78 & 0.74 \\ 0.70 & 0.77 \\ 0.58 & 0.61 \end{pmatrix}, \qquad (\kappa_E, \kappa_M, \kappa_D) = (30, 22, 16).$$

Let $Q_{0,z,m}(p)$ denote the p-quantile of the state-specific null law and define score bands using $(p_z^{\mathrm{lo}}, p_z^{\mathrm{hi}}) = ((.10, .90), (.10, .90), (.45, .90))$. Thus $B_t^{(m)} = L$ below the lower quantile, H above the upper quantile, and M otherwise. The deterministic transition model is

$$Z_{t+1}^{(m)} = T\left(Z_t^{(m)}, B_t^{(m)}\right), \qquad \begin{array}{c|ccc} T & L & M & H \\ \hline E & E & E & M \\ M & E & M & D \\ D & M & D & D \end{array}.$$

The same null-derived cutpoints apply under both outcomes, so each state path is reconstructible from preceding scores. A separate scalar GRU predicts each null stream's next-score mean without state labels, the other scorer, or oracle means. Post-fit nearest-prototype decoding diagnoses state recovery but does not affect deployment.

We run 20 repetitions. Each uses 3,000 null traces for reference fitting, 3,000 disjoint null traces for PAC calibration, and 3,000 deployment traces per outcome. The fixed bet is 0.9, and $\alpha \in \{.01, .05, .10, .15, .20, .25, .30, .35, .40, .45, .50\}$.

Oracle and learned composition. For component factors $E_t^{(m)}$ and wealths $W_t^{(m)} = \prod_{u \le t} E_u^{(m)}$, we evaluate

$$\begin{aligned} W_t^{\mathrm{wealth}} &= \tfrac{1}{2} W_t^{(1)} + \tfrac{1}{2} W_t^{(2)}, \\ W_t^{\mathrm{factor}} &= \prod_{u=1}^{t} \left(\tfrac{1}{2} E_u^{(1)} + \tfrac{1}{2} E_u^{(2)}\right), \\ p_t^{(1)} &= \mathbb{1}\left\{W_{t-1}^{(1)} > W_{t-1}^{(2)}\right\} + \tfrac{1}{2}\mathbb{1}\left\{W_{t-1}^{(1)} = W_{t-1}^{(2)}\right\}, \qquad p_t^{(2)} = 1 - p_t^{(1)}, \\ W_t^{\mathrm{leader}} &= \prod_{u=1}^{t} \left(p_u^{(1)} E_u^{(1)} + p_u^{(2)} E_u^{(2)}\right). \end{aligned}$$

Exact state-conditional null means locally normalize the oracle components, establishing the structural claim for all three policies. Learned GRU references and the aligned Qwen2.5-PRM/Math-Shepherd experiment in Fig. 4 remain empirical. Every changed aggregate receives a new PAC threshold; borrowed thresholds are transfer diagnostics.

Splits and active horizons. Except for the source-anchored AgentBench construction above, each trial uses an outcome-stratified 30% primary calibration, 30% auxiliary, and 40% deployment split. Trace IDs are disjoint, deployment order is shuffled without changing within-trace order, and the procedure is repeated over ten seeds. Methods requiring class contrast, including PAC calibration and actor–critic learning, use fitting alternatives. For AgentBench, every canonical standard/test trace remains in deployment and training-source traces fill the remaining partitions, as specified above. Every fitted statistic has an active horizon equal to the largest step supported by its fitted reference, model, or calibration replay. A shorter trace ends at its observed terminal step. Rows after the active horizon may be retained for visualization but are inactive: their factors, alarms, and run length do not enter evaluation. Paired composition uses the smaller component horizon.

Method	Statistic or component fitting	Threshold or validation use
Direct raw	None	Numerical α cutoff
Fixed mean	Pooled primary and auxiliary null traces	Direct Ville/RandVille crossing
Learned mean/bet	Pooled primary and auxiliary traces; alternatives used for the learned bet	20% of eligible fitting IDs for validation
Actor–critic	Pooled primary and auxiliary null and alternative traces	20% internal validation and checkpoint selection
CRC/UCB	Pooled primary and auxiliary null traces	Scalar cutoff fitted to null risk
PAC calibration	Statistic fitted on the primary partition when needed	Auxiliary-null active-path maxima calibrate the cutoff

Table 2: Calibration-partition use by monitor family under the default 30% primary, 30% auxiliary, and 40% deployment split. Recurrent learned components reserve 20% only within the fitting partition shown.

C.2 Monitor Implementations

Baselines and calibrated decision rules. The direct raw baseline alarms when the running maximum score strictly exceeds the numerical α cutoff. CRC selects the least conservative grid cutoff whose finite-sample corrected null risk is at most the target; Bentkus-UCB and binomial control use a one-sided concentration bound or exact binomial test with failure probability δ. Each produces a sticky alarm from the running score maximum.

Sadhuka et al. (2026) fits a separate full-prefix density-ratio classifier at each supported step. Ville and RandVille replay this statistic directly. Its PAC version, and PAC-wrapped raw scores and composition aggregates, calibrate the statistic's auxiliary-null path maximum with a one-sided distribution-free tolerance bound. The error budget is $(\alpha_{\text{tail}}, \delta) = (0.9\alpha, 0.1\alpha)$, following Sadhuka et al. (2026).

For across-trace experiments, each completed trace is reduced to its mean score. The raw comparator uses that scalar directly; CRC/Bentkus applies its single-trace threshold repeatedly; Bonferroni fits the same rule at α/H for a separately declared stream length H; and the mean e-process compounds bounded mean-test factors across trace means. The calibration mean and empirical-Bernstein UCB shown in the appendix are plug-in reference comparisons.

Learned conditional mean test. The marginal mean monitor estimates a separate null score mean at each step and carries the last supported value forward; its UCB variant adds an empirical-Bernstein radius with confidence parameter 0.1. Both are fitted references, not population conditional bounds.

The conditional-mean monitor instead uses a one-layer GRU with the preceding score, start indicator, and normalized step index. Before observing S_t, a sigmoid head predicts $\widehat{\mu}_t \in [0, 1]$; the factor is $1 + \lambda_t(S_t - \widehat{\mu}_t)$. Null next-score mean-squared error fits the reference, and the current score updates the state only after the mean and bet are produced.

The five predictable betting rules follow Waudby-Smith & Ramdas (2024): unit sets $\lambda_t = 1$; aGRAPA clips an estimated nonnegative growth-optimal rate; GRAPA selects the fixed-grid bet with greatest preceding log growth; mixture weights all grid bets by preceding wealth; and the learned GRU–MLP maps the reference state to a bounded bet while rewarding alternative log growth and penalizing positive null mean factors in three time bins.

Learned actor–critic test. Both initialization routes use the same causal GRU state h_{t-1}, a predictable critic $v_t = v(h_{t-1})$, and an actor $a_t = a(h_{t-1}, S_t)$. Their uncorrected local log-factor is $a_t - v_t$.

Raw-score Stage 1 fixes $a_{i,t} = S_{i,t}$ and fits the encoder and critic on null traces. Stage 2 initializes the actor at the identity map with centered residual

$$a_\theta(h, s) = s + r_\theta(h, s) - r_\theta(h, 0),$$

which removes any history-only residual shift.

Conditional likelihood-ratio (CLR) Stage 1 uses matched trace-balanced binary cross-entropy: the actor predicts the trace label after seeing S_t, and the critic predicts it from h_{t-1}. Writing $\mathrm{BCE}_{\mathrm{tr}}$ for

Component	Learning rate	Epochs and update unit	Objective or selection
Conditional mean	10^{-3}	25–600; batches of 64 traces; patience 25	One-layer GRU/sigmoid; null next-score MSE
Learned bet	10^{-3}	25–400; full development tensor per epoch; patience 25	Alternative growth; null penalty 10 over 3 bins
Raw Stage 1	10^{-3}	25–400; full development tensor; patience 25	Critic Loss 1 with identity actor
CLR Stage 1	10^{-3}	100-epoch critic-led warm-up; at most 800 total; patience 25	Trace-balanced actor/critic BCE
Raw Stage 2	actor 10^{-4}; critic 10^{-3}	5:1 updates; at most 400 cycles; minimum/patience 25/25	Frozen encoder; one initial critic update
CLR Stage 2	actor 10^{-4}; critic 10^{-3}	10:1 updates; at most 200 cycles; minimum/patience 13/13	Frozen encoder; one initial critic update
Checkpoint	–	Both Stage 2 schedules allow at most 2,000 actor updates	Null mean factor ≤ 1.05, then maximum alternative growth
Correction	–	$\kappa = 0, .0025, \ldots, .10$	13 geometrically spaced α values from .01 to .25

Table 3: Optimization and selection settings for recurrent monitor components. All use hidden size 16, 20% validation, AdamW weight decay 10^{-4}, and gradient clipping at 1. Learned betting and actor–critic optimization use full padded development tensors, not conditional-mean minibatches.

within-trace averaging followed by equal class averaging, the warm-start objective is

$$\mathcal{L}_{\text{warm}} = \text{BCE}_{\text{tr}}(a_{i,t}, Y_i) + \text{BCE}_{\text{tr}}(v_{i,t}, Y_i).$$

During its critic-led warm-up, the actor route into the encoder is detached. For both routes, Stage 2 freezes the encoder, takes one initial critic update, and alternates actor updates with critic repair. Checkpoints must have held-out, trace-balanced null mean factor at most 1.05, after which alternative growth is maximized; if none is feasible, the smallest null violation is retained.

Deployment correction. Post-fit correction freezes the model and subtracts a validation-selected κ from every active local log-factor; the direct statistic uses $\kappa = 0$. The smallest candidate satisfying every validation-null Ville-family target is selected, or the candidate with the smallest worst excess if none is feasible. This empirical correction does not retroactively certify the learned factors.

C.3 Training and Evaluation Protocol

All recurrent learned components use hidden size 16, reserve 20% of eligible fitting IDs for validation, and use AdamW with weight decay 10^{-4} and gradient clipping at 1. The remaining settings are:

Model selection never uses deployment traces. Direct Ville/RandVille models may pool primary and auxiliary development partitions before their internal training/validation split; PAC fitting follows the partitioning above.

Completed artifacts store configurations, seeds, source hashes, split IDs, checkpoints, and thresholds. Fitting and replay use PyTorch, NumPy, SciPy, pandas, and scikit-learn; the conditional-mean GRU runs on CPU. Generator, scorer, engine, and residual-probe inference occurs during source preparation, not per monitor trial. These are locked settings, not per-task sweeps. Hardware and wall-clock metadata were not recorded uniformly, so runtime measurements are not aggregated.

We use the FPR, power, and censored nARL definitions in Eq. 3 and the surrounding main-text discussion, with alarms and censoring restricted to each active horizon. Metrics are computed within each trial and equally averaged; uncertainties are one population SD across the ten real-data trial values (20 repetitions for the synthetic experiment). Scorer AUROC, stepwise means, state recovery, learned-reference error, Ville/RandVille FPR, and actor–critic results before or after correction remain empirical diagnostics on the evaluated sources.

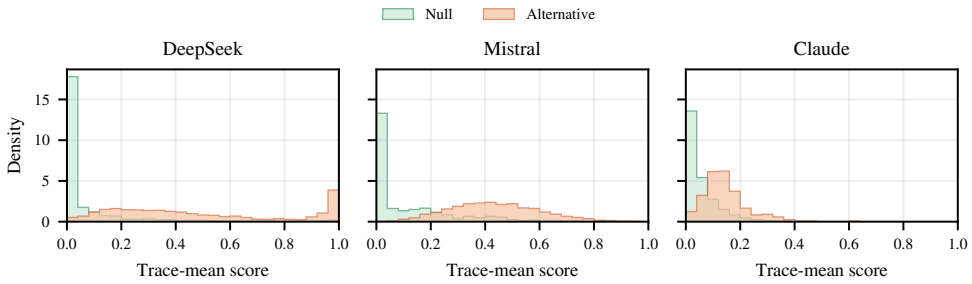
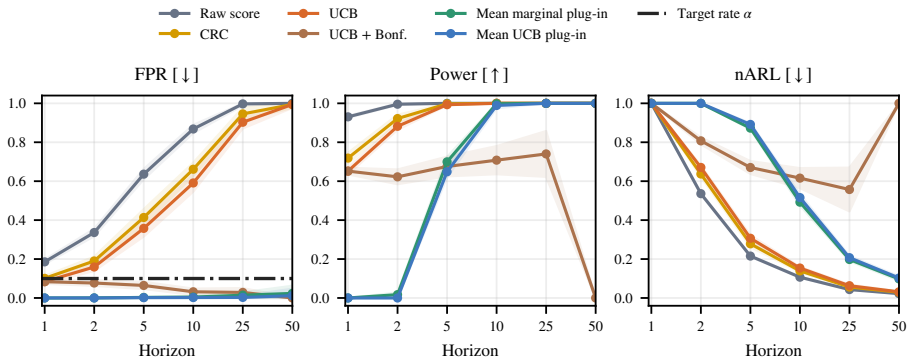


Figure 5: **Across-trace score populations.** Null and alternative distributions of completed-trace mean Qwen2.5-PRM scores for MATH solutions generated by DeepSeekMath, Mistral, and Claude. Collapsing each solution to one scalar removes its within-trace history from the sequential unit; the remaining class separation contextualizes the power differences in the horizon and target-rate sweeps below.

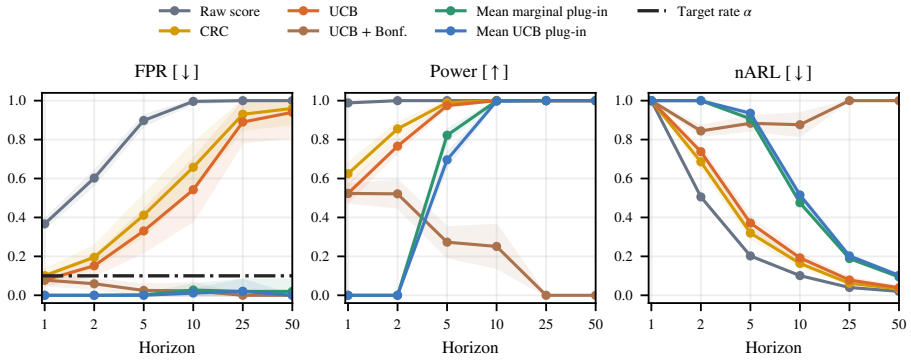
D Additional Experimental Results

D.1 Across-Trace Monitoring

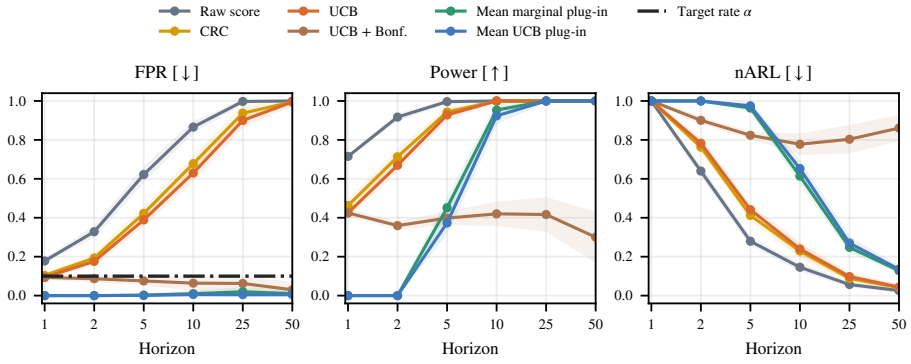
These results expand the across-trace side of Fig. 2. Each completed reasoning trace is reduced to its mean Qwen2.5-PRM score and becomes one item in the deployment stream. Fig. 5 shows the scalar problem that remains after this reduction. The horizon and target-rate comparisons in Fig. 6 and Fig. 7 extend the main-text Mistral result to DeepSeekMath and Claude, separating thresholds calibrated for one atomic decision from evidence designed for repeated accumulation.



(a) DeepSeekMath-7B-Instruct.

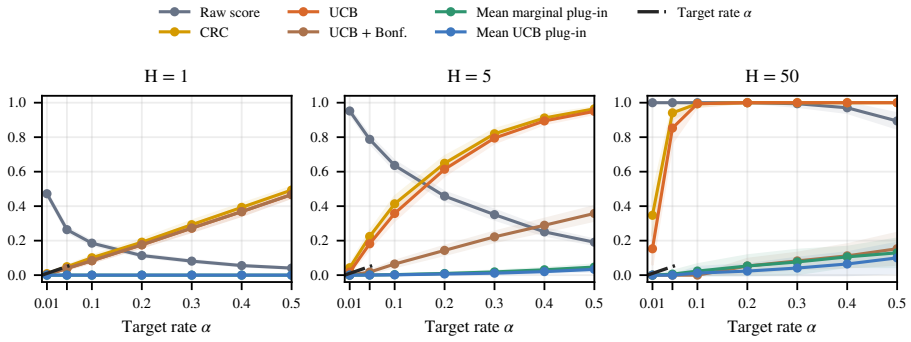


(b) Mistral-7B-Instruct-v0.3.

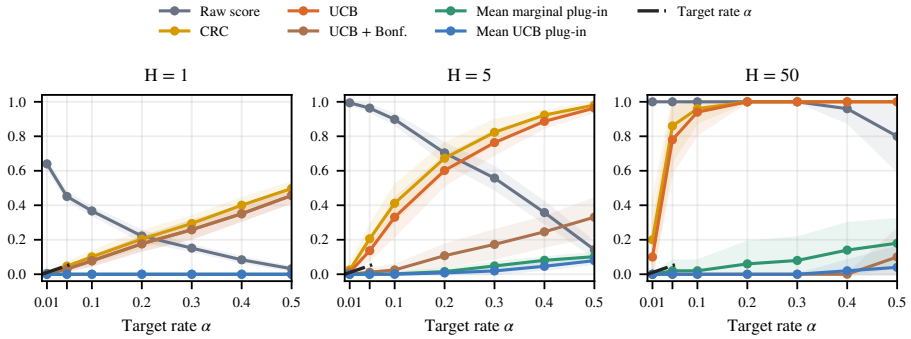


(c) Claude Haiku 4.5.

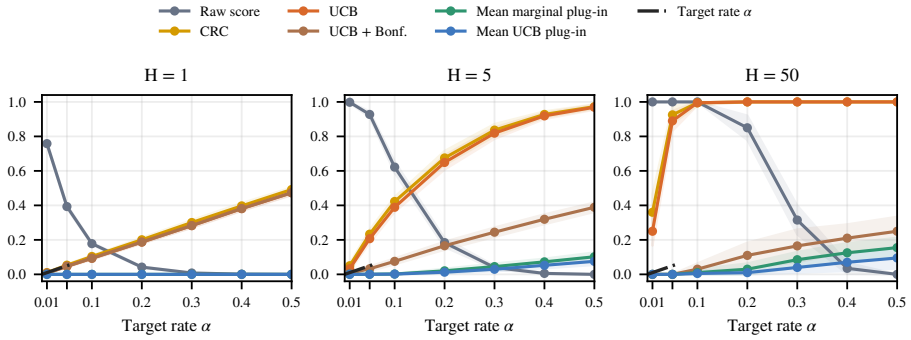
Figure 6: **Across-trace horizon sensitivity.** FPR on all-null streams and power and nARL on all-alternative streams as the deployment horizon grows at $\alpha = 0.10$. Repeated atomic CRC/UCB decisions accumulate false alarms, whereas trace-mean evidence targets the complete stream; fixed-horizon Bonferroni correction controls accumulation at a cost in power. Rows show MATH solutions scored by Qwen2.5-PRM. Curves are means ± 1 SD across ten trials.



(a) DeepSeekMath-7B-Instruct.



(b) Mistral-7B-Instruct-v0.3.



(c) Claude Haiku 4.5.

Figure 7: **Across-trace target-rate sensitivity.** Empirical FPR across the target-rate grid at horizons $H \in \{1, 5, 50\}$. The three reasoning sources reproduce the distinction in Fig. 2: atomic decision errors accumulate with repeated use, while compounded trace-mean evidence is comparatively stable as the stream grows. Rows show MATH solutions scored by Qwen2.5-PRM. Curves are means ± 1 SD across ten trials.

D.2 WITHIN-TRACE REASONING

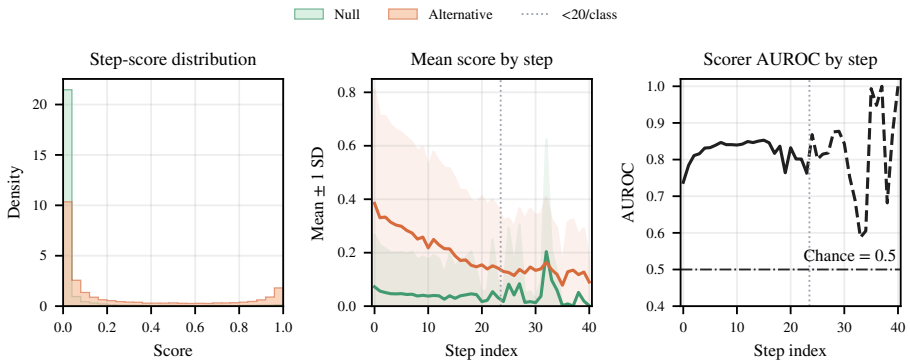
The within-trace results decompose the summary in Fig. 3 by the components of the monitoring problem. We first compare Qwen2.5-PRM score separation and null drift across the three generators, then evaluate fixed-decision baselines, conditional-mean estimation, predictable betting, and actor–critic normalization. Reading each figure by row exposes how the same design behaves under the broad class support of DeepSeekMath, the paired-score Mistral setting, and the smaller alternative class of Claude.

Score regimes and calibrated baselines. The scorer diagnostics in Fig. 8 contextualize the power and normalization behavior in the subsequent results. Fig. 9 then compares direct scores, CRC, Bentkus-UCB, binomial control, and E-valuator with PAC calibration. The raw score is an uncalibrated comparator; the remaining methods calibrate a fixed decision rather than local evidence.

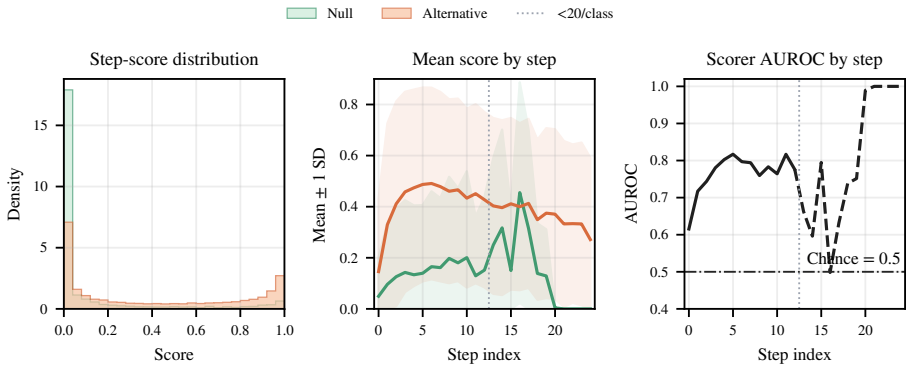
Conditional-mean estimation and betting. Fig. 10 holds the learned GRU reference fixed and varies only the predictable betting rule. The RandVille comparisons in Fig. 11 make residual normalization error more visible than ordinary Ville crossing. Together, they distinguish gains from selecting a stronger bet from gains due to learning a history-dependent reference.

Actor–critic normalization and correction. Fig. 12 compares raw-score and CLR initializations before and after empirical post-fit correction. Consistent with the main results, CLR provides the stronger power initialization across the reasoning sources. Correction shows the empirical FPR–power cost of adding validation-selected slack, but does not supply a history-uniform normalization bound.

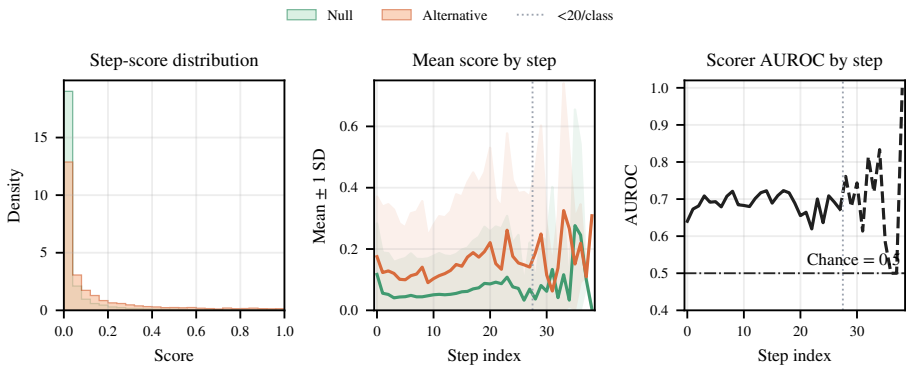
Calibration-size sensitivity. We vary the nested calibration pool while retaining the original deployment split and all other monitor settings. The four pools contain $n \in \{500, 1{,}000, 2{,}000, 3{,}000\}$ traces (10%, 20%, 40%, and 60% of the source); each is divided evenly into fitting-calibration and auxiliary subsets of $n \in \{250, 500, 1{,}000, 1{,}500\}$ traces. The deployment split remains fixed at $n = 2{,}000$ traces across all settings. Fig. 13 is an empirical finite-sample sensitivity diagnostic: changing the pool size shifts both observed FPR and its trial variation, with the direction and magnitude of those shifts differing across monitors.



(a) DeepSeekMath-7B-Instruct.

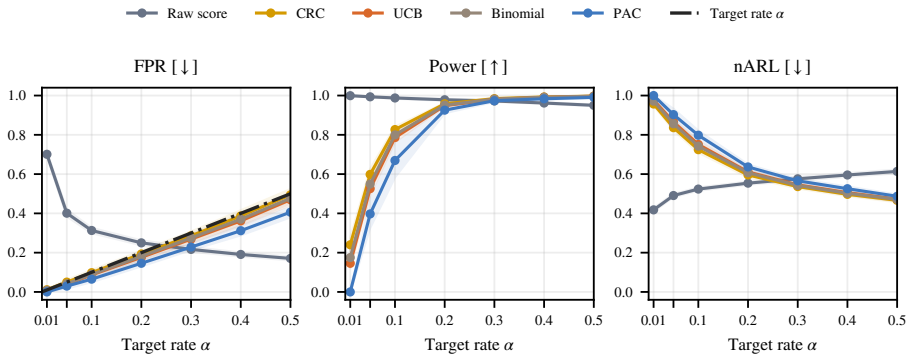


(b) Mistral-7B-Instruct-v0.3.

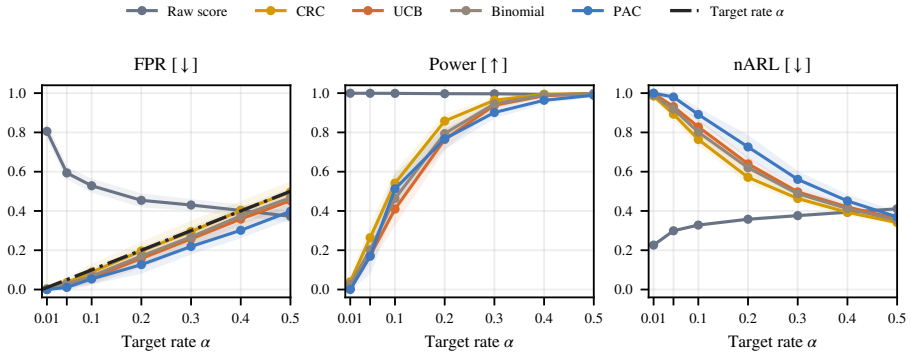


(c) Claude Haiku 4.5.

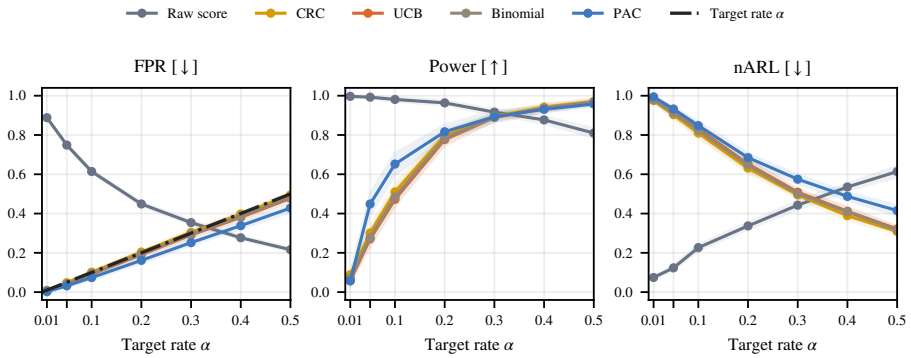
Figure 8: **Reasoning-score diagnostics.** Qwen2.5-PRM score distributions, class-conditional mean scores over the trace, and stepwise AUROC for the three MATH generators. DeepSeekMath provides substantial support in both outcome classes, whereas Claude has comparatively few incorrect alternatives. Dotted tails mark low-support steps beyond the reliable monitoring range.



(a) DeepSeekMath-7B-Instruct.

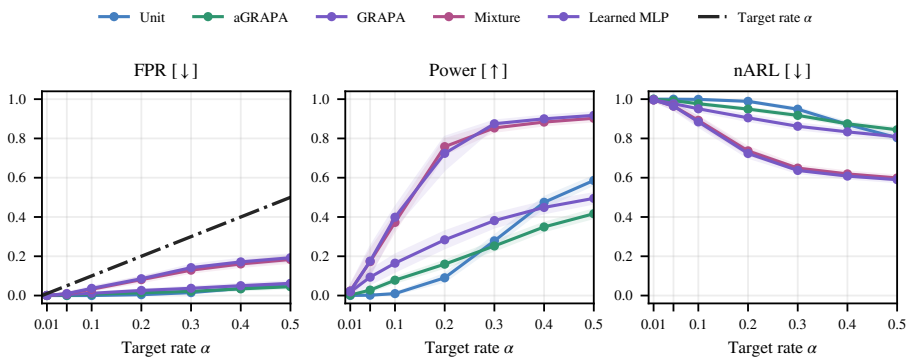


(b) Mistral-7B-Instruct-v0.3.

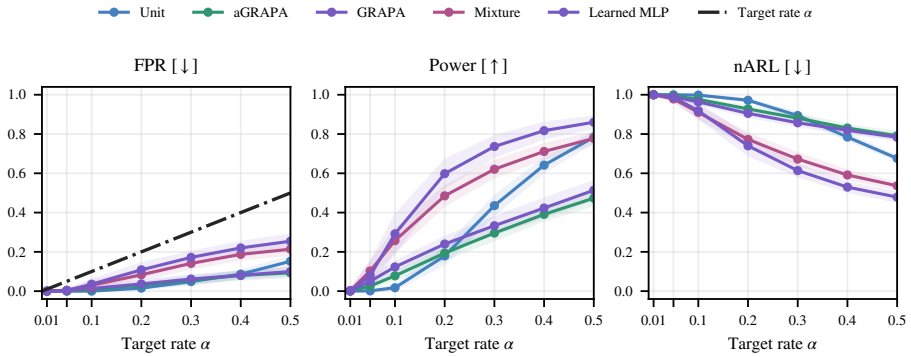


(c) Claude Haiku 4.5.

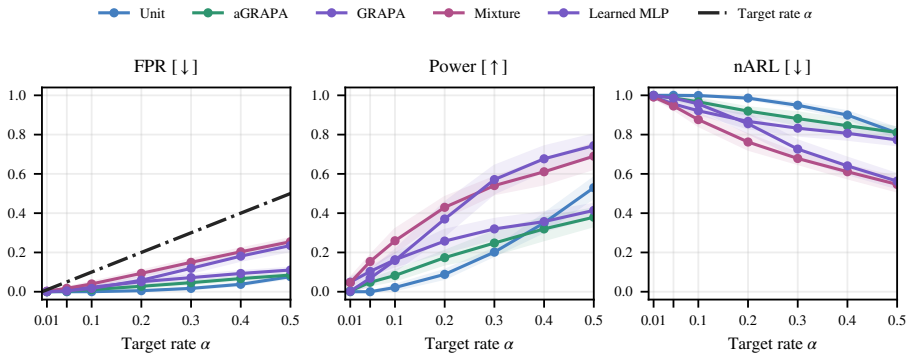
Figure 9: **Within-trace calibrated baselines.** FPR, power, and nARL for direct Qwen2.5-PRM scores, CRC, Bentkus-UCB, binomial risk control, and E-valuator with PAC calibration. The raw score provides an uncalibrated comparator; the remaining methods calibrate the fixed alarm event and remain effective despite history-dependent score behavior, complementing the evidence-normalization results below. Curves are means ± 1 SD across ten trials.



(a) DeepSeekMath-7B-Instruct.

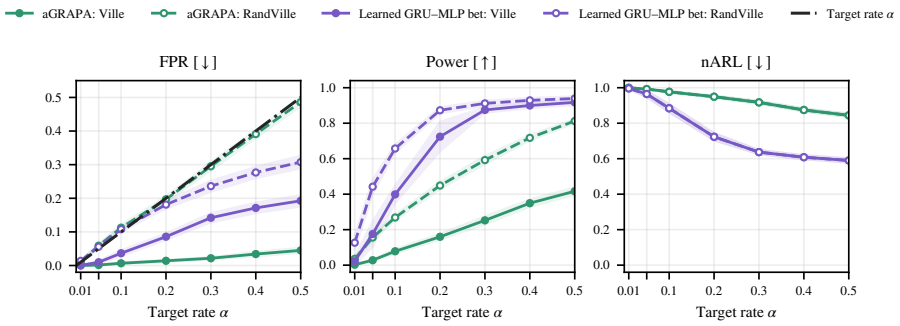


(b) Mistral-7B-Instruct-v0.3.

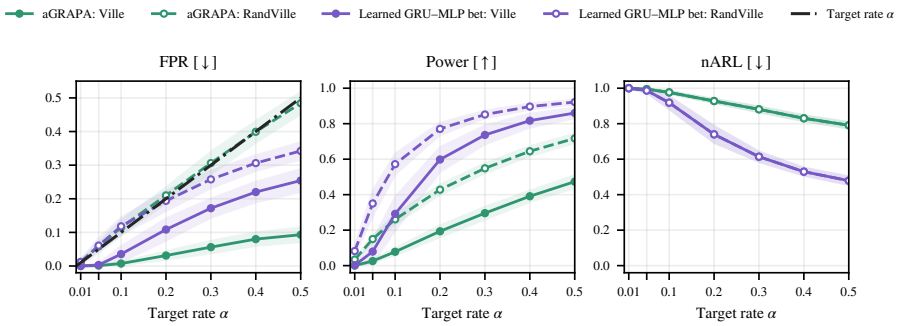


(c) Claude Haiku 4.5.

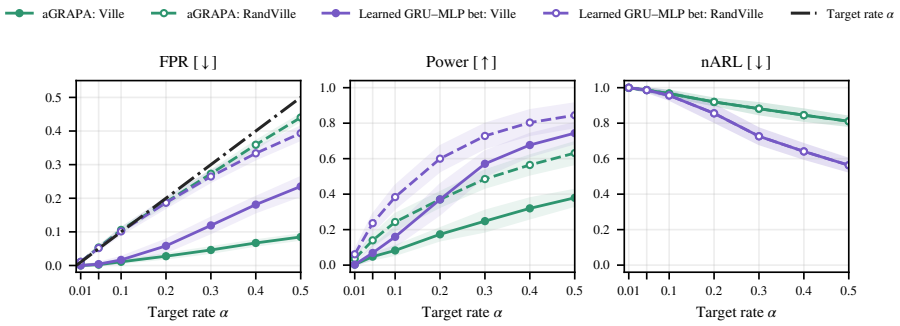
Figure 10: **Betting-rule ablation under a shared learned reference.** Unit, aGRAPA, GRAPA, mixture, and learned GRU–MLP bets are compared under ordinary Ville crossing while the fitted history-dependent reference is held fixed. Bet selection changes power and nARL, but cannot by itself repair reference misspecification. Curves are means ± 1 SD across ten trials.



(a) DeepSeekMath-7B-Instruct.

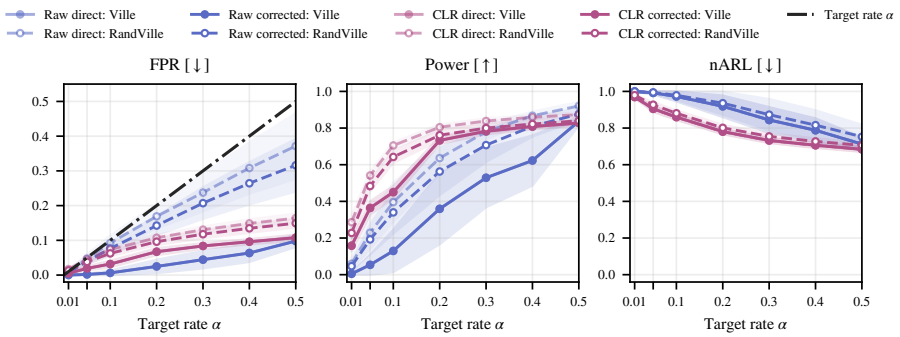


(b) Mistral-7B-Instruct-v0.3.

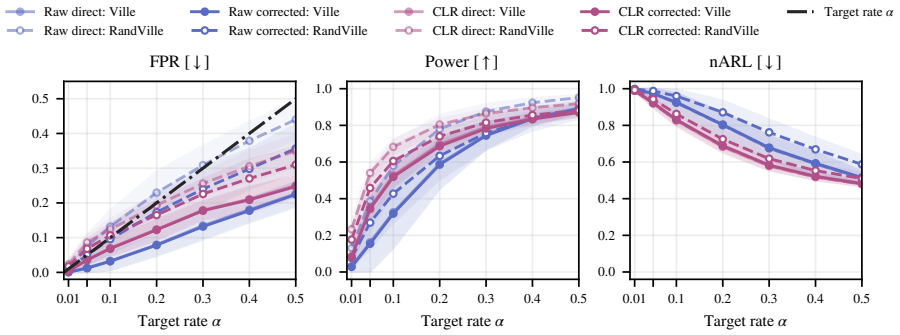


(c) Claude Haiku 4.5.

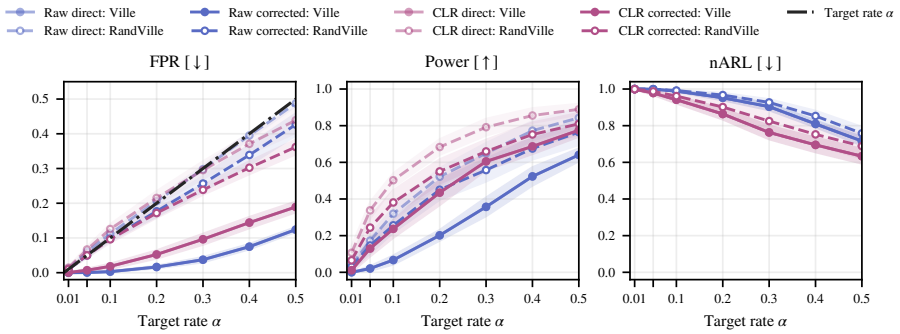
Figure 11: **Learned conditional-mean monitoring.** aGRAPA and learned GRU–MLP betting are evaluated with Ville and RandVille across the three reasoning sources. The more sensitive randomized threshold exposes residual normalization error that conservative ordinary crossing can conceal; these learned-reference curves are therefore empirical diagnostics rather than conditional-validity certificates. Curves are means ± 1 SD across ten trials.



(a) DeepSeekMath-7B-Instruct.



(b) Mistral-7B-Instruct-v0.3.



(c) Claude Haiku 4.5.

Figure 12: **Actor–critic initialization and post-fit correction.** Raw-score and CLR initializations are compared before and after validation-selected correction under Ville and RandVille. CLR consistently provides the stronger power initialization, while correction exposes the empirical trade-off between reducing accumulated null drift and retaining power. These operating points do not establish history-uniform normalization. Curves are means ± 1 SD across ten trials.

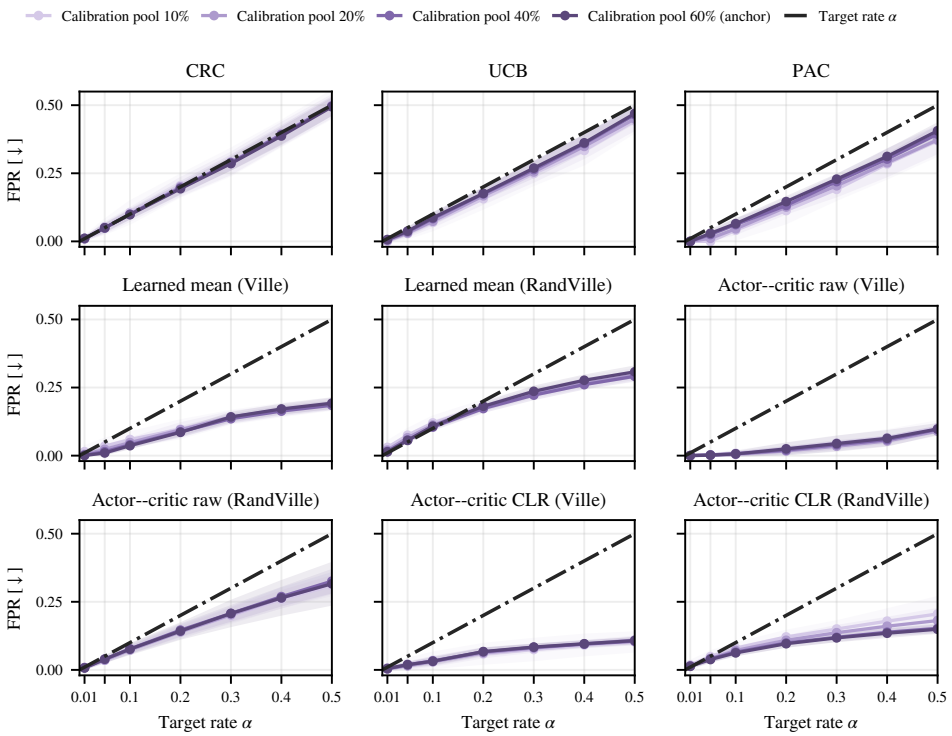
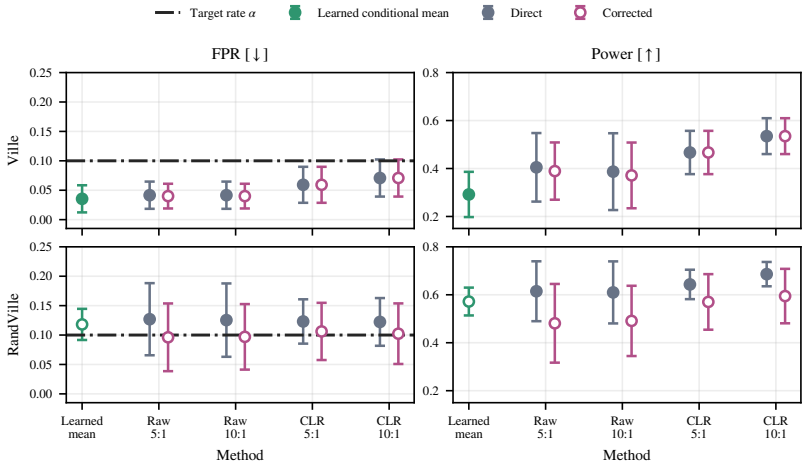


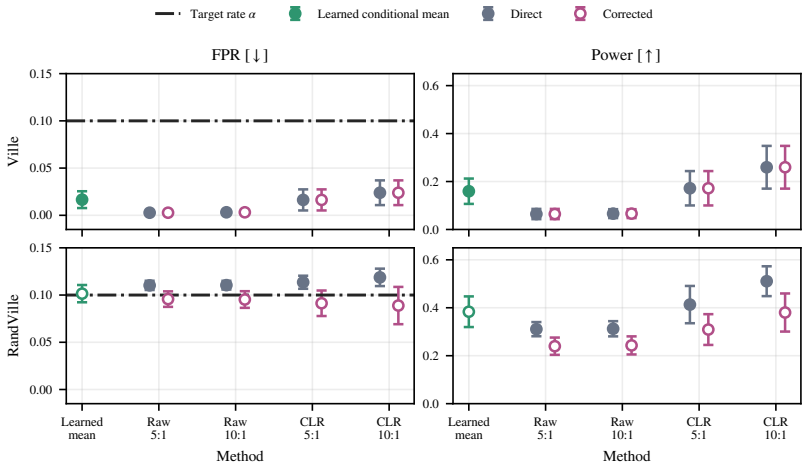
Figure 13: **Calibration-size sensitivity.** Empirical FPR across the target-rate grid for DeepSeekMath MATH reasoning scored by Qwen2.5-PRM. Each panel fixes one monitor while colored curves vary the nested calibration pool over $n \in \{500, 1{,}000, 2{,}000, 3{,}000\}$ traces, divided evenly between fitting-calibration and auxiliary subsets. Pool size changes both observed FPR and trial variation, with monitor-dependent direction and magnitude. Curves are means ± 1 SD across ten trials; the diagonal marks the target rate. Learned-monitor results are empirical finite-sample diagnostics.

D.3 Focused Actor–Critic Correction Diagnostics

Fig. 14 expands the actor–critic result in Fig. 3 at $\alpha = 0.10$, comparing both initializations under 5:1 and 10:1 actor-to-critic update schedules. Direct and corrected results are empirical operating points: correction can reduce null drift, but does not supply a history-uniform normalization bound.



(a) Mistral-7B-Instruct-v0.3.



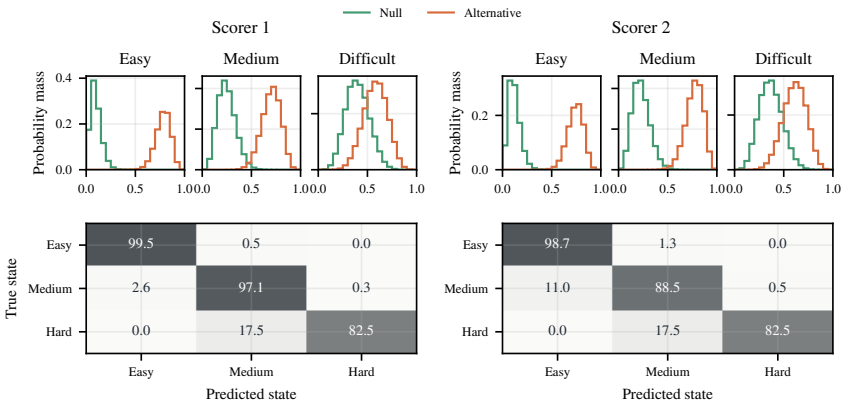
(b) Claude Haiku 4.5.

Figure 14: **Focused actor–critic correction at $\alpha = 0.10$.** The learned conditional mean and raw-score and CLR-initialized actor–critic models are compared under 5:1 and 10:1 update schedules, before and after post-fit correction. Results are empirical means ± 1 SD across ten trials.

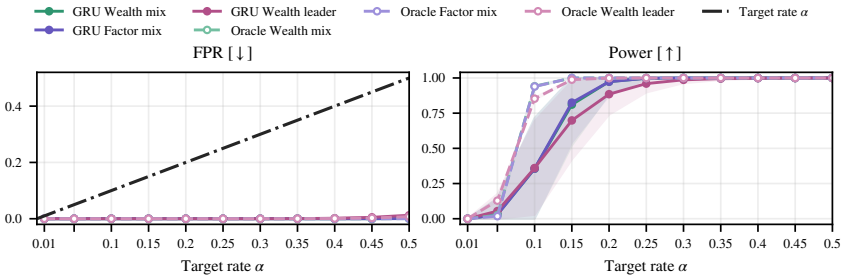
D.4 Sufficiency and Composition

Fig. 15 separates the structural composition result from its learned approximation. Oracle conditional means establish locally normalized components, whereas state recovery and composition with fitted GRU references remain empirical diagnostics.

Fig. 16 uses aligned Qwen2.5-PRM and Math-Shepherd scores on Mistral traces. Evidence mixtures can exploit complementary components, while PAC thresholds must be recalibrated after the aggregate statistic or routing policy changes.

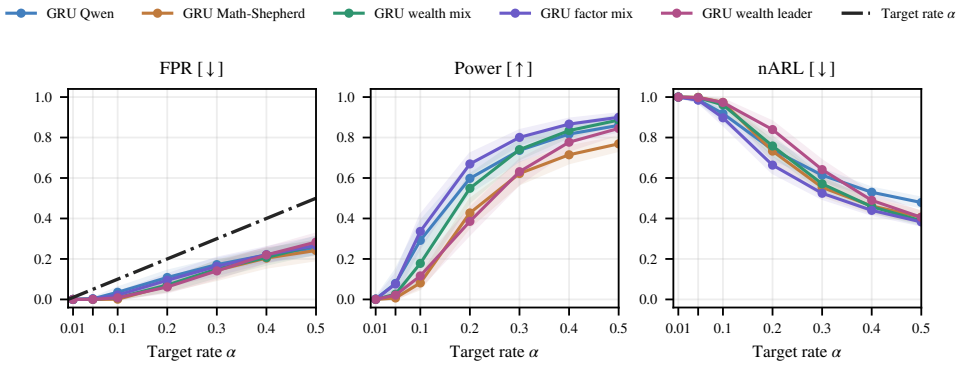


(a) State recovery for the two synthetic scorers.

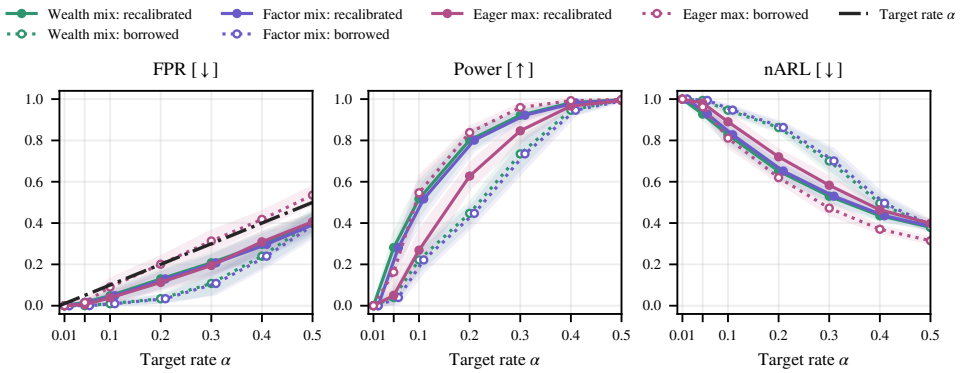


(b) Oracle and learned evidence composition.

Figure 15: **Controlled sufficient-state recovery and composition.** *Top:* Statewise score laws and GRU confusion matrices evaluate recovery of three predictable difficulty states. *Bottom:* Oracle conditional means establish locally normalized composition; fitted GRU references evaluate the same policies empirically. Composition curves are means ± 1 SD across 20 repetitions.



(a) Composition of learned evidence factors.



(b) PAC calibration after changing the aggregate statistic.

Figure 16: **Real-data evidence composition and PAC recalibration.** Aligned Mistral reasoning traces receive Qwen2.5-PRM and Math-Shepherd scores. *Top:* Wealth and factor mixtures and predictable wealth-leader routing can exploit complementary learned evidence and improve power over either component. *Bottom:* Fresh thresholds calibrate each fixed PAC aggregate separately, whereas borrowed mean or componentwise thresholds can lose power through conservatism or exceed the target FPR. The comparison illustrates the main-text distinction between composing locally normalized evidence and recalibrating a changed final decision statistic. Curves are means ± 1 SD across ten trials.

D.5 Other Monitoring Domains

Tab. 4 and Tab. 5 complement the main-text RandVille results with ordinary Ville crossing and fixed-decision baselines. Because these methods calibrate different statistical objects, power should be read jointly with empirical FPR.

Table 4: **Cross-domain monitoring under Ville crossing.** Empirical FPR and power at $\alpha = 0.20$; entries are means ± 1 SD across ten trials.

Domain	Cond. Mean		Actor–Critic (Raw)		Actor–Critic (CLR)	
	FPR (↓)	Power (↑)	FPR (↓)	Power (↑)	FPR (↓)	Power (↑)
Interaction Tasks						
Red Teaming	0.01± 0.01	0.08± 0.07	0.04± 0.03	0.19± 0.13	0.08± 0.01	0.33± 0.01
Chess	0.13± 0.03	0.53± 0.07	0.12± 0.02	0.61± 0.07	0.09± 0.05	0.40± 0.16
Generation Tasks						
MATH	0.05± 0.01	0.28± 0.06	0.03± 0.01	0.22± 0.02	0.08± 0.01	0.39± 0.03
Coding (MBPP)	0.00± 0.00	0.00± 0.00	0.02± 0.03	0.20± 0.17	0.10± 0.02	0.64± 0.06
Agentic Tasks						
ALFWorld	0.13± 0.02	0.99± 0.00	0.06± 0.01	0.95± 0.01	0.18± 0.02	0.93± 0.02
DBBench	0.01± 0.00	0.08± 0.01	0.00± 0.00	0.01± 0.01	0.12± 0.01	0.70± 0.01
KnowledgeGraph	0.10± 0.02	0.78± 0.02	0.11± 0.07	0.71± 0.07	0.17± 0.07	0.78± 0.08
WebShop	0.00± 0.00	0.04± 0.02	0.04± 0.02	0.24± 0.06	0.08± 0.02	0.30± 0.03

Table 5: **Cross-domain calibrated baselines.** Empirical FPR and power at $\alpha = 0.20$; entries are means ± 1 SD across ten trials and shaded FPR cells exceed the target.

Domain	CRC		UCB		PAC	
	FPR (↓)	Power (↑)	FPR (↓)	Power (↑)	FPR (↓)	Power (↑)
Interaction Tasks						
Red Teaming	0.20± 0.01	0.56± 0.01	0.19± 0.01	0.54± 0.01	0.17± 0.01	0.50± 0.01
Chess	0.20± 0.02	0.78± 0.03	0.17± 0.03	0.74± 0.03	0.12± 0.03	0.43± 0.05
Generation Tasks						
MATH	0.20± 0.01	0.58± 0.02	0.18± 0.01	0.56± 0.02	0.15± 0.01	0.59± 0.03
Coding (MBPP)	0.21± 0.01	0.90± 0.02	0.19± 0.01	0.90± 0.02	0.17± 0.02	0.92± 0.02
Agentic Tasks						
ALFWorld	0.20± 0.02	0.99± 0.00	0.18± 0.02	0.98± 0.00	0.15± 0.02	0.99± 0.01
DBBench	0.19± 0.01	0.77± 0.01	0.18± 0.01	0.76± 0.01	0.15± 0.02	0.73± 0.02
KnowledgeGraph	0.30± 0.02	0.88± 0.01	0.24± 0.03	0.86± 0.02	0.12± 0.04	0.71± 0.07
WebShop	0.21± 0.02	0.44± 0.01	0.19± 0.02	0.42± 0.01	0.16± 0.02	0.42± 0.02

Fig. 17 relates these outcomes to score separation, temporal drift, and late-step null support. KnowledgeGraph remains discriminative, but its rising null scores and limited support make conditional normalization difficult to estimate; PAC instead calibrates the final event and remains below the target.

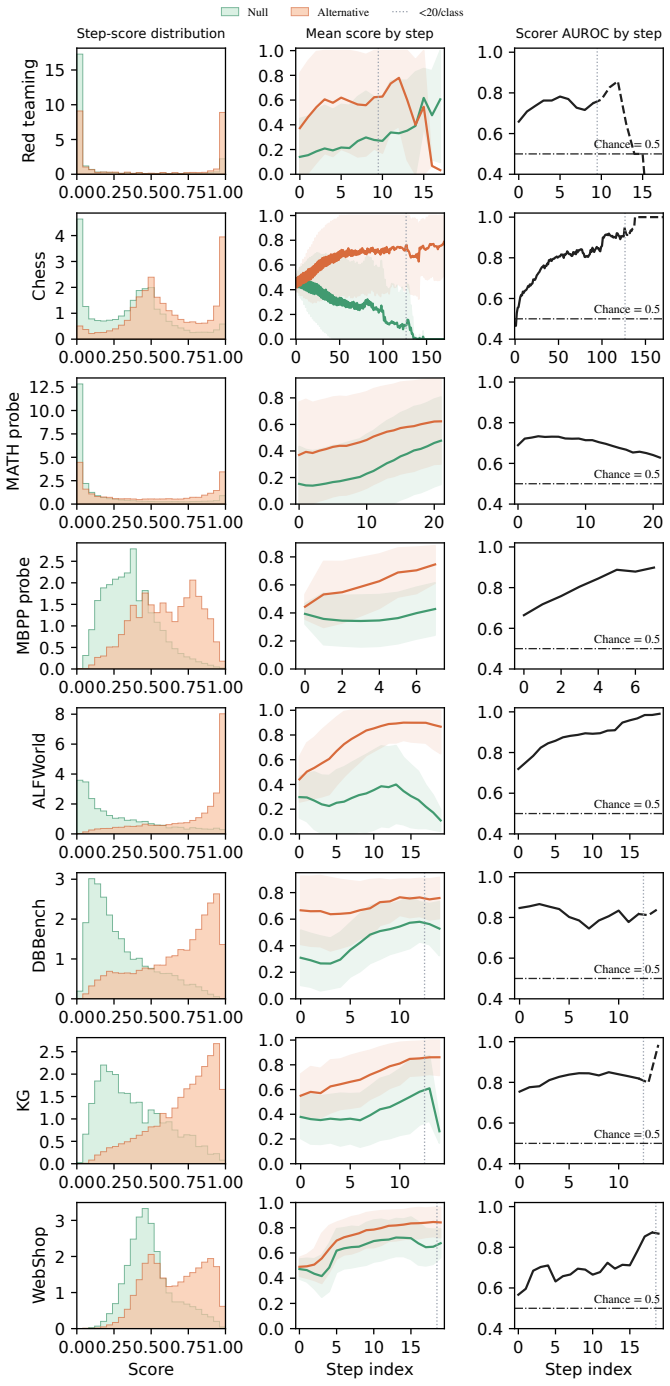


Figure 17: **Cross-domain scorer diagnostics.** The eight rows correspond to the interaction, generation, and agentic targets in the result tables. Each row shows score distributions, class-conditional mean scores over the trace, and stepwise AUROC; dotted tails mark low-support steps. Differences in class separation, temporal drift, and late-step null support contextualize the task-level power and normalization behavior reported in the main text. In particular, KnowledgeGraph remains discriminative while its rising null scores and limited late-step support make history-dependent normalization difficult to estimate.