Certified Self-Consistency: Statistical Guarantees and Test-Time Training for Reliable Reasoning in LLMs

P. Cordero-Encinar and A. B. Duncan

Department of Mathematics, Imperial College London, UK.
{*paula.cordero-encinar22, a.duncan*}@*imperial.ac.uk*

Website **Code**

Abstract

Recent advances such as self-consistency and test-time reinforcement learning (TTRL) improve the reliability of large language models (LLMs) without additional supervision, yet their underlying mechanisms and statistical guarantees remain poorly understood. We present a unified framework for certifiable inference in LLMs, showing that majority voting provides a statistical certificate of self-consistency: under mild assumptions, the aggregated answer coincides with the mode of the model's terminal distribution with high probability. We derive finite-sample and anytime-valid concentration bounds that quantify this confidence, and introduce the Martingale Majority Certificate (MMC), a sequential stopping rule that adaptively determines when sufficient samples have been drawn. We further prove that label-free post-training methods such as TTRL implicitly sharpen the answer distribution by exponentially tilting it toward its mode, thereby reducing the number of samples required for certification. Building on this insight, we propose new post-training objectives that explicitly optimise this trade-off between sharpness and bias. Together, these results explain and connect two central test-time scaling strategies, self-consistency and TTRL, within a single statistical framework for label-free, certifiable reliability in reasoning LLMs.

1 Introduction

Large language models (LLMs) have demonstrated striking performance across a range of reasoning tasks, from mathematical problem solving to code generation (Brown et al., 2020; OpenAI, 2023). A key advance has been *chain-of-thought* (CoT) prompting, which guides models to produce explicit intermediate steps before returning a final answer (Wei et al., 2022; Kojima et al., 2022). CoT substantially improves accuracy on problem-solving benchmarks (Cobbe et al., 2021; Lewkowycz et al., 2022). The quality of CoT reasoning depends strongly on the decoding strategy adopted at inference time. Deterministic decoding (e.g. greedy or low-temperature sampling) yields a single trajectory but limits exploration, often causing the model to commit early to an incorrect reasoning path. By contrast, stochastic decoding methods such as nucleus or temperature sampling encourage diversity over reasoning paths, revealing alternative chains of thought that may reach the correct solution. This is exploited in *test-time scaling* strategies, which seek to improve the reliability and accuracy of model responses at inference time by exploring and aggregating information through multiple rollouts of the model. While this requires more compute at test time, such approaches demonstrably improve performance without the need for retraining (Yao et al., 2023; Besta et al., 2024), particularly for small-footprint models (Chan et al., 2025). In the context of LLMs, a wide range of test-time scaling approaches have emerged, ranging from sampling and aggregation approaches (e.g. majority voting, self-consistency); trajectory extension approaches, which force longer rollouts to encourage more complete reasoning (e.g. budget forcing, multi-hop reasoning); or search-and-exploration based approaches, which systematically explore multiple reasoning branches through search (e.g. Tree of Thoughts, MCST). If available, a verifier (e.g. a proof checker or an external model), can be used to guide these strategies towards higher quality outputs.

We can formalise an LLM rollout as a stochastic decoding process

$$(Y_t)_{t\geq 0}, \quad Y_t \in \mathcal{V},$$

where $\mathcal{V}$ is the vocabulary and the process is initialised by a prompt pr. At each step the model samples

$$Y_t \sim \pi_\phi(\cdot \mid Y_{<t}, pr),$$

from a conditional policy parametrised by weights ϕ. The *thinking phase* consists of the random evolution of this sequence until a termination token is produced, at which point the model emits the response, starting from a random stopping time τ. We denote by

$$X := g(Y_{\tau:}) \in \mathcal{A}$$

the canonicalised terminal answer, obtained by applying a deterministic extraction map g. The induced terminal distribution $\mathbf{p} = \mathrm{Law}(X)$ over the answer set $\mathcal{A}$ captures the model's epistemic uncertainty about its own final output. In an ideal reasoning model, we would like rollouts to exhibit rich variability in $Y_{1:\tau-1}$ (the reasoning trajectories), yet concentrate mass in the final answer X (the outcome). That is, we seek diversity over reasoning paths, but consistency over terminal responses.

In supervised or verifier-equipped settings, correctness can be externally validated. In open-ended reasoning tasks, such supervision is unavailable. In the absence of external rewards, a model must act relative to its own uncertainty. Letting $a \in \mathcal{A}$ denote the chosen output and $X \sim \mathbf{p}$ the stochastic model response, the expected 0–1 loss is $\mathbb{E}[1\{a \neq X\}]$. The Bayes-optimal decision minimising this loss is the mode

$$c^\star = \arg\max_j p_j,$$

which corresponds to the model's most probable self-consistent answer. Hence, under symmetric loss, recovering the mode is the optimal *model-relative* prediction. When a verifier is absent, certifying that a model's reported answer coincides with this mode provides a natural measure of reliability.

Statistical certificates of self-consistency. In practice, the terminal probabilities $\mathbf{p}$ are unknown and can be estimated only through multiple independent rollouts $X_1, \dots, X_n$. The simplest estimator of the mode is the *majority vote*

$$\widehat{c}_n := \arg\max_j \hat{p}_{n,j}, \qquad \hat{p}_{n,j} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}\{X_i = j\}.$$

This estimator forms the basis of *self-consistency* test-time scaling (Wang et al., 2022; Madaan et al., 2023), which has been shown to stabilise CoT reasoning and improve benchmark accuracy (Anil et al., 2023; Zuo et al., 2025). From a statistical standpoint, majority voting is the Bayes-optimal estimator of $c^\star$ under 0–1 loss, and an associated upper bound on $\mathbb{P}[\widehat{c}_n \neq c^*]$ provides a *statistical certificate of self-consistency*: a quantitative guarantee that the aggregated answer coincides with the mode of the terminal law $\mathbf{p}$ with high probability. Under standard regularity conditions (e.g. conditional independence of rollouts and a unique mode of $\mathbf{p}$), the majority-vote estimator is consistent, satisfying $\Pr[\widehat{c}_n = c^\star] \to 1$ as $n \to \infty$. A more practical question concerns the finite-sample regime: how large must n be to guarantee, with confidence $1 - \varepsilon$, that $\widehat{c}_n$ already equals $c^\star$?

To address this, we derive a hierarchy of finite-sample and asymptotic certificates, leveraging Hoeffding, Bernstein, Chernoff–Markov, and Sanov concentration bounds for the error probability $\mathbb{P}[\widehat{c}_n \neq c^\star]$. Although not tight in the small-sample regime, these bounds clarify how reliability scales with the ensemble size and with the *mode margin* $\delta = p_{c^\star} - p_{j^\star}$, i.e. the gap between the top two answer probabilities.

If the probabilities p_j were known, one could invert these bounds to determine the number of samples required to achieve a desired confidence $1 - \varepsilon$. In reality, both p_j and δ must be estimated on the fly. This motivates a *sequential* formulation: as rollouts arrive, can we determine adaptively when the current majority is statistically reliable? We introduce the *Martingale Majority Certificate (MMC)*, a sequential procedure based on e-values and Ville's inequality (Ville, 1939; Howard et al., 2021), which adaptively tests whether the empirical leader remains significantly ahead of its nearest rival and of all others combined. This guarantees that at the (random) stopping time τ,

$$\Pr[\widehat{c}_{n_\tau} \neq c^\star] \leq \varepsilon,$$

thus providing an *anytime-valid certificate* of model self-consistency.

Why test-time training helps. Recent work on label-free post-training, such as *test-time reinforcement learning* (TTRL), adapts model parameters online by optimising KL-regularised objectives with respect to its own rollouts (Zuo et al., 2025; Akyürek et al., 2025). These methods empirically improve reliability but their mechanism remains opaque. We show that such objectives correspond to an *exponential tilting* of the terminal law $\mathbf{p}$, yielding a sharpened distribution more concentrated around its mode. This transformation increases the mode margin, improving the signal-to-noise ratio of the margin random variable $\Delta_{j^\star} = \mathbf{1}\{X = c^\star\} - \mathbf{1}\{X = j^\star\}$, and thereby reducing the number of samples required for certification. However, it also introduces a controlled bias relative to the original distribution, governed by the KL regularisation strength. Thus, TTRL provides a complementary lever: by reshaping $\mathbf{p}$ to enlarge δ, it lowers the compute required for reliable self-consistency.

Emergent calibration in reasoning models. Beyond the theoretical and algorithmic results, our experiments reveal a notable empirical regularity: the *signal-to-noise ratio* (SNR) of the margin variable $\Delta_{j^\star} = \mathbf{1}\{X = c^\star\} - \mathbf{1}\{X = j^\star\}$, which quantifies the sharpness of the model's terminal answer distribution, correlates strongly with external measures of problem difficulty (Figure 2c). Across the MATH-500 benchmark, harder problems exhibit systematically lower and more variable SNR values, while easier problems yield sharply peaked distributions concentrated around a single answer.

This behaviour is non-trivial: the model has no access to ground-truth difficulty labels, yet its own epistemic uncertainty, reflected in the variability of its rollouts, aligns closely with these labels. This suggests an *emergent form of calibration* in reasoning LLMs: without explicit supervision or external verification, models appear to "know when they do not know." In statistical terms, the SNR acts as a label-free proxy for epistemic uncertainty and, consequently, for task difficulty.

This observation links our theoretical framework to observable model behaviour. The same margin variable that governs finite-sample concentration and sequential certification (Sections 2–3) also provides a practical signal for compute-adaptive inference: when the SNR is low, additional rollouts or verifier checks can be triggered, whereas high-SNR cases can be certified with fewer samples. Hence, the SNR not only underpins the theory of certified self-consistency, but also yields a measurable and actionable indicator of reliability in reasoning models.

Our contributions. We develop a framework for *certifiable inference in chain-of-thought LLMs*, viewing majority voting as a statistical certificate for the terminal law $\mathbf{p} = \mathrm{Law}(X)$. Specifically:

1. **Finite-sample and asymptotic certificates.** We derive explicit Hoeffding, Bernstein, and Sanov-type bounds for $\mathbb{P}[\widehat{c}_n \neq c^\star]$, characterising how reliability improves with ensemble size as a function of the mode margin δ.
2. **Anytime-valid stopping certificates.** We propose the *Martingale Majority Certificate (MMC)*, a sequential test that adaptively determines when sufficient rollouts have been drawn, guaranteeing $\Pr[\widehat{c}_n \neq c^\star] \leq \varepsilon$ at stopping.
3. **Explaining test-time reinforcement learning.** We formalise the connection between KL-regularised TTRL objectives and exponential tilting of $\mathbf{p}$, explaining why these methods improve reliability by increasing the mode margin and thereby reducing the sample complexity for certification. Building on this insight, we introduce alternative post-training objectives optimising this trade-off between sharpness and bias.
4. **Empirical link between uncertainty and problem difficulty.** We show that the signal-to-noise ratio (SNR) of the margin variable $\Delta_{j^\star}$, which governs our statistical certificates, correlates strongly with externally defined difficulty levels, revealing an emergent form of calibration in reasoning LLMs.

Together, these results provide a principled strategy for *certifying that an LLM's output coincides with its own most probable prediction* through self-consistency. By linking concentration bounds, martingale stopping rules, and test-time reinforcement learning, we provide a unified statistical framework of when and why self-consistency yields certifiable reliability in reasoning models, and how test-time adaptation can further reduce the computational cost of this certification. Figure 1 summarises the components of our framework.

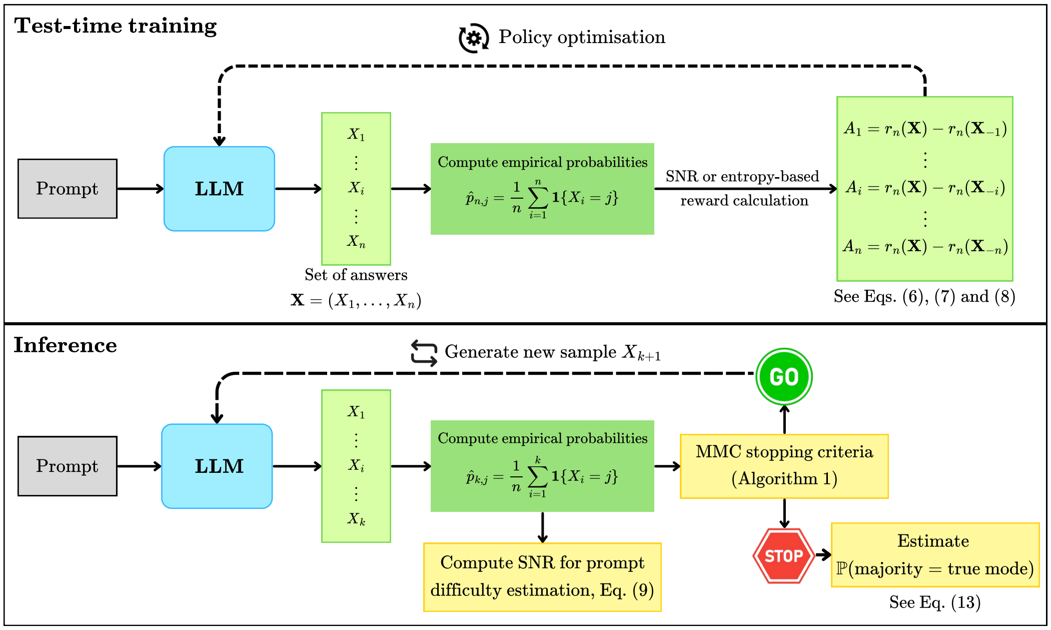

Test-time training
Policy optimisation
Prompt
LLM
X_1
$\vdots$
X_i
$\vdots$
X_n
Set of answers
$\mathbf{X} = (X_1, \dots, X_n)$
Compute empirical probabilities
$\hat{p}_{n,j} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}\{X_i = j\}$
SNR or entropy-based
reward calculation
$A_1 = r_n(\mathbf{X}) - r_n(\mathbf{X}_{-1})$
$\vdots$
$A_i = r_n(\mathbf{X}) - r_n(\mathbf{X}_{-i})$
$\vdots$
$A_n = r_n(\mathbf{X}) - r_n(\mathbf{X}_{-n})$
See Eqs. (6), (7) and (8)
Inference
Generate new sample X_{k+1}
GO
Prompt
LLM
X_1
$\vdots$
X_i
$\vdots$
X_k
Compute empirical probabilities
$\hat{p}_{k,j} = \frac{1}{n}\sum_{i=1}^{k} \mathbf{1}\{X_i = j\}$
MMC stopping criteria
(Algorithm 1)
Compute SNR for prompt
difficulty estimation, Eq. (9)
STOP
Estimate
$\mathbb{P}$(majority = true mode)
See Eq. (13)

Figure 1: **Overview of the proposed framework.** Given a prompt, the model generates multiple reasoning rollouts from the reference distribution $\pi_{\text{ref}}(\cdot|pr)$. The resulting terminal answers are aggregated via majority voting, viewed as mode estimation under sampling uncertainty. The Martingale Majority Certificate (MMC) monitors the empirical margin and provides an *anytime-valid* stopping rule for certification. Test-time training with SNR or entropy-based adaptation sharpens the terminal distribution, thereby increasing the signal-to-noise ratio (SNR) and reducing the number of samples required for certification.

2 Statistical guarantees for majority voting

It is well known that majority voting is *consistent*: under i.i.d. rollouts and a unique mode $p_{c^\star} > \max_{j \neq c^\star} p_j$, we have that $\widehat{c}_n \to c^\star$ a.s. as $n \to \infty$. This is a direct extension of Condorcet's original jury theory (de Condorcet, 1785) to the multi-class setting (List & Goodin, 2001). Our goal in this section is to quantify the error, $\mathbb{P}[\widehat{c}_n \neq c^\star]$, i.e. when the majority vote over n i.i.d. rollouts $X_1, \dots, X_n \sim \text{Cat}(\mathbf{p})$ fails to return the true mode $c^\star = \arg\max_j p_j$, and how this error scales with the ensemble size n.

Setting and scope. We analyse an oracle setting where the terminal answer distribution $\mathbf{p} = (p_1, \dots, p_k)$ is known. This isolates what drives certainty under majority aggregation, specifically, how the error $\Pr[\widehat{c}_n \neq c^\star]$ scales with the mode margin $\delta = p_{c^\star} - p_{j^\star}$, the variances σ_j^2 of the margin random variables $\Delta_j = \mathbf{1}\{X = \hat{c}\} - \mathbf{1}\{X = j\}$, and the signal-to-noise ratio of $\Delta_{j^\star}$. The resulting finite-sample bounds and asymptotic rates provide insight into the determinants of reliability, and form the basis of an operational certificate for inference in Section 3, where we demonstrate how $\mathbf{p}$ can be simultaneously inferred from rollouts. Throughout we assume i.i.d. rollouts (conditional on the prompt) and a unique mode $p_{c^\star} > \max_{j \neq c^\star} p_j$; violations (e.g., strong correlations or ties) weaken guarantees and are handled adaptively by MMC.

Figure 3 compares the main bounds below with empirical estimates; full proofs are deferred to Appendix A.

2.1 Exact error probability with oracle $\mathbf{p}$

When $\mathbf{p}$ is known, the error probability admits an exact multinomial expression.

Theorem 2.1 (Exact small-sample probability)**.** *For all $n \geq 1$,*

$$\Pr[\widehat{c}_n \neq c^\star] \;=\; \sum_{\substack{x \in \mathbb{N}^k \\ x_1 + \cdots + x_k = n \\ x_{c^\star} \leq \max_{j \neq c^\star} x_j}} \frac{n!}{x_1! \cdots x_k!}\, p_1^{x_1} \cdots p_k^{x_k}.$$

This formula provides the ground truth for the oracle setting and is particularly useful for validating bounds. For small ensembles ($n \lesssim 50$), it is possible to compute this effectively via a dynamic-programming scheme (see Appendix A.1), but quickly becomes intractable for increasing n. Theorem 2.1 is not illuminating about the drivers of certainty. To see these more clearly, we leverage concentration bounds which provide exponentially decaying finite-sample bounds.

2.2 Finite-sample certificates

Under a unique mode and conditional independence of rollouts, majority voting admits exponentially decaying error bounds which are valid for any finite number of samples. We collect the main instances into a single statement.

Theorem 2.2 (Finite-sample certificate)**.** *Assume $p_{c^\star} > \max_{j \neq c^\star} p_j$. Then for all $n \geq 1$,*

$$\Pr[\widehat{c}_n \neq c^\star] \;\leq\; \sum_{j \neq c^\star} \min\Bigg\{ \underbrace{\exp\Big(-\tfrac{n}{2}(p_{c^\star} - p_j)^2\Big)}_{\text{Hoeffding}},\quad \underbrace{\exp\Big(-\tfrac{n(p_{c^\star}-p_j)^2}{2\sigma_j^2 + \frac{2}{3}(p_{c^\star}-p_j) + \frac{2}{3}(p_{c^\star}-p_j)^2}\Big)}_{\text{Bernstein}},$$
$$\underbrace{\exp\Big(n \log\big(1 - (\sqrt{p_{c^\star}} - \sqrt{p_j})^2\big)\Big)}_{\text{Chernoff–Markov}} \Bigg\}.$$

Introducing the probability gap $\delta^2 = \min_{j \neq c^\star}(p_{c^\star} - p_j)^2$, Hoeffding's inequality implies that

$$\mathbb{P}[\widehat{c}_n \neq c^\star] \leq (k-1)e^{-n\delta^2/2}.$$

From this we obtain that $n \geq -\frac{2}{\delta^2} \log\left(\frac{\varepsilon}{k-1}\right)$ samples are sufficient to guarantee that the majority vote is correct with probability at least $1 - \varepsilon$.

Interpretation. We observe that the probability gaps $p_{c^\star} - p_j$ play a major role in these bounds. While Hoeffding's rate depends only on the gap, Bernstein tightens the rate when variances are smaller, offering an advantage when few rivals have non-negligible mass. These bounds can be further tightened through the introduction of additional prefactors (Bahadur & Rao, 1960). A full statement with explicit constants and proofs can be found in Appendices A.2-A.4. A weighted-majority extension (heterogeneous experts) of Hoeffding's bound is deferred to Appendix A.2.1.

2.3 Asymptotic consistency and the governing rate

As n grows, the above finite sample bounds yield *exponential* improvement in reliability. In the asymptotic regime ($n \to \infty$) we are able to leverage additional strategies which yield different perspectives on the driving factors. There are two complementary asymptotic lenses:

(i) Gaussian/CLT regime. Viewing the multinomial counts through a multivariate central limit theorem (CLT) yields normal tail approximations for the pairwise margins $N_{c^\star} - N_j$. These can be further refined through Berry–Esseen corrections, which provide $O(n^{-1/2})$ refinements.

(ii) Large-deviations (Sanov/Cramér) regime. A large-deviation analysis (Dembo & Zeitouni, 2010) characterises the exact first-order exponent: $\Pr[\widehat{c}_n \neq c^\star] = \exp(-nI^\star(\mathbf{p}) + o(n))$, where $I^\star(\mathbf{p})$ is the minimal KL divergence to a distribution in which a rival ties the leader. Bahadur–Rao–type refinements provide $\Theta(n^{-1/2})$ prefactors to further tighten these approximations.

The two views agree to second order: for small margins $\delta = p_{c^\star} - p_{j^\star} \ll p_{j^\star}$, the large-deviation exponent expands as $I^\star(\mathbf{p}) = \delta^2/(2\sigma_{j^\star}^2) + O(\delta^3)$, matching the CLT rate (1). Practically, the CLT bound gives a transparent dependence on SNR and is useful for interpretable sample-complexity proxies, while the Sanov rate is preferable when a sharp exponent is needed or when inverting for n.

The results are detailed in the following theorem, which summarises both the CLT and large-deviations regimes.

Theorem 2.3 (Asymptotic consistency). *Assume $p_{c^\star} > \max_{j \neq c^\star} p_j$. Then, as $n \to \infty$,*

$$\begin{aligned} \Pr[\widehat{c}_n = c^\star] &= 1 - \sum_{j \neq c^\star} \Phi\Big(- \tfrac{(p_{c^\star} - p_j)\sqrt{n}}{\sigma_j} \Big) \left[1 + O(n^{-1/2})\right] \\ &\geq 1 - \frac{k-1}{2} \exp\Big\{ - \frac{n}{2} \min_{j \neq c^\star} \Big(\tfrac{p_{c^\star} - p_j}{\sigma_j} \Big)^2 \Big\}, \end{aligned} \tag{1}$$

where Φ is the standard normal CDF and $\sigma_j^2 = p_{c^\star} + p_j - (p_{c^\star} - p_j)^2$. Moreover,

$$\Pr[\widehat{c}_n \neq c^\star] \; = \; \exp\big(- n \, I^\star(\mathbf{p}) + o(n) \big),$$

$$I^\star(\mathbf{p}) = \min_{j \neq c^\star} \inf_{\mathbf{q}: \, q_{c^\star} = q_j} D_{\mathrm{KL}}(\mathbf{q} \| \mathbf{p}) = - \log\Big(1 - (\sqrt{p_{c^\star}} - \sqrt{p_{j^\star}})^2 \Big).$$

Motivated by the Gaussian bound we define the *signal-to-noise ratio* (SNR) by

$$\mathrm{SNR}(\Delta_{j^\star}) = \frac{\delta^2}{2p_{c^\star} - \delta - \delta^2}$$

where $\delta = p_{c^\star} - p_{j^\star}$. The Gaussian bound reveals that the decay rate is governed by the *worst* signal-to-noise ratio of the margin variables:

$$\min_{j \neq c^\star} \Big(\tfrac{p_{c^\star} - p_j}{\sigma_j} \Big)^2 \; = \; \Big(\tfrac{p_{c^\star} - p_{j^\star}}{\sigma_{j^\star}} \Big)^2 \; = \; \mathrm{SNR}(\Delta_{j^\star}). \tag{2}$$

We also note that the rate function $I^\star(\mathbf{p})$ recovers the same rate as the Chernoff-Markov bound in Theorem 2.2. For small margins $\delta \ll p_{j^\star}$, the large-deviation exponent admits the expansion

$$I^\star(\mathbf{p}) = \delta^2 / \big(2\sigma_{j^\star}^2\big) + O(\delta^3),$$

consistent with the Gaussian rate. Proofs are given in Appendices A.5 and A.6.

From these results, we see that majority voting acts as a statistical amplifier: under a unique mode and conditionally-independent rollouts, the error probability decays exponentially in n. The governing rate is the SNR of the margin $\Delta_{j^\star}$ in (2). This same quantity controls the Martingale Majority Certificate (Section 3) and motivates test-time training objectives that enlarge the mode margin and improve sample efficiency (Section 4).

3 Martingale Majority Certificate: a practical stopping rule

In this section we introduce the *Martingale Majority Certificate* (MMC), a principled stopping rule that adaptively decides when to stop sampling rollouts while controlling the error of returning the empirical majority. Rather than fixing n in advance, MMC updates after each new sample and stops once the empirical evidence is sufficient.

We consider the following setting: at step n, we have samples $X_1, \ldots, X_n \sim X$ from the terminal distribution over $\{1, \ldots, k\}$, generated from n independent rollouts, where k is possibly unknown. These are independent and identically distributed, conditioned on the prompt pr. The true-but-unknown class probabilities are $p_j = \mathbb{P}[X = j \mid pr]$, and the empirical frequencies are $\hat{p}_{n,j}$.

Our goal is to construct a stopping rule that guarantees, with high confidence, that the majority vote $\widehat{c}_n = \arg\max_j \, \hat{p}_{n,j}$ coincides with the true mode $c^\star = \arg\max_j \, p_j$. Formally, we seek a strategy such that, at the stopping iteration n_τ, $\mathbb{P}\left[\widehat{c}_{n_\tau} \neq c^\star\right] \leq \varepsilon$. The central challenge in the LLM setting is the potentially large number of possible outcomes. A naive stopping rule would require pairwise comparisons of the empirical probabilities across all classes $i \neq j$, $i, j \in \{1, \ldots, k\}$, which becomes computationally prohibitive as k grows.

To address this, we exploit the observation that the mass of the terminal law is typically concentrated on a few classes $m \ll k$. Thus, instead of considering all classes individually, we aggregate votes into three categories: (i) the current leader $\widehat{c}_n$, (ii) the top-$(m-1)$ runner-ups, $j^\star_{n,1}, \dots, j^\star_{n,m-1}$, where $j^\star_{n,i} = \arg\max_{j \neq \widehat{c}_n, j^\star_{n,1}, \dots, j^\star_{n,i-1}} \hat{p}_{n,j}$, and (iii) all the *others*. Note that

$$\begin{aligned}\widehat{c}_n = c^\star \iff & \big(\forall\, i \,\in \{1,\dots,m-1\};\ p_{\widehat{c}_n} > p_{j^\star_{n,i}}\big) \text{ AND } \big(\forall\, j \in \{\textit{others}\};\ p_{\widehat{c}_n} > p_j\big) \\ \Longleftarrow\ & \big(\forall\, i \,\in \{1,\dots,m-1\};\ p_{\widehat{c}_n} > p_{j^\star_{n,i}}\big) \text{ AND } \big(p_{\widehat{c}_n} > \textstyle\sum_{j \in \textit{ others}} p_j\big).\end{aligned}$$

Accordingly, we perform two tests: leader vs top-$(m-1)$ runner-ups and leader vs *others*. We stop only when both conditions are satisfied with high probability, ensuring that $\widehat{c}_n$ coincides with the true mode with high confidence. In what follows, we focus on the case $m = 2$, a detailed construction of the stopping rule for general m is provided in Appendix B.5.

3.1 Anytime-valid e-processes

At round $n \geq 1$, *before* observing X_n, set the predictable top-2 labels as

$$A_{n-1} := \widehat{c}_{n-1}, \qquad B_{n-1} := j^\star_{n-1},$$

which are measurable w.r.t. $\mathcal{F}_{n-1} = \sigma(X_1, \dots, X_{n-1})$ (ties broken deterministically). We maintain the following *recursive, predictable* counts

$$\begin{array}{lrcl} \textbf{Leader hits:} & s_n &=& s_{n-1} + \mathbf{1}\{X_n = A_{n-1}\}, \quad s_0 = 0, \\ \textbf{Runner-up hits (for the A vs B test):} & f_n &=& f_{n-1} + \mathbf{1}\{X_n = B_{n-1}\}, \quad f_0 = 0, \\ \textbf{Others hits (for the A vs others test):} & o_n &=& o_{n-1} + \mathbf{1}\{X_n \notin \{A_{n-1}, B_{n-1}\}\}, \quad o_0 = 0. \end{array}$$

Thus the sample sizes are

$$M_n := s_n + f_n, \qquad T_n := s_n + o_n.$$

Let $(\pi^{\text{run}}_n)_{n \geq 1}$ and $(\pi^{\text{oth}}_n)_{n \geq 1}$ be predictable priors (each π_n is $\mathcal{F}_{n-1}$-measurable) supported on $(1/2, 1]$. Define the two mixture e-processes recursively (with optional skipping) by

$$e^{\text{run}}_n = \begin{cases} e^{\text{run}}_{n-1} \cdot 2 \displaystyle\int \theta\, \pi^{\text{run}}_n(d\theta), & X_n = A_{n-1}, \\ e^{\text{run}}_{n-1} \cdot 2 \displaystyle\int (1-\theta)\, \pi^{\text{run}}_n(d\theta), & X_n = B_{n-1}, \\ e^{\text{run}}_{n-1}, & \text{otherwise}, \end{cases}$$

$$e^{\text{oth}}_n = \begin{cases} e^{\text{oth}}_{n-1} \cdot 2 \displaystyle\int \lambda\, \pi^{\text{oth}}_n(d\lambda), & X_n = A_{n-1}, \\ e^{\text{oth}}_{n-1} \cdot 2 \displaystyle\int (1-\lambda)\, \pi^{\text{oth}}_n(d\lambda), & X_n \notin \{A_{n-1}, B_{n-1}\}, \\ e^{\text{oth}}_{n-1}, & \text{if } X_n = B_{n-1}, \end{cases}$$

with $e^{\text{run}}_0 = e^{\text{oth}}_0 = 1$.

Equivalently, by aggregating the per-round factors,

$$e^{\text{run}}_n = 2^{M_n} \int \prod_{i=1}^{n} \theta_i^{\mathbf{1}\{X_i = A_{i-1}\}} (1-\theta_i)^{\mathbf{1}\{X_i = B_{i-1}\}}\, \Pi^{\text{run}}_n(d\boldsymbol{\theta}), \tag{3}$$

$$e^{\text{oth}}_n = 2^{T_n} \int \prod_{i=1}^{n} \lambda_i^{\mathbf{1}\{X_i = A_{i-1}\}} (1-\lambda)_i^{\mathbf{1}\{X_i \notin \{A_{i-1}, B_{i-1}\}\}}\, \Pi^{\text{oth}}_n(d\boldsymbol{\lambda}), \tag{4}$$

where Π^{run}_n (resp. Π^{oth}_n) denotes a prior on the vector $\boldsymbol{\theta}$ (resp. $\boldsymbol{\lambda}$) and must be predictable, i.e. $\mathcal{F}_{n-1}$-measurable. If Π_n is a product distribution, we are re-mixing, i.e. not sharing information across steps. If it is not a product distribution, we have the opportunity to be a bit more efficient.

The following theorem shows that the e-processes defined above provide anytime-valid tests.

Theorem 3.1 (Anytime validity)**.** *Let $p_j = \mathbb{P}[X = j \mid pr]$. For the* A vs B *test (leader vs runner-up), define $\theta_n = \frac{p_{A_{n-1}}}{p_{A_{n-1}} + p_{B_{n-1}}}$ and the one-sided composite null*

$$H^{\text{run}}_0: \quad \theta_n \leq \tfrac{1}{2} \ \big(\textit{equivalently } p_{A_{n-1}} \leq p_{B_{n-1}}\big) \ \textit{at every round } n.$$

For the A vs others *test, define* $\lambda_n = \frac{p_{A_{n-1}}}{p_{A_{n-1}}+\sum_{j\notin\{A_{n-1},B_{n-1}\}} p_j} = \frac{p_{A_{n-1}}}{1-p_{B_{n-1}}}$ *and the composite null*

$$H_0^{\text{oth}}: \quad \lambda_n \le \tfrac{1}{2} \ \big(\textit{equivalently } p_{A_{n-1}} \le \textstyle\sum_{j\notin\{A_{n-1},B_{n-1}\}} p_j\big) \ \textit{at every round } n.$$

Then $\{e_n^{\text{run}}\}_{n\ge 0}$ *and* $\{e_n^{\text{oth}}\}_{n\ge 0}$ *defined in (3), (4) are non-negative test* supermartingales *w.r.t.* $\{\mathcal{F}_n\}$*, even with predictable, data-dependent priors and optional skipping. Under the boundary (simple) nulls (*$\theta_n \equiv \frac{1}{2}$ *or* $\lambda_n \equiv \frac{1}{2}$ *on their informative rounds), they are test* martingales*. Consequently, by Ville's inequality, for any stopping time,*

$$\sup_{\mathbb{P}\in H_0^{\text{run}}} \mathbb{P}\Big(\sup_{n\ge 0} e_n^{\text{run}} \ge 1/\varepsilon\Big) \le \varepsilon, \qquad \sup_{\mathbb{P}\in H_0^{\text{oth}}} \mathbb{P}\Big(\sup_{n\ge 0} e_n^{\text{oth}} \ge 1/\varepsilon\Big) \le \varepsilon.$$

The proof is provided in Appendix B.1.

Corollary 3.2 (Union null for stopping)**.** *Let* $H_0 := H_0^{\text{run}} \cup H_0^{\text{oth}}$*. Define the MMC stopping time* $N := \inf\{n : e_n^{\text{run}} \ge 1/\varepsilon \text{ and } e_n^{\text{oth}} \ge 1/\varepsilon\}$*. Then* $\sup_{\mathbb{P}\in H_0} \mathbb{P}(N < \infty) \le \varepsilon$*.*

Remark 3.3 (Why o_n excludes B_{n-1})**.** *The A vs others null is* $p_A \le \sum_{j\notin\{A,B\}} p_j$*, which is equivalent to* $\lambda \le 1/2$ *when we map successes to* $X = A$ *and failures to* $X \notin \{A, B\}$*. Including B among failures would test* $p_A \le 1/2$ *(absolute majority), which is unnecessarily strong.*

Pseudocode for implementing the MMC stopping rule is provided in Algorithm 1. If the maximum sample budget is reached, we return an upper bound $\hat{\varepsilon}$ on $\mathbb{P}[\hat{c}_n \ne c^\star]$. Details on how to compute $\hat{\varepsilon}$ are provided in Appendix B.2.

Algorithm 1 Martingale Majority Certificate stopping rule

```
Require: confidence level $\varepsilon$, budget $N_{\text{budget}}$, prior hyperparameters; deterministic tie-break rule
Init: $n \leftarrow 0$; for all $j \in \{1,\dots,k\}$ set label counts $N_j \leftarrow 0$; $s_0 = f_0 = o_0 \leftarrow 0$; $e_0^{\text{run}} = e_0^{\text{oth}} \leftarrow 1$
while True do
    Predictable top-2: set $A_n \leftarrow \arg\max_j N_j$, $B_n \leftarrow$ second largest (ties broken deterministically)
    Cache counts (pre-update): $\tilde{s} \leftarrow s_n$, $\tilde{f} \leftarrow f_n$, $\tilde{o} \leftarrow o_n$
    Draw a new vote: sample $X \sim \mathbb{P}[\,\cdot\,|pr]$;                ▷ the only source of randomness per round
    Per-round ratio (A vs B):
        $\rho_{\text{run}} = \begin{cases} 2\int \theta\, \pi_n^{\text{run}}(d\theta), & X = A_n, \\ 2\int (1-\theta)\, \pi_n^{\text{run}}(d\theta), & X = B_n, \\ 1, & \text{otherwise,} \end{cases}$
    Per-round ratio (A vs others):
        $\rho_{\text{oth}} = \begin{cases} 2\int \lambda\, \pi_n^{\text{oth}}(d\lambda), & X = A_n, \\ 2\int (1-\lambda)\, \pi_n^{\text{oth}}(d\lambda), & X \notin \{A_n, B_n\}, \\ 1, & \text{if } X = B_n, \end{cases}$
    Update $e$-values: $e_{n+1}^{\text{run}} \leftarrow e_n^{\text{run}} \cdot \rho_{\text{run}}$,  $e_{n+1}^{\text{oth}} \leftarrow e_n^{\text{oth}} \cdot \rho_{\text{oth}}$
    Update recursive counts:
        $(s_{n+1}, f_{n+1}, o_{n+1}) = \begin{cases} (\tilde{s}+1, \tilde{f}, \tilde{o}), & X = A_n, \\ (\tilde{s}, \tilde{f}+1, \tilde{o}), & X = B_n, \\ (\tilde{s}, \tilde{f}, \tilde{o}+1), & \text{otherwise.} \end{cases}$
    Update label counts: $N_X \leftarrow N_X + 1$;  $n \leftarrow n + 1$
    Check stop: if $e_n^{\text{run}} \ge 1/\varepsilon$ and $e_n^{\text{oth}} \ge 1/\varepsilon$ then
        set $\hat{c} \leftarrow \arg\max_j N_j$; return $(\hat{c},$ stopped$)$
    Budget: if $n \ge N_{\text{budget}}$ then return $(\arg\max_j N_j,$ abstained$)$
```

3.2 Two practical priors: truncated $\text{Beta}(a, b)$ and an updating point prior

We introduce two priors to compute the e-processes. Their performance is evaluated on synthetic data in Appendix B.6.

A. Truncated $\mathrm{Beta}(a, b)$ **prior on** $(\frac{1}{2}, 1]$. For convenience, define the *upper–half Beta mass*

$$\mathsf{B}_{>1/2}(a, b) \;:=\; \int_{1/2}^{1} t^{a-1}(1-t)^{b-1}\, dt\,.$$

Here we use a *single* latent parameter (shared across informative rounds), that is,

$$\Pi_n^{\mathrm{run}}(d\boldsymbol{\theta}) \propto \theta^{a-1}(1-\theta)^{b-1}\mathbf{1}\{\theta > 1/2\}\prod_{i=1}^{n}\delta_\theta(d\theta_i),$$
$$\Pi_n^{\mathrm{oth}}(d\boldsymbol{\lambda}) \propto \lambda^{a-1}(1-\lambda)^{b-1}\mathbf{1}\{\lambda > 1/2\}\prod_{i=1}^{n}\delta_\lambda(d\lambda_i).$$

The mixture e-values admit closed forms in terms of $\mathsf{B}_{>1/2}$:

$$e_n^{\mathrm{run}} = 2^{M_n}\,\frac{\mathsf{B}_{>1/2}(a+s_n,\, b+f_n)}{\mathsf{B}_{>1/2}(a,b)}\,, \qquad e_n^{\mathrm{oth}} = 2^{T_n}\,\frac{\mathsf{B}_{>1/2}(a+s_n,\, b+o_n)}{\mathsf{B}_{>1/2}(a,b)}\,.$$

These can be updated online by using *ratios*:

$$\frac{e_n^{\mathrm{run}}}{e_{n-1}^{\mathrm{run}}} = \begin{cases} 2\,\dfrac{\mathsf{B}_{>1/2}(a+s_{n-1}+1,\, b+f_{n-1})}{\mathsf{B}_{>1/2}(a+s_{n-1},\, b+f_{n-1})}, & X_n = A_{n-1}, \\ 2\,\dfrac{\mathsf{B}_{>1/2}(a+s_{n-1},\, b+f_{n-1}+1)}{\mathsf{B}_{>1/2}(a+s_{n-1},\, b+f_{n-1})}, & X_n = B_{n-1}, \\ 1, & \text{otherwise,} \end{cases}$$

$$\frac{e_n^{\mathrm{oth}}}{e_{n-1}^{\mathrm{oth}}} = \begin{cases} 2\,\dfrac{\mathsf{B}_{>1/2}(a+s_{n-1}+1,\, b+o_{n-1})}{\mathsf{B}_{>1/2}(a+s_{n-1},\, b+o_{n-1})}, & X_n = A_{n-1}, \\ 2\,\dfrac{\mathsf{B}_{>1/2}(a+s_{n-1},\, b+o_{n-1}+1)}{\mathsf{B}_{>1/2}(a+s_{n-1},\, b+o_{n-1})}, & X_n \notin \{A_{n-1}, B_{n-1}\}, \\ 1, & X_n = B_{n-1}. \end{cases}$$

Recommended hyperparameters. $a = b = \frac{1}{2}$ (Jeffreys) or $a = b = 1$ (Laplace) are robust defaults. Truncation to $(1/2, 1]$ ensures support under the one-sided alternative and yields the required supermartingale property for the composite null via the boundary case $\theta = \frac{1}{2}$ (resp. $\lambda = \frac{1}{2}$).

B. Updating plug-in point prior. In this case, we share information across the two tests by maintaining a single plug–in estimate of the multinomial parameters for the predictable top–2 and the aggregated others.

Fix smoothing hyperparameters $(\alpha_A, \alpha_B, \alpha_O) > 0$ and set

$$\hat p_{A,n} \;:=\; \tfrac{s_{n-1}+\alpha_A}{L_{n-1}+\alpha_A+\alpha_B+\alpha_O}, \quad \hat p_{B,n} \;:=\; \tfrac{f_{n-1}+\alpha_B}{L_{n-1}+\alpha_A+\alpha_B+\alpha_O}, \quad \hat p_{O,n} \;:=\; \tfrac{o_{n-1}+\alpha_O}{L_{n-1}+\alpha_A+\alpha_B+\alpha_O},$$

where $L_{n-1} := s_{n-1} + f_{n-1} + o_{n-1}$. Define the one–dimensional informative-round parameters

$$\theta_n^\star \;:=\; \mathrm{clip}\left(\frac{\hat p_{A,n}}{\hat p_{A,n} + \hat p_{B,n}},\; \tfrac{1}{2}+\varepsilon,\; 1-\varepsilon\right), \qquad \lambda_n^\star \;:=\; \mathrm{clip}\left(\frac{\hat p_{A,n}}{1-\hat p_{B,n}},\; \tfrac{1}{2}+\varepsilon,\; 1-\varepsilon\right),$$

where $\varepsilon \in (0, 10^{-3}]$ ensures numerical stability. We consider two different e-processes:

(B.1) Consider the shared-parameter priors

$$\Pi_n^{\mathrm{run}}(d\boldsymbol{\theta}) = \prod_{i=1}^{n}\delta_{\theta_n^\star}(d\theta_i), \qquad \Pi_n^{\mathrm{oth}}(d\boldsymbol{\lambda}) = \prod_{i=1}^{n}\delta_{\lambda_n^\star}(d\lambda_i).$$

The corresponding mixture e-values are given by

$$e_n^{\mathrm{run}} = 2^{M_n}(\theta_n^\star)^{s_n}(1-\theta_n^\star)^{f_n}, \qquad e_n^{\mathrm{oth}} = 2^{T_n}(\theta_n^\star)^{s_n}(1-\theta_n^\star)^{o_n}\,.$$

(B.2) The second one is defined by its per-round update factors

$$\frac{e_n^{\mathrm{run}}}{e_{n-1}^{\mathrm{run}}} = \begin{cases} 2\,\theta_n^\star, & X_n = A_{n-1}, \\ 2\,(1-\theta_n^\star), & X_n = B_{n-1}, \\ 1, & \text{otherwise}, \end{cases} \qquad \frac{e_n^{\mathrm{oth}}}{e_{n-1}^{\mathrm{oth}}} = \begin{cases} 2\,\lambda_n^\star, & X_n = A_{n-1}, \\ 2\,(1-\lambda_n^\star), & X_n \notin \{A_{n-1}, B_{n-1}\}, \\ 1, & X_n = B_{n-1}. \end{cases}$$

By construction, $\theta_n^\star, \lambda_n^\star$ are $\mathcal{F}_{n-1}$–measurable and lie in $(\frac{1}{2}, 1]$ after clipping, so Theorem 3.1 applies: $\{e_n^{\mathrm{run}}\}$ and $\{e_n^{\mathrm{oth}}\}$ are non-negative test supermartingales under their respective composite nulls, and test martingales under the boundary nulls. Ville's inequality then yields time–uniform guarantees.

Heuristic sample complexity. If the informative–round parameter $\vartheta \in (\frac{1}{2}, 1)$ is well tracked by the plug–in estimate, each e–process in (B.1) crosses $1/\varepsilon$ after roughly $\log(1/\varepsilon)/D_{\mathrm{KL}}(\mathrm{Ber}(\vartheta)\|\mathrm{Ber}(\frac{1}{2}))$ informative draws. See Appendix B.3 for details.

4 Optimising sample efficiency through test-time training

Our ultimate goal is to minimise the number of samples required from the LLM for the majority vote to return the correct answer with high confidence $1-\varepsilon$. From the analysis in Section 3, the expected stopping time of the MMC scales approximately as

$$N \;\approx\; \frac{2(p_{\hat{c}} + p_{j^\star})}{(p_{\hat{c}} - p_{j^\star})^2}\,\log(1/\varepsilon), \tag{5}$$

so that small mode margins $\delta = p_{\hat{c}} - p_{j^\star}$ lead to rapidly increasing sample requirements (see Appendix B.3 for details). The key question is whether test-time adaptation can reshape the terminal distribution to enlarge this margin, thereby improving sample efficiency.

Effect of test-time training. Test-time reinforcement learning (TTRL; Zuo et al., 2025) adapts model parameters at inference time by maximising a KL-regularised objective based on self-generated rewards. Given a prompt pr, let $(Y_t)_{t\geq 0}$ be the autoregressive token process from a reference distribution $\pi_{\mathrm{ref}}(\cdot\,|\,pr)$ on trajectories. Let $X = g(Y_{\tau:})$, where τ is the time at which the answer is generated, which is a (finite a.s.) stopping time with respect to the canonical filtration.

Given n trajectories $Y_1, \ldots, Y_n \sim \pi_{\mathrm{ref}}$, yielding answers $X_1, \ldots, X_n$, let $\widehat{c}_n$ be the associated majority vote. The reward introduced in Zuo et al. (2025) is $r_n(Y_i) = \mathbf{1}\{X_i = \widehat{c}_n\}$. The associated KL-regularised optimisation over trajectory laws parametrised by $\pi_\phi \ll \pi_{\mathrm{ref}}$ is given by

$$\max_\phi \; \mathbb{E}_{Y\sim\pi_\phi(\cdot|pr)}[r_n(Y)] - \beta\, D_{\mathrm{KL}}(\pi_\phi\|\pi_{\mathrm{ref}}).$$

The optimal policy is an *exponentially tilted* distribution

$$\pi^\star(Y|pr) = \frac{e^{r_n(Y)/\beta}\,\pi_{\mathrm{ref}}(Y|pr)}{Z_\beta(pr)}, \qquad Z_\beta(pr) = 1 + \pi_{\mathrm{ref}}(\widehat{c}_n|pr)\big(e^{1/\beta} - 1\big),$$

where the denominator is the normalising constant $Z_\beta = \mathbb{E}_{\pi_{\mathrm{ref}}}[e^{r_n(Y)/\beta}]$. Writing $\kappa = 1/\beta$, the tilting sharpens the terminal law around the majority mode and monotonically increases the signal-to-noise ratio (SNR) of the margin variable $\Delta_{j_n^\star} = \mathbf{1}\{X = \widehat{c}_n\} - \mathbf{1}\{X = j_n^\star\}$:

$$\tfrac{d}{d\kappa}\mathrm{SNR}\big(\Delta_{j_n^\star}\big)(\kappa) \geq 0,$$

with equality only if $p_{\widehat{c}_n} = 1$, i.e. the distribution is a Dirac delta at the majority vote. Strict monotonicity holds between values of κ for which the runner-up $j_n^\star$ remains fixed; at swap points the SNR function is continuous but non-differentiable. Thus, increasing κ (i.e. stronger tilting) consistently improves the margin and reduces the number of samples required for certification. See Appendix C.1 for further details.

Two new test-time RL objectives. We introduce two label-free group-level rewards designed to optimise the trade-off between sharpness and bias. Let $\mathbf{X} = (X_1, \ldots, X_n)$ be a set of answers arising from rollouts $\mathbf{Y} = (Y_1, \ldots, Y_n)$ for a given prompt, with $\widehat{c}_n$ denoting the majority vote and $j_n^\star$ the runner-up. Define $N_j = \sum_i \mathbf{1}\{X_i = j\}$.

(i) **SNR-based reward.** Directly leveraging the SNR as a driving factor in the efficiency of the MMC scheme we introduce the first reward

$$r_n^{(1)}(\mathbf{Y}) = \widehat{\mathrm{SNR}}(\Delta_{j_n^\star})(\mathbf{X}) = \frac{(N_{\widehat{c}_n} - N_{j_n^\star})^2}{n\big(N_{\widehat{c}_n} + N_{j_n^\star}\big) - (N_{\widehat{c}_n} - N_{j_n^\star})^2} \xrightarrow[n\to\infty]{} \mathrm{SNR}(\Delta_{j_n^\star}). \tag{6}$$

This objective aims to directly maximise $\mathrm{SNR}(\Delta_{j_n^*})$, which is equivalent to minimising the expected number of samples required to obtain statistical certificates for the majority vote.

(ii) **Entropy-based reward.** As we want to encourage a more peaked terminal distribution, another natural option is negative entropy, i.e.

$$r_n^{(2)}(\mathbf{Y}) = \widehat{H}_n(\mathbf{X}) = \sum_{j:N_j>0} \frac{N_j}{n} \log \frac{N_j}{n} \xrightarrow[n\to\infty]{} \sum_j p_j \log p_j = -H(p). \tag{7}$$

Maximising $\widehat{H}_n$ *minimises* the Shannon entropy of the answer distribution, encouraging a sharper, lower-entropy distribution.

Solving the corresponding KL-regularised variational problems (Appendices C.2, C.3) yields the respective optimisers. As with the TTRL tilt, $\mathrm{SNR}(\Delta_{j_n^\star})$ is non-decreasing, implying that sharper distributions require fewer samples for reliable certification. It is important to emphasise that our proposed entropy-based reward differs from that of (Agarwal et al., 2025).

The entropy reward $r_n^{(2)}$ should be understood as penalising entropy of the terminal distribution of the trajectory distribution, not to the full trajectory law itself. Formally, let $\pi_{\mathrm{ref}}(Y_{0:\tau})$ denote the reference distribution over reasoning trajectories with terminal variable $X = g(Y_{\tau:})$, and write $p_{\mathrm{ref}}(x) = \pi_{\mathrm{ref}}(X = x)$ for its induced marginal. Applying the KL chain rule,

$$D_{\mathrm{KL}}(\pi_\phi \| \pi_{\mathrm{ref}}) = D_{\mathrm{KL}}(q \| p_{\mathrm{ref}}) + \mathbb{E}_{x\sim q}\big[D_{\mathrm{KL}}(\pi_\phi(\cdot|X = x) \| \pi_{\mathrm{ref}}(\cdot|X = x))\big],$$

where $q(x) = \pi_\phi(X = x)$ is the terminal marginal of the adapted policy. Because the entropy reward depends only on X, the second term is minimised when $\pi_\phi(\cdot|X = x) = \pi_{\mathrm{ref}}(\cdot|X = x)$ for all x. Hence, the KL-regularised variational problem over the base measure reduces to one over the marginal q alone:

$$\max_{q\in\Delta(\mathcal{X})} \big\{ -H(q) - \beta\, D_{\mathrm{KL}}(q \| p_{\mathrm{ref}}) \big\}.$$

The unique maximiser of this objective is $q^\star(x) \propto p_{\mathrm{ref}}(x)^\kappa$ with $\kappa = \beta/(\beta - 1) > 1$. Hence the test-time adaptation *tempers the terminal marginal* $p_{\mathrm{ref}}(x)$, while preserving the reference conditional trajectory law $\pi_{\mathrm{ref}}(\cdot|X = x)$. In particular,

$$\pi_\phi^\star(Y_{0:\tau}) = \pi_{\mathrm{ref}}(Y_{0:\tau} \mid X)\, q^\star(X) \;\neq\; \frac{\pi_{\mathrm{ref}}(Y_{0:\tau})^\kappa}{\int \pi_{\mathrm{ref}}(Y_{0:\tau})^\kappa \, dY_{0:\tau}},$$

except in the degenerate case where $\pi_{\mathrm{ref}}(\cdot|X = x)$ is uniform for all x. The tempering therefore sharpens only the distribution of final answers, not the full sequence distribution. This gives us the best of both worlds: promoting certainty when providing a final answer, but permitting exploration of diverse pathways during the chain-of-thought reasoning process. In particular, this should not be confused with *low temperature scaling*, where the conditional next-token distributions of the full trajectory is tempered according to a temperature schedule Wang et al. (2020).

Because the reward functions couple multiple variables, the corresponding gradient estimates can exhibit high variance. To reduce this variance, we adopt a leave-one-out control variate approach (Tang et al., 2025), resulting in the following effective advantage functions for Y_i

$$A_i^{(1)} = \widehat{\mathrm{SNR}}(\Delta_{j_n^\star})(\mathbf{X}) - \widehat{\mathrm{SNR}}(\Delta_{j_n^\star})(\mathbf{X}_{-i}), \qquad A_i^{(2)} = \hat{H}_n(\mathbf{X}) - \hat{H}_{n-1}(\mathbf{X}_{-i}). \tag{8}$$

This preserves unbiasedness and substantially reduce gradient variance in REINFORCE-style optimisation.

We post-train our models using the GRPO algorithm (Shao et al., 2024) for each of these rewards. Details can be found in Appendix C. By contrast with the TTRL reward $r_n(Y) = 1\{X = \hat{c}_n\}$, a benefit of both SNR- and entropy- based rewards is that these yield smoother signals of consensus. In practice, this results in significantly faster and more stable convergence of the RL-loss function, consistent with similar observations made in Huang et al. (2025); Ma et al. (2025); Tao et al. (2025).

5 SNR AS A LABEL-FREE ESTIMATOR OF TASK DIFFICULTY

The preceding analysis establishes that signal-to-noise ratio plays a governing role in certifying self-consistency, as well as in the associated test-time training objectives. Given n rollouts $\{Y_i\}_{i=1}^n$ from a prompt pr, with terminal answers $X_i = g(Y_{i,\tau:})$, let $\widehat{c}_n$ and $j_n^\star$ denote the empirical leader and runner-up. We compute an empirical estimate of the SNR given by,

$$\widehat{\mathrm{SNR}}(\Delta_{j_n^\star})(\mathbf{X}) = \frac{(N_{\hat{c}_n} - N_{j_n^\star})^2}{n(N_{\hat{c}_n} + N_{j_n^\star}) - (N_{\hat{c}_n} - N_{j_n^\star})^2}, \tag{9}$$

where $N_j = \sum_i 1\{X_i = j\}$. This statistic can be computed directly from model rollouts and requires no access to external signals.

In Figures 2c and 2d we plot the estimated SNR values, generated over the MATH-500 benchmark against the reported problem level, with 1 being the easiest and 5 being the hardest, Lightman et al. (2023). We observe that $\widehat{\mathrm{SNR}}$ values correlate strongly with ground-truth difficulty levels: harder problems exhibit systematically lower SNR and greater variability. This emergent calibration occurs without supervision: the model's own epistemic uncertainty, quantified via SNR, consistently aligns with external difficulty labels. As values of $\widehat{\mathrm{SNR}}$ correspond to sharply peaked terminal marginals in which the model consistently produces the same answer across rollouts, we observe that "easy" prompts yield high-SNR and thus low epistemic uncertainty. Conversely, low SNR values arise for diffuse or multi-modal terminal distributions, suggesting that reasoning models demonstrate uncertainty around harder or more ambiguous questions. The observations align with previous works which seek to use uncertainty as a proxy for problem difficulty, Wang et al. (2025a); Wan et al. (2025); Fu et al. (2025), with the aim of dynamically allocating resources.

6 NUMERICAL EXPERIMENTS

The goal of this section is threefold: (1) to evaluate the performance of our proposed test-time RL objectives (Section 4), (2) to empirically demonstrate that inference-time training strategies reduce the number of samples required by the MMC stopping rule (Algorithm 1) to obtain statistical certificates, compared to pre-trained models, and (3) to show that the SNR serves as a label-free proxy for problem difficulty. Additional experimental details are provided in Appendix D.

6.1 EXPERIMENTAL SETUP

Models and benchmarks. We use both base and instruct models of various scales, specifically Qwen2.5-Math-1.5B, Qwen2.5-Math-7B (Yang et al., 2024), Qwen2.5-7B (Yang et al., 2025) and LLaMA-3.1-8B-Instruct (Grattafiori et al., 2024). We consider three mathematical reasoning benchmarks: AIME 2024, AMC (Li et al., 2024a), and MATH-500 (Hendrycks et al., 2021).

Methods and evaluation. For test-time training, we use the VERL framework (Sheng et al., 2024) with the GRPO algorithm (Shao et al., 2024) on 8×H100 Nvidia GPUs. We apply our method to each benchmark individually and report both pass@1 and majority-vote accuracy (see Appendix D). We compare the performance of our proposed RL objectives with TTRL (Zuo et al., 2025).

6.2 RESULTS

Table 1 reports the pass@1 performance of various inference-time training strategies across different benchmarks and models. An extended version, comparing the improvements in pass@1 accuracy for both the score and format score, is provided in Table 3 (Appendix D). Overall, these strategies consistently enhance pass@1 performance, with the effect being particularly pronounced for Qwen2.5-Math-1.5B, the smallest model. This suggest that such test-time methods help reveal the model's underlying capabilities.

Besides, we analyse how test-time training reduces the number of samples required to guarantee, with high confidence, that the majority vote $\widehat{c}_n$ matches the true mode $c^\star$. Specifically, Table 2 reports the majority vote accuracy and the required number of samples under the MMC stopping

Table 1: Comparison of pass@1 performance before and after applying test-time training strategies.

	AIME	AMC	Math-500
Qwen2.5-7B	9.4	31.2	59.1
SNR (Ours)	23.3	51.8	80.3
Entropy (Ours)	20.0	49.2	77.6
Zuo et al. (2025)	24.3	53.4	79.6
Llama-3.1-8B	4.4	21.8	48.2
SNR (Ours)	13.4	29.3	59.2
Entropy (Ours)	13.3	27.0	55.4
Zuo et al. (2025)	10.0	32.3	63.7

	AIME	AMC	Math-500
Qwen2.5-Math-7B	10.6	31.0	47.1
SNR (Ours)	36.7	65.0	84.5
Entropy (Ours)	38.3	65.4	82.4
Zuo et al. (2025)	37.9	63.5	83.6
Qwen2.5-Math-1.5B	7.1	28.1	31.4
SNR (Ours)	16.3	45.4	72.0
Entropy (Ours)	15.6	45.9	70.8
Zuo et al. (2025)	15.8	48.4	71.9

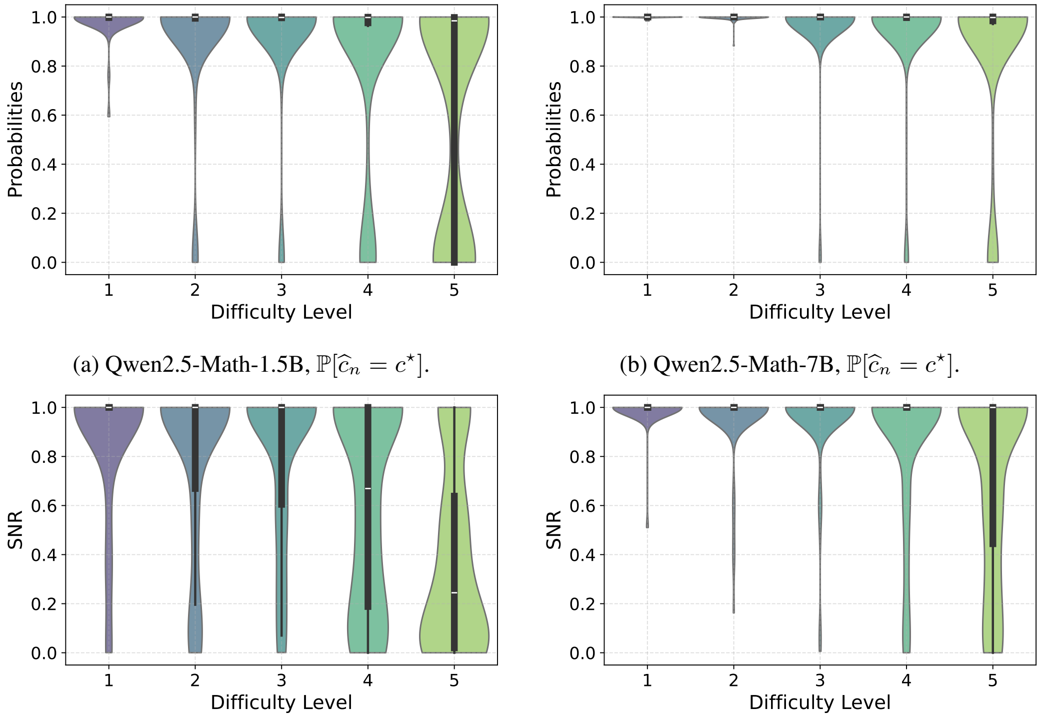
Table 2: Comparison of majority vote accuracy and required number of samples under the MMC stopping rule ($\mathbf{N}_{\text{adaptive}}$) for $\varepsilon = 0.1$ and 0.4 between the pre-trained model and after test-time training with SNR-based rewards. Performance is compared to that obtained using the full sample budget N_{budget} (✗). Results are shown for the MATH-500 dataset.

$\mathbf{N}_{\text{budget}}$	Adaptive sampling?	Qwen2.5-Math-7B		Qwen2.5-Math-1.5B	
		Pre-trained % ($\mathbf{N}_{\text{adaptive}}$)	Test-time trained % ($\mathbf{N}_{\text{adaptive}}$)	Pre-trained % ($\mathbf{N}_{\text{adaptive}}$)	Test-time trained % ($\mathbf{N}_{\text{adaptive}}$)
10	✗	61.6	85.2	36.0	78.6
	✓ $\varepsilon = 0.1$	61.6 **(9.7)**	85.2 **(9.4)**	36.0 **(9.9)**	78.6 **(9.4)**
	✓ $\varepsilon = 0.4$	61.6 **(9.2)**	85.2 **(8.9)**	36.0 **(9.7)**	78.6 **(8.6)**
50	✗	62.2	85.6	37.6	80.8
	✓ $\varepsilon = 0.1$	61.8 **(39.3)**	85.6 **(37.6)**	37.6 **(45.6)**	80.8 **(34.1)**
	✓ $\varepsilon = 0.4$	61.8 **(38.0)**	85.4 **(33.4)**	37.4 **(43.0)**	80.8 **(31.2)**
100	✗	62.2	85.6	36.6	81.2
	✓ $\varepsilon = 0.1$	62.2 **(74.9)**	85.6 **(67.2)**	36.8 **(86.5)**	81.0 **(61.2)**
	✓ $\varepsilon = 0.4$	62.2 **(73.1)**	85.4 **(60.8)**	36.4 **(81.8)**	80.8 **(56.9)**

rule (N_{adaptive}) for two confidence levels, $\varepsilon = 0.1$ and 0.4, comparing the pre-trained model with the model after test-time training using SNR-based rewards. For reference, we also include the majority vote accuracy obtained when using the full sample budget N_{budget}.

We observe that the MMC adaptive sampling scheme substantially reduces the number of samples without causing a noticeable degradation in performance. Moreover, the number of samples required under the MMC stopping rule further decreases after applying test-time training, relative to the pre-trained model. This effect is examined in more detail in Table 5, which reports the reduction in the ratio between N_{adaptive} and N_{budget} (given their approximately linear relationship). The decrease in this ratio after test-time training is most pronounced for the smaller 1.5B model. Improving sample efficiency is particularly important, as it directly translates to lower inference costs.

Finally, since the MATH-500 dataset classifies questions into five levels of increasing difficulty, we analyse the distributions of the estimated lower bound of the probability $\mathbb{P}[\widehat{c}_n = c^\star]$, as well as the estimated signal-to-noise ratio $\widehat{\text{SNR}}(\Delta_{j_n^\star})$ across these difficulty levels. Figure 2 shows that harder questions exhibit greater variability for both $\mathbb{P}[\widehat{c}_n = c^\star]$ and the SNR. In addition, for the smaller 1.5B model, both the probabilities and SNR distributions are more concentrated near zero for difficult questions compared to the 7B model. These observations further support the idea that the SNR can serve as a label-free proxy for problem difficulty.



(a) Qwen2.5-Math-1.5B, $\mathbb{P}[\widehat{c}_n = c^\star]$.

(b) Qwen2.5-Math-7B, $\mathbb{P}[\widehat{c}_n = c^\star]$.

(c) Qwen2.5-Math-1.5B, $\mathrm{SNR}(\Delta_{j_n^\star})$.

(d) Qwen2.5-Math-7B, $\mathrm{SNR}(\Delta_{j_n^\star})$.

Figure 2: Distribution of the estimated lower bound of the probability $\mathbb{P}[\widehat{c}_n = c^\star]$ (computed via Beta approximations) and the signal-to-noise ratio $\mathrm{SNR}(\Delta_{j_n^\star})$ when applying the MMC stopping rule with $\varepsilon = 0.1$ and $N_{\text{budget}} = 100$. Results are obtained after test-time training with SNR-based rewards on the MATH-500 dataset.

7 Related work

Classical majority aggregation. The study of majority voting as a mechanism for error reduction dates back to Condorcet's jury theorem, which shows that under independence and competence above chance, majority aggregation recovers the correct decision with probability approaching one as the ensemble size grows (de Condorcet, 1785). Subsequent work has analysed correlated jurors (Ladha, 1992), multiclass outcomes (List & Goodin, 2001), and asymptotic behaviour (Boland, 1989). Concentration inequalities have long been used to control majority error in the binary case, providing simple finite-sample bounds on the probability of incorrect aggregation. In this work, we build on these results with the aim of systematically understanding the multinomial setting relevant for LLM outputs, and to reinterpret the resulting bounds explicitly as *certificates* of model reliability. Various extensions to Condorcet's original formalism have been considered. A closely related line of work models heterogeneous and possibly biased voters via the Dawid–Skene framework (Dawid & Skene, 1979), which introduces latent confusion matrices for each voter, estimating them via Expectation-Maximisation. This generalises majority vote to settings with unequal competence and asymmetric errors in the multiclass case. Subsequent extensions incorporate item difficulty and worker ability, yielding models akin to Item Response Theory (Bock, 1997), Bayesian treatments and priors over confusion matrices (Raykar et al., 2010; Liu et al., 2012; Kim & Ghahramani, 2012). These frameworks have been leveraged in the context of LLMs, both for assessing quality of data annotation e.g. Whitehill et al. (2009); Welinder et al. (2010), as well as for aggregation and combination of outputs from heterogeneous models (Yao et al., 2024; Song et al., 2025), or for uncertainty quantification (Kang et al., 2025a). Adapting our anytime statistical certificates in these more general settings will be the scope of future work.

Self-consistency and ensembles in LLMs. In the context of chain-of-thought (CoT) prompting, majority voting is widely known as *self-consistency* (Wang et al., 2022). By sampling multiple reasoning trajectories and returning the empirical mode, self-consistency significantly improves accuracy on reasoning benchmarks. Extensions include iterative refinement and self-feedback loops (Madaan et al., 2023; Shinn et al., 2023) and ensemble-style aggregation in large-scale systems such as PaLM 2 (Anil et al., 2023) and GPT-4 (OpenAI, 2023). These approaches demonstrate empirically that aggregation mitigates stochasticity in reasoning and that the marginal benefit of additional samples is highly instance-dependent.

More recent work has begun to address this dependency explicitly through *adaptive self-consistency*, where the number of sampled trajectories is determined dynamically through a stopping rule, informed by model uncertainty or rollout agreement. (Aggarwal et al., 2023; Li et al., 2024b; Wan et al., 2025). Difficulty-adaptive sampling schemes (Wang et al., 2025a) and early-stopping strategies such as *Self-Truncation Best-of-N* (ST-BoN; Wang et al. (2025b)) aim to minimise test-time compute while maintaining accuracy by halting when the vote distribution stabilises. Related adaptive compute frameworks learn to predict, mid-generation, whether further sampling would change the outcome (Manvi et al., 2024; Liu et al., 2024; Chen et al., 2024), thereby allocating more samples to difficult or ambiguous prompts and fewer to easy ones.

While the above adaptive self-consistency strategies share the same goal of halting rollouts when the empirical vote distribution stabilises, they provide no formal control over reliability. Our Martingale Majority Certificate (MMC) makes this principle explicit by framing aggregation as an *anytime-valid* hypothesis test through e-values (Ramdas & Wang, 2025). This guarantees uniform, finite-sample error control for all stopping times, offering a statistically grounded analogue to these heuristic adaptive-sampling strategies.

Test-time training and reinforcement learning. A complementary line of work has investigated *test-time adaptation*, in which the model is updated online at inference time. Early approaches include entropy minimisation and self-training in computer vision. More recently, test-time reinforcement learning (TTRL) has been introduced for LLMs, where the model is adapted by optimising KL-regularised objectives with respect to its own rollouts (Zuo et al., 2025). Related methods such as Akyürek et al. (2025) and Prasad et al. (2025) similarly adapt models at inference time to sharpen predictions and improve reliability. Similarly, Wen et al. (2025), propose a method called Internal Coherence Maximization (ICM), which fine-tunes pretrained language models without any external labels by maximising mutual predictability and logical consistency among the model's own generated labels. In Prabhudesai et al. (2025) and Kang et al. (2025b) the authors use token-level negative entropy as a reward signal for test-time reinforcement learning. Finally, Shafayat et al. (2025) explores RL post-training leveraging a consensus reward identical to Zuo et al. (2025), but without KL-regularisation with respect to the base measure, demonstrating it can generate measurable improvements, before the inevitable collapse.

While these approaches empirically demonstrate measurable improvements, their mechanism has not been theoretically clarified. Firstly, our analysis provides a unifying perspective: KL-regularised TTRL objectives correspond to exponential tilting of the terminal distribution, and entropy-penalising rewards are equivalent to marginal tempering. This explains why such methods increase the mode margin and thereby reduce the number of samples required for certification. Secondly, our work clarifies the essential role played by the KL-regularisation, without which the model eventually collapses under post-training.

8 Discussion

Our results unify several strands of recent work on reliable inference in LLMs, self-consistency, adaptive compute allocation, and test-time reinforcement learning (TTRL), under a common statistical perspective. Through this lens, majority voting emerges naturally as a means of estimating the mode of the terminal distribution. The validity of the majority vote as an estimate of the mode can be certified by finite-sample and asymptotic bounds. The Martingale Majority Certificate (MMC) extends this view by providing an operational test-time algorithm that determines, from model rollouts alone, when a response is statistically self-consistent. This recasts test-time scaling as a sequential decision problem with formal coverage guarantees, contrasting with heuristic stopping rules based on agreement or entropy thresholds.

Our analysis clarifies the underlying mechanism by which TTRL and related post-training approaches improve reasoning reliability: KL-regularised optimisation corresponds to an exponential tilting of the terminal law, sharpening it around its mode and increasing the signal-to-noise ratio (SNR) of the margin variable. This insight explains empirical observations of enhanced consistency after test-time adaptation, and motivates new label-free objectives such as our SNR- and entropy-based rewards, which explicitly target this trade-off between sharpness and bias. Unlike prior work that tunes temperature or per-token distributions, our formulation operates on the terminal marginal, preserving exploration during reasoning while promoting confidence in the final answer.

Beyond immediate applications to reasoning benchmarks, our framework offers a principled path toward certifiable reliability in language models. By unifying classical concentration theory, martingale testing, and reinforcement-style post-training within one formal structure, we obtain statistical interpretability for inference-time adaptation. This could extend naturally to multi-agent ensembles, verifier–generator systems, and other domains where LLMs operate under uncertainty. Future work will explore applying anytime-valid certificates to correlated rollouts, structured output spaces, and multi-verifier settings, as well as combining them with learned difficulty estimators for dynamic compute scheduling.

In this work, we have demonstrated the efficacy of anytime-valid certificates in the simplified setting of problems with discrete, multiple-choice outputs. It is worth emphasising that the MMC does not require a-priori knowledge of the set of possible answers. While this would enable one to apply similar approaches to free-text answers, this would still require some degree of response canonicalisation. Future work will explore alternative reformulations of MMC, which circumvent the need for 'binning' similar responses, while still providing statistical certificates.

9 Limitations

Our analysis assumes conditionally independent rollouts given a fixed prompt and context, corresponding exactly to standard stochastic decoding (e.g. temperature or nucleus sampling). This assumption holds for the inference regime considered here, where each completion is sampled independently from the model's conditional distribution, but future extensions could address adaptive or verifier-guided sampling strategies that introduce dependencies across rollouts. A second limitation concerns calibration: our SNR- and entropy-based quantities rely on the model's internal probabilities to reflect true epistemic uncertainty, which may not hold for all models or decoding temperatures. Empirically, our evaluation focuses on single-turn reasoning benchmarks; applying the framework to multi-turn dialogue, program synthesis, and structured prediction remains an open direction. Although anytime-valid stopping improves expected efficiency, generating multiple trajectories still incurs substantial compute cost. Future work will explore correlated-rollout models, calibration corrections, and hierarchical extensions to improve the robustness and scalability of certified reasoning.

References

Shivam Agarwal, Zimin Zhang, Lifan Yuan, Jiawei Han, and Hao Peng. The Unreasonable Effectiveness of Entropy Minimization in LLM Reasoning. *arXiv preprint arXiv:2505.15134*, 2025.

Pranjal Aggarwal, Aman Madaan, Yiming Yang, et al. Let's sample step by step: Adaptive-consistency for efficient reasoning and coding with llms. In *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

Ekin Akyürek, Mehul Damani, Adam Zweiger, Linlu Qiu, Han Guo, Jyothish Pari, Yoon Kim, and Jacob Andreas. The Surprising Effectiveness of Test-Time Training for Few-Shot Learning. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025.

Rohan Anil, Ed H Chi, Aakanksha Chowdhery, and et al. PaLM 2 Technical Report. *arXiv preprint arXiv:2305.10403*, 2023.

R. R. Bahadur and R. Ranga Rao. On Deviations of the Sample Mean. *The Annals of Mathematical Statistics*, 31(4):1015–1027, 1960.

Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Michal Podstawski, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Hubert Niewiadomski, Piotr Nyczyk, and Torsten Hoefler. Graph of thoughts: Solving elaborate problems with large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024. doi: 10.1609/aaai.v38i16.29720.

R Darrell Bock. A brief history of item theory response. *Educational measurement: issues and practice*, 16(4):21–33, 1997.

Philip J. Boland. Majority systems and the condorcet jury theorem. *Journal of the Royal Statistical Society. Series D (The Statistician)*, 38(3):181–189, 1989. ISSN 00390526, 14679884. URL http://www.jstor.org/stable/2348873.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33, 2020.

Ryan Sze-Yin Chan, Federico Nanni, Tomas Lazauskas, Rosie Wood, Penelope Yong, Lionel Tarassenko, Mark Girolami, James Geddes, and Andrew Duncan. Retrieval-augmented reasoning with lean language models. *The Alan Turing Institute*, 2025. doi: 10.5281/ZENODO.16408412.

Yanxi Chen, Xuchen Pan, Yaliang Li, Bolin Ding, and Jingren Zhou. Ee-llm: Large-scale training and inference of early-exit large language models with 3d parallelism. In *International Conference on Machine Learning*, pp. 7163–7189. PMLR, 2024.

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

A. P. Dawid and A. M. Skene. Maximum Likelihood Estimation of Observer Error-Rates Using the EM Algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):20–28, 1979.

Marie Jean Antoine Nicolas Caritat de Condorcet. *Essai sur l'application de l'analyse à la probabilité des décisions rendues à la pluralité des voix*. Imprimerie Royale, Paris, 1785. Reprint: AMS Chelsea, 1972.

Amir Dembo and Ofer Zeitouni. *Large Deviations Techniques and Applications*. Springer Berlin, Heidelberg, 2010.

Yichao Fu, Xuewei Wang, Yuandong Tian, and Jiawei Zhao. Deep think with confidence. *arXiv preprint arXiv:2508.15260*, 2025.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, et al. The Llama 3 Herd of Models. *arXiv preprint arXiv:2407.21783*, 2024.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

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

Yuzhen Huang, Weihao Zeng, Xingshan Zeng, Qi Zhu, and Junxian He. Pitfalls of rule-and model-based verifiers–a case study on mathematical reasoning. *arXiv preprint arXiv:2505.22203*, 2025.

Sungmin Kang, Yavuz Faruk Bakman, Duygu Nur Yaldiz, Baturalp Buyukates, and Salman Avestimehr. Uncertainty quantification for hallucination detection in large language models: Foundations, methodology, and future directions. *arXiv preprint arXiv:2510.12040*, 2025a.

Zhewei Kang, Xuandong Zhao, and Dawn Song. Scalable best-of-n selection for large language models via self-certainty. *arXiv preprint arXiv:2502.18581*, 2025b.

Hyun-Chul Kim and Zoubin Ghahramani. Bayesian classifier combination. In *Artificial Intelligence and Statistics*, pp. 619–627. PMLR, 2012.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *Advances in Neural Information Processing Systems*, 35: 22199–22213, 2022.

K.K. Ladha. The Condorcet Jury Theorem, Free Speech and Correlated Votes. *American Journal of Political Science*, 36(3):617–634, 1992.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models. In *Advances in Neural Information Processing Systems*, 2022.

Jia Li, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Huang, Kashif Rasul, Longhui Yu, Albert Q Jiang, Ziju Shen, et al. Numinamath: The largest public dataset in ai4maths with 860k pairs of competition math problems and solutions. *Hugging Face repository*, 13(9):9, 2024a.

Yiwei Li, Peiwen Yuan, Shaoxiong Feng, Boyuan Pan, Xinglin Wang, Bin Sun, Heda Wang, and Kan Li. Escape Sky-high Cost: Early-stopping Self-Consistency for Multi-step Reasoning. In *The Twelfth International Conference on Learning Representations*, 2024b.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*, 2023.

Christian List and Robert E. Goodin. Epistemic democracy: Generalizing the condorcet jury theorem. *Journal of Political Philosophy*, 9(3):277–306, 2001. ISSN 0963-8016. doi: 10.1111/1467-9760.00128.

Jiahao Liu, Qifan Wang, Jingang Wang, and Xunliang Cai. Speculative decoding via early-exiting for faster llm inference with thompson sampling control mechanism. *arXiv preprint arXiv:2406.03853*, 2024.

Qiang Liu, Jian Peng, and Alexander T Ihler. Variational inference for crowdsourcing. *Advances in neural information processing systems*, 25, 2012.

Xueguang Ma, Qian Liu, Dongfu Jiang, Ge Zhang, Zejun Ma, and Wenhu Chen. General-reasoner: Advancing llm reasoning across all domains. *arXiv preprint arXiv:2505.14652*, 2025.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-Refine: Iterative Refinement with Self-Feedback. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Rohin Manvi, Anikait Singh, and Stefano Ermon. Adaptive inference-time compute: Llms can predict if they can do better, even mid-generation. *arXiv preprint arXiv:2410.02725*, 2024.

OpenAI. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Mihir Prabhudesai, Lili Chen, Alex Ippoliti, Katerina Fragkiadaki, Hao Liu, and Deepak Pathak. Maximizing confidence alone improves reasoning. *arXiv preprint arXiv:2505.22660*, 2025.

Archiki Prasad, Weizhe Yuan, Richard Yuanzhe Pang, Jing Xu, Maryam Fazel-Zarandi, Mohit Bansal, Sainbayar Sukhbaatar, Jason E Weston, and Jane Yu. Self-consistency preference optimization. In *Forty-second International Conference on Machine Learning*, 2025.

Aaditya Ramdas and Ruodu Wang. Hypothesis testing with e-values. *Foundations and Trends® in Statistics*, 1(1-2):1–390, 2025. ISSN 2978-4212. doi: 10.1561/3600000002.

Vikas C Raykar, Shipeng Yu, Linda H Zhao, Gerardo Hermosillo Valadez, Charles Florin, Luca Bogoni, and Linda Moy. Learning from crowds. *Journal of machine learning research*, 11(4), 2010.

Sheikh Shafayat, Fahim Tajwar, Ruslan Salakhutdinov, Jeff Schneider, and Andrea Zanette. Can large reasoning models self-train? *arXiv preprint arXiv:2505.21444*, 2025.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. *arXiv preprint arXiv:2402.03300*, 2024.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. HybridFlow: A Flexible and Efficient RLHF Framework. *arXiv preprint arXiv: 2409.19256*, 2024.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems*, 36:8634–8652, 2023.

Wei Song, Zhenya Huang, Cheng Cheng, Weibo Gao, Bihan Xu, GuanHao Zhao, Fei Wang, and Runze Wu. Irt-router: Effective and interpretable multi-llm routing via item response theory. *arXiv preprint arXiv:2506.01048*, 2025.

Yunhao Tang, Kunhao Zheng, Gabriel Synnaeve, and Remi Munos. Optimizing Language Models for Inference Time Objectives using Reinforcement Learning. In *Forty-second International Conference on Machine Learning*, 2025.

Leitian Tao, Ilia Kulikov, Swarnadeep Saha, Tianlu Wang, Jing Xu, Yixuan Li, Jason E Weston, and Ping Yu. Hybrid reinforcement: When reward is sparse, it's better to be dense. *arXiv preprint arXiv:2510.07242*, 2025.

Jean Ville. *Étude critique de la notion de collectif*. Gauthier-Villars, 1939.

Guangya Wan, Yuqi Wu, Jie Chen, and Sheng Li. Reasoning Aware Self-Consistency: Leveraging Reasoning Paths for Efficient LLM Sampling. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 3613–3635, 2025.

Pei-Hsin Wang, Sheng-Iou Hsieh, Shih-Chieh Chang, Yu-Ting Chen, Jia-Yu Pan, Wei Wei, and Da-Chang Juan. Contextual temperature for language modeling. *arXiv preprint arXiv:2012.13575*, 2020.

Xinglin Wang, Shaoxiong Feng, Yiwei Li, Peiwen Yuan, Yueqi Zhang, Chuyi Tan, Boyuan Pan, Yao Hu, and Kan Li. Make every penny count: Difficulty-adaptive self-consistency for cost-efficient reasoning. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 6904–6917, 2025a.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Maarten Bosma, Ed H. Chi, Quoc V. Le, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022.

Yiming Wang, Pei Zhang, Siyuan Huang, Baosong Yang, Zhuosheng Zhang, Fei Huang, and Rui Wang. Sampling-efficient test-time scaling: Self-estimating the best-of-n sampling in early decoding. *arXiv preprint arXiv:2503.01422*, 2025b.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35, 2022.

Peter Welinder, Steve Branson, Pietro Perona, and Serge Belongie. The multidimensional wisdom of crowds. *Advances in neural information processing systems*, 23, 2010.

Jiaxin Wen, Zachary Ankner, Arushi Somani, Peter Hase, Samuel Marks, Jacob Goldman-Wetzler, Linda Petrini, Henry Sleight, Collin Burns, He He, et al. Unsupervised elicitation of language models. *arXiv preprint arXiv:2506.10139*, 2025.

Jacob Whitehill, Ting-fan Wu, Jacob Bergsma, Javier Movellan, and Paul Ruvolo. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. *Advances in neural information processing systems*, 22, 2009.

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.*, 8(3–4):229–256, 1992.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, Keming Lu, Mingfeng Xue, Runji Lin, Tianyu Liu, Xingzhang Ren, and Zhenru Zhang. Qwen2.5-Math Technical Report: Toward Mathematical Expert Model via Self-Improvement. *arXiv preprint arXiv:2409.12122*, 2024.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 Technical Report. *arXiv preprint arXiv:2412.15115*, 2025.

Peiran Yao, Jerin George Mathew, Shehraj Singh, Donatella Firmani, and Denilson Barbosa. A bayesian approach towards crowdsourcing the truths from llms. In *NeurIPS 2024 Workshop on Bayesian Decision-making and Uncertainty*, 2024.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems*, 2023.

Yuxin Zuo, Kaiyan Zhang, Shang Qu, Li Sheng, Xuekai Zhu, Biqing Qi, Youbang Sun, Ganqu Cui, Ning Ding, and Bowen Zhou. TTRL: Test-Time Reinforcement Learning. *arXiv preprint arXiv:2504.16084*, 2025.

A PROOFS OF SECTION 2

A.1 SMALL SAMPLE REGIME

Hoeffding, Bernstein, and Chernoff–Markov bounds become less effective in the small-sample regime, i.e., when the number of voters satisfies $n \lesssim 50$. In this setting, the exact error probability can be obtained by leveraging the properties of the multinomial distribution. We provide below an efficient dynamic programming (DP) approach to compute this probability.

A.1.1 DYNAMIC PROGRAMMING FOR EXACT MULTINOMIAL PROBABILITIES

For each category j, define

$$P_j(x) = \frac{p_j^x}{x!}, \quad x = 0, \dots, n,$$

and store the values $P_j = (P_j(0), \dots, P_j(n))$ in an array of length $n + 1$. The entries can be generated iteratively using the recurrence

$$P_j(x+1) = \frac{p_j}{(x+1)} P_j(x).$$

After processing a subset of the rival categories, we define a state of the dynamic program as

$$\text{state } (t, m, s) \quad \text{with} \quad \begin{cases} t & = \text{votes for the true category } c^\star, \\ m & = \text{current } \textit{maximum} \text{ vote count among the rivals processed so far,} \\ s & = \textit{total} \text{ vote count allocated to processed categories.} \end{cases}$$

Formally, the DP table is

$$\mathrm{DP}_i(t, m, s) \;=\; \frac{1}{s!}\,\mathbb{P}\Big[N_{c^\star}^{(s)} = t,\; \max_{j \in \{1,\dots,i\} \setminus \{c^\star\}} N_j^{(s)} = m,\; \sum_{j \in \{c^\star, 1, \dots, i\}} N_i^{(s)} = s\Big],$$

where i denotes the number of categories processed.

Initial table. Before incorporating any rivals, we only consider the true category $c^\star$

$$\mathrm{DP}_1(t, 0, t) = P_{c^\star}(t), \qquad t = 0, \dots, n.$$

since the maximum vote count among zero rivals is naturally 0.

Transition when adding a new rival j. Suppose we have already computed $\mathrm{DP}_i(\cdot,\cdot,\cdot)$. We now incorporate category j. For each triple (t, m, s), we consider vote counts $x = 0, \dots, n - s$ drawn from $P_j(x)$ and define

$$\text{newMax} = \max\{m, x\}.$$

The DP table is updated according to

$$\mathrm{DP}_{i+1}(t, \text{newMax}, s + x) \mathrel{+}= \; \mathrm{DP}_i(t, m, s)\; P_j(x).$$

This implementation stores states (t, m, s) with $t + m \leq s \leq n$, and transitions $x = 0, \dots, n - s$. Because of these constraints, the total number of reachable states is $\mathcal{O}(n^3)$, rather than the naive $\mathcal{O}(n^4)$. Iterating over all k categories therefore yields a worst-case time complexity of

$$\mathcal{O}(kn^3).$$

The memory complexity is $\mathcal{O}(n^3)$.

After processing all $k - 1$ rivals, the DP table $\mathrm{DP}_k(\cdot,\cdot,\cdot)$ is complete. The error probability is then obtained by summing over all states where the true category does not have a strict majority and the total number of votes is equal to n,

$$\mathbb{P}(\hat{c}_n \neq c^\star) \;=\; \sum_{t=0}^{n} \sum_{m=t}^{n} n!\, \mathrm{DP}_k(t, m, n).$$

For large values of n, factorial terms may cause numerical underflow or overflow. To prevent this, we compute the entries of the DP table in log space.

A.2 Hoeffding bound

Proof. For each rival $j \neq c^\star$, consider the random variable

$$Z_j^{(n)} = N_{c^\star}^{(n)} - N_j^{(n)} = \sum_{i=1}^{n} \Big(\mathbf{1}\{X_i = c^\star\} - \mathbf{1}\{X_i = j\} \Big). \tag{10}$$

The summands $Y_i^j = \mathbf{1}\{X_i = c^\star\} - \mathbf{1}\{X_i = j\}$ are independent identically distributed random variables, bounded in $[-1, 1]$, with expected value $\mu_j = p_{c^\star} - p_j > 0$. Applying Hoeffding's inequality we obtain

$$\mathbb{P}\big[Z_j^{(n)} \leq 0\big] \leq \mathbb{P}\left[\frac{Z_j^{(n)}}{n} - \mu_j \leq -\mu_j\right] = \mathbb{P}\left[-Z_j^{(n)} + \mathbb{E}[Z_j^{(n)}] \geq \mathbb{E}[Z_j^{(n)}]\right] \leq \exp\left(-\frac{n}{2}(p_{c^\star} - p_j)^2\right).$$

The event $\{\hat{c}_n \neq c^\star\}$ implies $Z_j^{(n)} \leq 0$ for some $j \neq c^\star$. Thus,

$$\mathbb{P}\big[\hat{c}_n \neq c\big] \leq \sum_{j \neq c} \mathbb{P}\big[Z_j^{(n)} \leq 0\big] \leq \sum_{j \neq c} \exp\left(-\frac{n}{2}(p_{c^\star} - p_j)^2\right),$$

which establishes the exponential bound. This can be further simplified as

$$\mathbb{P}\big[\hat{c}_n \neq c\big] \leq (k-1) \exp\left(-\frac{n}{2} \min_{j \neq c} (p_{c^\star} - p_j)^2\right).$$

Since the upper bound decays to 0 as $n \to \infty$, and k is finite, we obtain

$$\mathbb{P}[\hat{c}_n = c^\star] \to 1 \quad \text{as} \quad n \to \infty.$$

□

A.2.1 Weighted majority vote for experts with different accuracies

We now consider a setting where we have access to multiple models with varying expertise: some are cheaper but less accurate, while others are more expensive but more precise. To capture this heterogeneity, we assign each expert a weight, denoted by ω_ℓ, that reflects its reliability. Specifically, we assume there are L experts, and expert ℓ contributes n_ℓ samples.

Instead of using a simple majority vote, we aggregate predictions via a weighted majority vote

$$\hat{c}_n^\omega = \arg\max_j \sum_{\ell=1}^{L} \omega_\ell N_j^{(n_\ell)},$$

where $N_j^{(n_\ell)} = \sum_{i=1}^{n^\ell} \mathbf{1}\big\{X_i^{(\ell)} = j\big\}$ is the number of samples from expert ℓ predicting label j. The total sample size is $n = \sum_\ell n_\ell$.

In this setting, the data across experts are non-exchangeable, since each expert has a different own distribution over labels.

Error bound for weighted majority voting. We derive an error bound using Hoeffding's inequality. For every rival $j \neq c^\star$, define the weighted margin

$$Z_{j,\omega}^{(n)} = N_{c,\omega}^{(n)} - N_{j,\omega}^{(n)} = \sum_{\ell=1}^{L} \sum_{i=1}^{n_\ell} \omega_\ell \left(\mathbf{1}\{X_i^{(\ell)} = c^\star\} - \mathbf{1}\{X_i^{(\ell)} = j\} \right).$$

Each summand $Y_{i,\ell}^j = \omega_\ell \left(\mathbf{1}\{X_i^{(\ell)} = c^\star\} - \mathbf{1}\{X_i^{(\ell)} = j\} \right)$ is independent and bounded between $-\omega_\ell$ and ω_ℓ. The sum $Z_{j,\omega}^{(n)}$ has mean

$$\mathbb{E}\left[Z_{j,\omega}^{(n)}\right] = \sum_{\ell=1}^{L} n_\ell \, \omega_\ell \left(p_{c^\star}^\ell - p_j^\ell\right).$$

Applying Hoeffding's inequality, we obtain

$$\begin{aligned}
\mathbb{P}\left[\hat{c}_n^{\omega} \neq c^\star\right] &\leq \sum_{j\neq c} \mathbb{P}\left[Z_{j,\omega}^{(n)} \leq 0\right] \leq \sum_{j \neq c^\star} \exp\left(-\frac{1}{2}\frac{\left(\sum_{\ell=1}^{L} n_\ell\, \omega_\ell \left(p_{c^\star}^{\ell} - p_j^{\ell}\right)\right)^2}{\sum_{\ell=1}^{L} n_\ell\, \omega_\ell^2}\right) \\
&\leq (k-1)\exp\left(-\frac{1}{2}\frac{\left(\sum_{\ell=1}^{L} n_\ell\, \omega_\ell \min_{j\neq c^\star}\left(p_{c^\star}^{\ell} - p_j^{\ell}\right)\right)^2}{\sum_{\ell=1}^{L} n_\ell\, \omega_\ell^2}\right) \\
&\leq (k-1)\exp\left(-\frac{1}{2}\sum_{\ell=1}^{L} n_\ell \frac{n_\ell\, \omega_\ell^2}{\sum_{\ell=1}^{L} n_\ell\, \omega_\ell^2} \min_{j\neq c^\star}\left(p_{c^\star}^{\ell} - p_j^{\ell}\right)^2\right).
\end{aligned}$$

If each expert contributes the same number of sample ($n_\ell = n$), the previous bound can be simplified as

$$\begin{aligned}
\mathbb{P}\left[\hat{c}_n^{\omega} \neq c^\star\right] &\leq \sum_{j\neq c^\star} \mathbb{P}\left[Z_{j,\omega}^{(n)} \leq 0\right] \leq \sum_{j \neq c^\star} \exp\left(-\frac{n}{2}\frac{\left(\sum_{\ell=1}^{L} \omega_\ell \left(p_{c^\star}^{\ell} - p_j^{\ell}\right)\right)^2}{\sum_{\ell=1}^{L} \omega_\ell^2}\right) \\
&\leq (k-1)\exp\left(-\frac{n}{2}\sum_{\ell=1}^{L}\left(\frac{\omega_\ell^2}{\sum_{\ell=1}^{L} \omega_\ell^2}\right) \min_{j\neq c^\star}\left(p_{c^\star}^{\ell} - p_j^{\ell}\right)^2\right).
\end{aligned}$$

Optimal weights based on expert accuracy. Recall that our decision rule maps the set of expert responses X to a final answer. We say a decision rule is *optimal* if it minimises the probability of error. Formally, letting D denote the rule, we want to minimise

$$\mathbb{P}\left[D(X) \neq c^\star\right], \quad X = (X_{i_\ell}^{(\ell)}),\ \ell = 1, \ldots, L, \quad i_\ell = 1, \ldots, n_\ell,$$

where $c^\star$ is the true answer. To derive the optimal decision rule, we make the following assumptions.

A 1. *Independence: conditioned on the ground-truth label $c^\star$, the random variables $X_i^{(\ell)}$, corresponding to the i-th response from expert ℓ, are mutually independent across both experts and repetitions.*

A 2. *Unbiased truth: the ground-truth label is uniformly distributed, i.e. $\mathbb{P}[c^\star = j] = 1/k$ for $j = 1, \ldots, k$.*

Suppose that we know the confusion matrix $\left(C_{ij}^{(\ell)}\right)_{ij}$ for each expert ℓ, where

$$C_{ij}^{(\ell)} = \mathbb{P}\left[X^{(\ell)} = i \middle| c^\star = j\right]$$

denotes the probability that model ℓ will record value i given j is the true response. Then, the decision rule that minimises the Bayes risk coincides with the Maximum a Posteriori (MAP) rule,

$$D^{\text{OPT}}(X) = \arg\max_j \ \log \mathbb{P}[c^\star = j | X].$$

By Bayes' theorem we have

$$\begin{aligned}
\arg\max_j \ \mathbb{P}[c^\star = j|X] &= \arg\max_j \ \mathbb{P}[c^\star = j]\mathbb{P}[X|c^\star = j] \\
&= \arg\max_j \ \prod_{\ell=1}^{L}\prod_{i=1}^{n_\ell} \mathbb{P}\left[X_i^{(\ell)} \middle| c^\star = j\right] = \prod_{\ell=1}^{L}\prod_{i=1}^{n_\ell} C_{j\ X_i^{(\ell)}}^{(\ell)},
\end{aligned}$$

which results into

$$D^{\text{OPT}}(X) = \arg\max_j \ \sum_{\ell=1}^{L}\sum_{i=1}^{n_\ell} \log C_{j\ X_i^{(\ell)}}^{(\ell)}.$$

Now, imagine that we only know each expert's overall competence level $q_\ell \in (0,1)$, defined as the probability of correctly predicting the true label,

$$q_\ell = \mathbb{P}\left[X^{(\ell)} = j \middle| c^\star = j\right],$$

but not the full confusion matrix. A natural approximation is to assume that errors are distributed uniformly across the $k-1$ incorrect labels, i.e.

$$\mathbb{P}[X^{(\ell)} = i \big| c^\star = j \neq i] = \frac{1-q_\ell}{k-1}.$$

Without this approximation, one would need to estimate the full confusion matrices. This can be done, for example, via the Expectation–Maximisation algorithm (Dawid & Skene, 1979).

A.3 BERNSTEIN BOUND

Proof. Let the random variable $Y_i^j = \mathbf{1}\{X_i = c^\star\} - \mathbf{1}\{X_i = j\}$. We have that

$$\mathbb{E}\left[Y_i^j - (p_{c^\star} - p_j)\right] = 0,$$

$$\left|Y_i^j - (p_{c^\star} - p_j)\right| = |\mathbf{1}\{X_i = c\} - \mathbf{1}\{X_i = j\} - (p_{c^\star} - p_j)| \leq 1 + (p_{c^\star} - p_j),$$

and

$$\sigma_j^2 = \mathbb{E}\left[\left(Y_i^j - (p_{c^\star} - p_j)\right)^2\right] = p_{c^\star} + p_j - (p_{c^\star} - p_j)^2.$$

Consider $Z_j^{(n)} = \sum_i^n Y_i^j$. Applying Bernstein's inequality gives

$$\mathbb{P}\big[Z_j^{(n)} \leq 0\big] \leq \mathbb{P}\left[-\frac{Z_j^{(n)}}{n} + \mu_j \geq \mu_j\right] \leq \exp\left(-\frac{n(p_{c^\star} - p_j)^2}{2\sigma_j^2 + \frac{2}{3}(p_{c^\star} - p_j) + \frac{2}{3}(p_{c^\star} - p_j)^2}\right).$$

Since the event $\{\hat{c}_n \neq c^\star\}$ implies $Z_j^{(n)} \leq 0$ for some $j \neq c^\star$, we obtain the bound

$$\mathbb{P}\big[\hat{c}_n \neq c^\star\big] \leq \sum_{j \neq c^\star} \mathbb{P}\big[Z_j^{(n)} \leq 0\big] \leq \sum_{j \neq c^\star} \exp\left(-\frac{n(p_{c^\star} - p_j)^2}{2\sigma_j^2 + \frac{2}{3}(p_{c^\star} - p_j) + \frac{2}{3}(p_{c^\star} - p_j)^2}\right).$$

Noting that $p_{c^\star} + p_j \leq 1$, we can obtain a simpler but looser bound

$$\mathbb{P}[\hat{c}_n \neq c^\star] \leq \sum_{j \neq c^\star} \exp\left(-\frac{n(p_{c^\star} - p_j)^2}{2\left(1 - \frac{2}{3}(p_{c^\star} - p_j)^2\right) + \frac{2}{3}(p_{c^\star} - p_j)}\right),$$

which only depends on the probability gaps $\delta_j = p_{c^\star} - p_j$. □

A.4 CHERNOFF-MARKOV BOUND

Proof. Using the Chernoff-Markov inequality, for any $\lambda < 0$ we have

$$\mathbb{P}\left[Z_j^{(n)} \leq 0\right] = \mathbb{P}\left[e^{\lambda Z_j^{(n)}} \geq 1\right] \leq \mathbb{E}\left[e^{\lambda Z_j^{(n)}}\right] = \left(\mathbb{E}\left[e^{\lambda Y_1^j}\right]\right)^n,$$

where we have used that the random variables Y_i^j are independent and identically distributed. The moment generating function of Y_1^j is

$$M(\lambda) = \mathbb{E}\left[e^{\lambda Y_1^j}\right] = p_{c^\star} e^{\lambda} + p_j e^{-\lambda} + 1 - (p_{c^\star} + p_j).$$

We now optimise $M(\lambda)$ over $\lambda < 0$. Since $p_{c^\star} > p_j$, there is no minimiser in $\lambda > 0$, so we can just extend the optimisation to $\lambda \in \mathbb{R}$. Setting the derivative to zero,

$$M'(\lambda) = p_{c^\star} e^{\lambda} - p_j e^{-\lambda} = 0$$

gives the minimiser

$$\lambda^\star = \frac{1}{2} \log(p_j / p_{c^\star}) < 0$$

with corresponding value

$$M(\lambda^*) = 1 - (p_{c^\star} + p_j) + 2\sqrt{p_{c^\star} p_j}.$$

Thus, the Chernoff-Markov bound becomes

$$\begin{aligned}\mathbb{P}\left[Z_j^{(n)} \leq 0\right] &\leq \left(1-(p_{c^\star}+p_j)+2\sqrt{p_{c^\star}p_j}\right)^n = \exp\left(n\log\left(1-(p_{c^\star}+p_j)+2\sqrt{p_{c^\star}p_j}\right)\right)\\ &= \exp\left(n\log\left(1-(\sqrt{p_{c^\star}}-\sqrt{p_j})^2\right)\right).\end{aligned}$$

Consequently, we have

$$\mathbb{P}\big[\hat{c}_n \neq c^\star\big] \leq \sum_{j\neq c^\star}\mathbb{P}\big[Z_j^{(n)} \leq 0\big] \leq \sum_{j\neq c^\star}\exp\left(n\log\left(1-(\sqrt{p_{c^\star}}-\sqrt{p_j})^2\right)\right).$$

□

A.5 CLT bound

Proof. For each rival j, consider the random variable $Z_j^{(n)}$ defined in (10), which is a sum of independent and identically distributed random variables with mean

$$\mu_j = p_{c^\star} - p_j$$

and variance

$$\sigma_j^2 = p_{c^\star} + p_j - (p_{c^\star}-p_j)^2.$$

By the central limit theorem,

$$\frac{Z_j^{(n)} - n(p_{c^\star}-p_j)}{\sqrt{n}} \xrightarrow{d} \mathcal{N}(0,\sigma_j^2).$$

Therefore, as $n\to\infty$, we have

$$\mathbb{P}\left[Z_j^{(n)} \leq 0\right] = \mathbb{P}\left[\frac{Z_j^{(n)} - n(p_{c^\star}-p_j)}{\sigma_j\sqrt{n}} \leq -\frac{(p_{c^\star}-p_j)\sqrt{n}}{\sigma_j}\right] = \Phi\Big(-\frac{(p_{c^\star}-p_j)\sqrt{n}}{\sigma_j}\Big)\,[1+O(n^{-1/2})],$$

where Φ denotes the CDF of a standard Gaussian random variable.

Majority voting fails if $Z_j^{(n)} \leq 0$ for some $j \neq c^\star$. Applying the union bound, we obtain

$$\mathbb{P}[\hat{c}_n \neq c^\star] \;\leq\; \sum_{j\neq c^\star}\mathbb{P}\left[Z_j^{(n)} \leq 0\right] \;=\; \sum_{j\neq c^\star}\Phi\Big(-\frac{(p_{c^\star}-p_j)\sqrt{n}}{\sigma_j}\Big)[1+O(n^{-1/2})].$$

To bound the Gaussian tail, we use Craig's formula

$$\Phi(-x) = \frac{1}{\pi}\int_0^{\pi/2}\exp\left(-\frac{x^2}{2\sin^2\theta}\right)d\theta \leq \frac{1}{2}e^{-\frac{x^2}{2}}, \qquad \text{for } x>0.$$

Substituting this bound gives

$$\mathbb{P}[\hat{c}_n \neq c^\star] \;\leq\; \frac{1}{2}\sum_{j\neq c^\star}\exp\left(-\frac{n}{2}\left(\frac{p_{c^\star}-p_j}{\sigma_j}\right)^2\right) \leq \frac{1}{2}(k-1)\exp\left(-\frac{n}{2}\min_{j\neq c^\star}\left(\frac{p_{c^\star}-p_j}{\sigma_j}\right)^2\right).$$

For fixed $p_{c^\star}$, the ratio $\frac{p_{c^\star}-p_j}{\sigma_j}$ is monotonically decreasing in p_j. Therefore, the smallest value, and hence the slowest exponential decay, is attained at the rival with the largest probability among the competitors. Denoting this *second-largest* vote probability by

$$p_{j^\star} = \max_{j\neq c^\star} p_j,$$

the convergence rate in the exponential bound above is determined by

$$\kappa = \frac{\delta}{2p_{c^\star}-\delta-\delta^2}, \qquad \delta = p_{c^\star} - p_{j^\star}.$$

Thus, the competitor that most threatens the accuracy of majority voting is precisely the category with the second–highest support.

The previous bound derived from the central limit theorem can be sharpened by incorporating two classical corrections. The first correction is the continuity term, that is, the correction term due to discreteness. Since the random variable $Z_j^{(n)}$ takes values in the discrete set $\{-n, \dots, n\}$, the event $Z_j^{(n)} \leq 0$ is equivalent to $Z_j^{(n)} \leq 1/2$. Hence,

$$\mathbb{P}\left[Z_j^{(n)} \leq 0\right] = \mathbb{P}\left[Z_j^{(n)} \leq 1/2\right].$$

Applying the central limit theorem approximation then yields, as $n \to \infty$,

$$\begin{aligned}\mathbb{P}\left[Z_j^{(n)} \leq 1/2\right] &= \mathbb{P}\left[\frac{Z_j^{(n)} - n(p_{c^\star} - p_j)}{\sigma_j\sqrt{n}} \leq \frac{1}{2\sigma_j\sqrt{n}} - \frac{\sqrt{n}(p_{c^\star} - p_j)}{\sigma_j}\right] \\ &\approx \Phi\left(\frac{\sqrt{n}(p_{c^\star} - p_j) - 1/(2\sqrt{n})}{\sigma_j}\right) = \frac{1}{2}\mathrm{erfc}\left(\frac{\sqrt{n}(p_{c^\star} - p_j) - 1/(2\sqrt{n})}{\sqrt{2}\,\sigma_j}\right).\end{aligned}$$

A further refinement comes from the Berry–Esseen theorem, which quantifies the uniform error of the central limit theorem approximation. In particular, for all n

$$\left|\mathbb{P}\left[\frac{Z_j^{(n)} - n(p_{c^\star} - p_j)}{\sigma_j\sqrt{n}} \leq x\right] - \Phi(x)\right| \leq \frac{C\rho_j}{\sigma_j^3\sqrt{n}},$$

where ρ_j denotes the third central moment,

$$\rho_j = \mathbb{E}\left[\left(Y_i^j - (p_{c^\star} - p_j)\right)^3\right] = (p_{c^\star} - p_j)(1 - 3(p_{c^\star} + p_j) + 2(p_{c^\star} - p_j)^2)$$

and $C \leq 0.56$ is a universal constant. Incorporating both corrections, we obtain the refined bound

$$\mathbb{P}[\hat{c}_n \neq c^\star] \leq \sum_{j \neq c^\star} \frac{1}{2}\mathrm{erfc}\left(\frac{\sqrt{n}(p_{c^\star} - p_j) - 1/(2\sqrt{n})}{\sqrt{2}\,\sigma_j}\right) + \frac{0.56\rho_j}{\sigma_j^3\sqrt{n}}.$$

□

A.6 Large-deviations regime

Proof. Let $\mathbf{p} = (p_1, \dots, p_k)$ denote the true probability distribution, and let $\hat{\mathbf{p}}_\mathbf{n} = (\hat{p}_{n,1}, \dots, \hat{p}_{n,k})$ be the empirical measure, where $\hat{p}_{n,j}$ are the empirical frequencies for each category

$$\hat{p}_{n,j} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}\{X_i = j\}.$$

Recall that $p_{c^\star} = \max_j p_j$. Define the set $\mathcal{B} \subseteq \Delta_k$ by

$$\mathcal{B} = \left\{\mathbf{q} \in \Delta_k :\ q_{c^\star} \leq \max_{j \neq c^\star} q_j\right\} = \{\mathbf{q} \in \Delta_k : q_{c^\star} \leq q_j,\ \text{for some } j \neq c^\star\}. \tag{11}$$

Step 1. Sanov upper bound. Sanov's theorem (large–deviation principle for types) states that for any Borel set $\mathcal{A} \subseteq \Delta_k$,

$$-\inf_{\mathbf{q} \in \mathring{\mathcal{A}}} D_{\mathrm{KL}}(\mathbf{q}\|\mathbf{p}) \leq \liminf_{n\to\infty} \frac{1}{n}\log\mathbb{P}\left(\hat{\mathbf{p}}_n \in \mathcal{A}\right) \leq \limsup_{n\to\infty} \frac{1}{n}\log\mathbb{P}\left(\hat{\mathbf{p}}_n \in \mathcal{A}\right) \leq -\inf_{\mathbf{q} \in \overline{\mathcal{A}}} D_{\mathrm{KL}}(\mathbf{q}\|\mathbf{p}),$$

where $\mathring{\mathcal{A}}$ and $\overline{\mathcal{A}}$ denote the interior and closure of $\mathcal{A}$, respectively.

For our purposes, let $\mathcal{A} = \mathcal{B}$ as defined in Eq. (11). Then

$$\mathring{\mathcal{B}} = \{\mathbf{q} \in \Delta_k : q_{c^\star} < q_j \text{ for some } j \neq c^\star\}, \quad \overline{\mathcal{B}} = \mathcal{B},$$

since $\mathcal{B}$ is closed.

Step 2. Error event as a type set. The majority rule is *incorrect* (i.e. $\hat{c}_n = \arg\max_j \ \hat{p}_{n,j} \neq c^\star$) exactly when $\hat{\mathbf{p}}_n \in \mathcal{B}$. Hence, applying Sanov's bounds yields

$$\mathbb{P}\left[\hat{c}_n \neq c^\star\right] = \exp\left(-n \inf_{\mathbf{q}\in\mathcal{B}} D_{\mathrm{KL}}(\mathbf{q}\|\mathbf{p}) + o(n)\right).$$

Step 3. Positivity of the exponent. If $p_{c^\star} > p_j$ for every $j \neq c^\star$, then the true distribution satisfies $\mathbf{p} \notin \mathcal{B}$. The infimum of the KL divergence over $\mathcal{B}$ is therefore attained on the boundary, i.e., at some $\mathbf{q}^\star \in \mathcal{B}$ with $q^\star_{c^\star} = q^\star_j$ for some $j \neq c^\star$. Thus, the large-deviation exponent is

$$I^\star(\mathbf{p}) = \min_{j\neq c^\star} \inf_{\mathbf{q}:\, q_{c^\star}=q_j} D_{\mathrm{KL}}(\mathbf{q}\|\mathbf{p}) > 0,$$

and the error probability decays exponentially

$$\mathbb{P}[\hat{c}_n \neq c^\star] = \exp(-nI^\star(\mathbf{p}) + o(n)).$$

□

A.6.1 Sanov exponent

The Sanov exponent $I^\star(\mathbf{p})$ admits a closed-form expression. We provide a detailed derivation below.

Recall that our objective is to compute

$$I^\star(\mathbf{p}) = \inf_{\mathbf{q}\in\mathcal{B}} D_{\mathrm{KL}}(\mathbf{q}\|\mathbf{p}), \qquad \mathcal{B} = \{\mathbf{q} \in \Delta_k : \ q_{c^\star} \leq \max_{j\neq c^\star} q_j\}. \tag{12}$$

Let the runner-up (second-largest competitor) be

$$j^\star = \arg\max_{j\neq c^\star} \ p_j.$$

Then the optimisation problem can be equivalently written as

$$I^\star(\mathbf{p}) = \min_{\mathbf{q}\in\Delta_k} \left[\sum_{i=1}^{k} q_i \log\left(\frac{q_i}{p_i}\right) \ : q_{j^\star} \geq q_{c^\star}\right].$$

Introducing Lagrange multipliers, we define the Lagrangian

$$\mathcal{L}(q, \lambda, \mu) = \sum_{i=1}^{k} q_i \log \frac{q_i}{p_i} + \lambda\left(\sum_{i=1}^{k} q_i - 1\right) + \mu(q_{c^\star} - q_{j^\star}),$$

where $\lambda \in \mathbb{R}$ and $\mu \geq 0$, to enforce $q_{c^\star} \leq q_{j^\star}$. The first-order conditions yield

$$\partial_{q_i}\mathcal{L} = \log q_i + 1 - \log p_i + \lambda = 0, \qquad \text{for} \quad i \neq c^\star, j^\star,$$

which implies

$$q_i = p_i e^{-(1+\lambda)}, \qquad \text{for} \quad i \neq c^\star, j^\star.$$

Similarly, for $q_{c^\star}$ and $q_{j^\star}$ we obtain

$$q_{c^\star} = p_{c^\star} e^{-(1+\lambda+\mu)}, \qquad q_{j^\star} = p_{j^\star} e^{-(1+\lambda-\mu)}.$$

Defining $Z = e^{-(1+\lambda)}$ and $s = e^{\mu}$, we can rewrite the solution as

$$\begin{aligned} q_i &= p_i Z, \qquad i \neq c^\star, j^\star \\ q_{c^\star} &= p_{c^\star} Z/s \\ q_{j^\star} &= p_{j^\star} Z s. \end{aligned}$$

Solving for s, we have

$$s = \sqrt{\frac{q_{j^\star}}{p_{j^\star}} \frac{p_{c^\star}}{q_{c^\star}}}.$$

Enforcing the constraint $q_{j^\star} \geq q_{c^\star}$ gives

$$s \geq \sqrt{\frac{p_{c^\star}}{p_{j^\star}}}.$$

On the other hand, enforcing the simplex constraint $\sum_i q_i = 1$ gives

$$Z\left[(1 - p_{c^\star} - p_{j^\star}) + \frac{p_{c^\star}}{s} + p_{j^\star} s\right] = 1.$$

Note that $Z > 0$. Substituting this, the KL divergence can be expressed as a function of s (since Z itself depends on s)

$$\begin{aligned} D_{\mathrm{KL}}(\mathbf{q}(s)\|\mathbf{p}) &= Z \log Z \left[(1 - p_{c^\star} - p_{j^\star}) + \frac{p_{c^\star}}{s} + p_{j^\star} s\right] + Z \log s \left(p_{j^\star} s - \frac{p_{c^\star}}{s}\right) \\ &= \log Z + Z \log s \left(p_{j^\star} s - \frac{p_{c^\star}}{s}\right). \end{aligned}$$

Minimising over $\mathbf{q}$ is equivalent to optimising over $s \geq \sqrt{p_{c^\star}/p_{j^\star}}$. Focusing on the first term, we observe that

$$\frac{d}{ds} \log Z = -\frac{-p_{c^\star}/s^2 + p_{j^\star}}{((1 - p_{c^\star} - p_{j^\star}) + p_{c^\star}/s + p_{j^\star} s)} \leq 0, \qquad \text{for } s \geq \sqrt{p_{c^\star}/p_{j^\star}}.$$

Therefore, $\log Z$ is strictly decreasing for $s \in \left(\sqrt{p_{c^\star}/p_{j^\star}}, \infty\right)$.

Furthermore, the derivative of the KL divergence with respect to s is

$$\frac{d}{ds} D_{\mathrm{KL}}(\mathbf{q}(s)\|\mathbf{p}) = Z \log s \left(p_{j^\star} + \frac{p_{c^\star}}{s^2} - Zs\left(p_{j^\star} - \frac{p_{c^\star}}{s^2}\right)^2\right), \qquad s \geq \sqrt{p_{c^\star}/p_{j^\star}} > 0.$$

Note that

$$\begin{aligned} s\left[\left(p_{j^\star} + \frac{p_{c^\star}}{s^2}\right) - Z\left(p_{j^\star} - \frac{p_{c^\star}}{s^2}\right)^2\right] &\geq \left[\left(p_{j^\star} s + \frac{p_{c^\star}}{s}\right) - \frac{(p_{j^\star} s - p_{c^\star}/s)^2}{p_{j^\star} s + p_{c^\star}/s}\right] \\ &\geq \left[\left(p_{j^\star} s + \frac{p_{c^\star}}{s}\right) - \frac{(p_{j^\star} s + p_{c^\star}/s)^2}{p_{j^\star} s + p_{c^\star}/s}\right] \geq 0. \end{aligned}$$

In particular, for $s > \sqrt{p_{c^\star}/p_{j^\star}}$

$$s\left[\left(p_{j^\star} + \frac{p_{c^\star}}{s^2}\right) - Z\left(p_{j^\star} - \frac{p_{c^\star}}{s^2}\right)^2\right] > 0.$$

Using this together with the fact that $Z > 0$ and $\log s > 0$ (since $s > 1$), it follows that

$$\frac{d}{ds} D_{\mathrm{KL}}(\mathbf{q}(s)\|\mathbf{p}) > 0, \qquad \text{for } s > \sqrt{p_{c^\star}/p_{j^\star}}.$$

Hence, $D_{\mathrm{KL}}(\mathbf{q}(s)\|\mathbf{p})$ is strictly increasing for $s \in (\sqrt{p_{c^\star}/p_{j^\star}}, \infty)$. The minimum is therefore attained at $s = \sqrt{p_{c^\star}/p_{j^\star}}$, which leads to

$$\begin{aligned} I^\star(\mathbf{p}) = \min_{\substack{\mathbf{q} \in \Delta_k \\ q_{j^\star} \geq q_{c^\star}}} D_{\mathrm{KL}}(\mathbf{q}\|\mathbf{p}) &= -\log\left(1 - p_{c^\star} - p_{j^\star} + 2\sqrt{p_{c^\star} p_{j^\star}}\right) \\ &= -\log\left(1 - \left(\sqrt{p_{c^\star}} - \sqrt{p_{j^\star}}\right)^2\right). \end{aligned}$$

Remark A.1.

1. *If there are multiple runners-up, there may be several minimisers* $\mathbf{q}^\star$ *in the set* $\mathcal{B}$ *defined in Eq. (12), but they all yield the same KL value. Therefore* $I^\star(\mathbf{p})$ *remains unchanged regardless of the number of runners-up.*
2. *If* $p_{c^\star} = p_{j^\star}$*, then* $I^\star(\mathbf{p}) = 0$.

Remark A.2. *By expanding the Sanov exponent in terms of the probability gap, we recover the error rate obtained via direct application of the central limit theorem. Specifically, for* $\delta = p_{c^\star} - p_{j^\star} \ll p_{j^\star}$,

$$I^\star(\mathbf{p}) = \frac{\delta^2}{4 p_{j^\star}} \left[1 + O(\delta/p_{j^\star})\right].$$

On the other hand, since $\sigma_{j^\star}^2 = 2p_{j^\star} + \delta - \delta^2$, we have $2\sigma_{j^\star}^2 = 4p_{j^\star} + O(\delta)$. Therefore,

$$\frac{\delta^2}{2\sigma_{j^\star}^2} = \frac{\delta^2}{4p_{j^\star}}[1 + O(\delta/p_{j^\star})],$$

which yields

$$I^\star(\mathbf{p}) = \frac{\delta^2}{2\sigma_{j^\star}^2} + O(\delta^3).$$

A.6.2 Bahadur-Rao correction

The random variable $Y_i^j = \mathbf{1}\{X_i = c^\star\} - \mathbf{1}\{X_i = j\} \in \{-1, 0, 1\}$ is integer-valued with span $d = 1$ and has logarithmic moment generating function

$$\Lambda_Y(\lambda) = \log M(\lambda) = \log \mathbb{E}\left[e^{\lambda Y_i^j}\right] = \log\left(p_{c^\star}e^{\lambda} + p_j e^{-\lambda} + 1 - (p_{c^\star} + p_j)\right).$$

Consider $S_j^{(n)} = -\sum_i^n Y_i^j = -Z_j^{(n)}$. We have that

$$\Lambda_{-Y}(\lambda) = \Lambda_Y(-\lambda).$$

Let $\lambda^\star = \frac{1}{2}\log(p_j/p_{c^\star}) < 0$ denote the minimiser of the moment generating function $M(\lambda)$ and define $\eta = -\lambda^\star > 0$. Then,

$$\Lambda'_{-Y}(\eta) = -\Lambda'_Y(-\eta) = -\Lambda'_Y(\lambda^\star) = 0.$$

We are interested in the event $S_j^{(n)} \geq q$, where $q = \Lambda'_{-Y}(\eta) = 0$. By Dembo & Zeitouni (2010, Theorem 3.7.4. (b)), as $n \to \infty$, we obtain the exact asymptotics

$$\mathbb{P}\left[S_j^{(n)} \geq nq\right] = \frac{1 + o(1)}{(1 - \exp(\lambda^\star))\sqrt{2\pi n \Lambda''_{-Y}(-\lambda^\star)}} \exp\left(-n\, \Lambda^\star_{-Y}(q)\right),$$

where $\Lambda^\star_{-Y}(q)$ is the Legendre transform given by

$$\begin{aligned}\Lambda^\star_{-Y}(q) &= \eta q - \Lambda_{-Y}(\eta) = -\Lambda_{-Y}(-\lambda^\star) = -\Lambda_Y(\lambda^\star) \\ &= -\log\left(1 - \left(\sqrt{p_{c^\star}} - \sqrt{p_j}\right)^2\right)\end{aligned}$$

and

$$\Lambda''_{-Y}(-\lambda^\star) = \Lambda_Y(\lambda^\star) = \frac{M''(\lambda^\star)}{M(\lambda^\star)} = \frac{2\sqrt{p_{c^\star}p_j}}{1 - (\sqrt{p_{c^\star}} - \sqrt{p_j})^2} := \tilde{\sigma}_j^2.$$

Finally, we have that as $n \to \infty$,

$$\begin{aligned}\mathbb{P}\left[\hat{c}_n \neq c^\star\right] &\leq \sum_{j \neq c^\star} \mathbb{P}\left[Z_j^{(n)} \leq 0\right] = \sum_{j \neq c^\star} \mathbb{P}\left[S_j^{(n)} \geq 0\right] \\ &= (1 + o(1)) \sum_{j \neq c^\star} \frac{1}{\sqrt{2\pi n}(1 - \exp(1/2\log(p_j/p_{c^\star})))\tilde{\sigma}_j} \exp\left(n\log(1 - (\sqrt{p_{c^\star}} - \sqrt{p_j})^2)\right) \\ &\sim \frac{1}{\sqrt{2\pi n}(1 - \exp(1/2\log(p_{j^\star}/p_{c^\star})))\tilde{\sigma}_{j^\star}} \exp\left(n\log(1 - (\sqrt{p_{c^\star}} - \sqrt{p_{j^\star}})^2)\right) \\ &\sim \frac{1}{\sqrt{2\pi n}(1 - \exp(1/2\log(p_{j^\star}/p_{c^\star})))\tilde{\sigma}_{j^\star}} \exp\left(-nI^\star(\mathbf{p})\right),\end{aligned}$$

where $j^\star = \arg\max_{j \neq c^\star}\ p_j$ is the runner-up.

A.7 Comparison of the different bounds

We perform numerical experiments on synthetic examples to empirically verify the tightness of the derived bounds. We consider a categorical probability distribution with $k = 3$ categories and a small probability gap $\delta = p_{c^\star} - p_{j^\star}$. In particular, we set $p_1 = 0.38$, $p_2 = 0.35$, $p_3 = 0.27$. To compute the empirical estimates of $P[\widehat{c}_n \neq c^\star]$, we employ a Monte Carlo approach with 10^6 samples.

The results are shown in Figure 3. We observe that the CLT bound, with continuity and Berry–Esseen corrections (CLT + CC + BE), provides a very tight estimate that converges to the exact multinomial result as the panel size of voters increases. In contrast, the Hoeffding, Bernstein and Chernoff bounds are noticeably looser. Finally, the Sanov bound with Bahadur-Rao correction (Sanov + BR) is expected to become increasingly tight for larger panels.

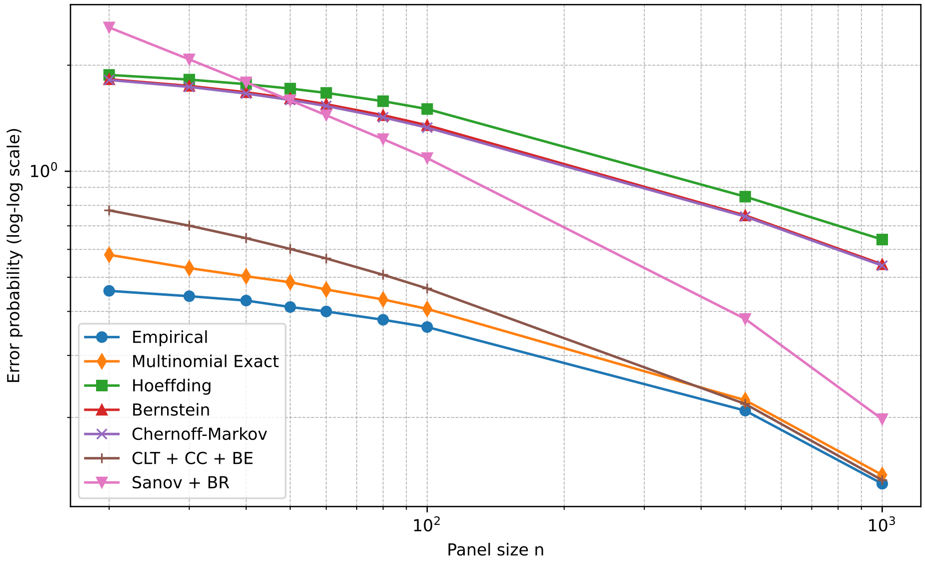


Figure 3: Comparison of empirical and theoretical bounds on $P[\widehat{c}_n \neq c^\star]$ for the probability distribution $\mathbf{p} = (0.38, 0.35, 0.27)$.

B Analysis of the stopping rule

We provide a more detailed analysis of the stopping rule introduced in Section 3.

B.1 Anytime-valid e-processes

Recall the predictable, recursive counts

$$A_{n-1} := \widehat{c}_{n-1}, \quad B_{n-1} := j^\star_{n-1}, \qquad \begin{aligned} s_n &= s_{n-1} + \mathbf{1}\{X_n = A_{n-1}\}, \\ f_n &= f_{n-1} + \mathbf{1}\{X_n = B_{n-1}\}, \\ o_n &= o_{n-1} + \mathbf{1}\{X_n \notin \{A_{n-1}, B_{n-1}\}\}, \end{aligned}$$

and the effective sizes $M_n := s_n + f_n$ (A vs B) and $T_n := s_n + o_n$ (A vs others).

Let $(\pi_n^{\text{run}})_{n\geq 1}$ and $(\pi_n^{\text{oth}})_{n\geq 1}$ be predictable priors (each π_n is $\mathcal{F}_{n-1}$-measurable) supported on $(1/2, 1]$. Define the two mixture e-processes recursively (with optional skipping) by

$$e_n^{\text{run}} = \begin{cases} e_{n-1}^{\text{run}} \cdot 2 \displaystyle\int \theta\, \pi_n^{\text{run}}(d\theta), & X_n = A_{n-1}, \\ e_{n-1}^{\text{run}} \cdot 2 \displaystyle\int (1-\theta)\, \pi_n^{\text{run}}(d\theta), & X_n = B_{n-1}, \\ e_{n-1}^{\text{run}}, & \text{otherwise}, \end{cases}$$

$$e_n^{\text{oth}} = \begin{cases} e_{n-1}^{\text{oth}} \cdot 2 \displaystyle\int \lambda\, \pi_n^{\text{oth}}(d\lambda), & X_n = A_{n-1}, \\ e_{n-1}^{\text{oth}} \cdot 2 \displaystyle\int (1-\lambda)\, \pi_n^{\text{oth}}(d\lambda), & X_n \notin \{A_{n-1}, B_{n-1}\}, \\ e_{n-1}^{\text{oth}}, & \text{if } X_n = B_{n-1}, \end{cases}$$

with $e_0^{\text{run}} = e_0^{\text{oth}} = 1$. By aggregating the per-round factors, we have the equivalent expression (3) and (4).

Before proving Theorem 3.1, we introduce the following auxiliary lemma.

Lemma B.1 (One-step mixture bound)**.** *Let π be any probability measure on $(1/2, 1]$ and define $m := \int u\, \pi(du) \in (1/2, 1]$. If $Y \sim \mathrm{Ber}(\vartheta)$ then*

$$\mathbb{E}\Big[2\int u^Y(1-u)^{1-Y}\,\pi(du)\Big] \;=\; 2\big(1-m+\vartheta(2m-1)\big),$$

which is increasing in ϑ and ≤ 1 for all $\vartheta \le \frac{1}{2}$, with equality at $\vartheta = \frac{1}{2}$.

Proof. $\mathbb{E}[u^Y(1-u)^{1-Y}] = \vartheta u + (1-\vartheta)(1-u) = (1-u) + \vartheta(2u-1)$; integrating over π yields the result. □

Theorem B.2 (Theorem 3.1 restated)**.** *Let $p_j = \mathbb{P}[X = j \mid pr]$. For the* A vs B *test (leader vs runner-up), define $\theta_n = \frac{p_{A_{n-1}}}{p_{A_{n-1}}+p_{B_{n-1}}}$ and the one-sided composite null*

$$H_0^{\mathrm{run}}: \quad \theta_n \le \tfrac{1}{2} \ \big(\textit{equivalently } p_{A_{n-1}} \le p_{B_{n-1}}\big) \textit{ at every round } n.$$

For the A vs others *test, define $\lambda_n = \frac{p_{A_{n-1}}}{p_{A_{n-1}}+\sum_{j\notin\{A_{n-1},B_{n-1}\}} p_j} = \frac{p_{A_{n-1}}}{1-p_{B_{n-1}}}$ and the composite null*

$$H_0^{\mathrm{oth}}: \quad \lambda_n \le \tfrac{1}{2} \ \big(\textit{equivalently } p_{A_{n-1}} \le \textstyle\sum_{j\notin\{A_{n-1},B_{n-1}\}} p_j\big) \textit{ at every round } n.$$

Then $\{e_n^{\mathrm{run}}\}_{n\ge 0}$ and $\{e_n^{\mathrm{oth}}\}_{n\ge 0}$ defined in (3), (4) are non-negative test supermartingales *w.r.t. $\{\mathcal{F}_n\}$, even with predictable, data-dependent priors and optional skipping. Under the boundary (simple) nulls ($\theta_n \equiv \frac{1}{2}$ or $\lambda_n \equiv \frac{1}{2}$ on their informative rounds), they are test* martingales*. Consequently, by Ville's inequality, for any stopping time,*

$$\sup_{\mathbb{P}\in H_0^{\mathrm{run}}} \mathbb{P}\Big(\sup_{n\ge 0} e_n^{\mathrm{run}} \ge 1/\varepsilon\Big) \le \varepsilon, \qquad \sup_{\mathbb{P}\in H_0^{\mathrm{oth}}} \mathbb{P}\Big(\sup_{n\ge 0} e_n^{\mathrm{oth}} \ge 1/\varepsilon\Big) \le \varepsilon.$$

Proof. Fix $n \ge 1$ and condition on $\mathcal{F}_{n-1}$, then A_{n-1}, B_{n-1} and the priors $\pi_n^{\mathrm{run}}, \pi_n^{\mathrm{oth}}$ are deterministic. For the *A vs B* process,

$$\frac{e_n^{\mathrm{run}}}{e_{n-1}^{\mathrm{run}}} = \begin{cases} 2\int \theta'\,\pi_n^{\mathrm{run}}(d\theta'), & X_n = A_{n-1}, \\ 2\int (1-\theta')\,\pi_n^{\mathrm{run}}(d\theta'), & X_n = B_{n-1}, \\ 1, & \text{otherwise.} \end{cases}$$

Write $q_n := p_{A_{n-1}} + p_{B_{n-1}}$ and $\theta_n := p_{A_{n-1}}/q_n$ (if $q_n = 0$ the step is skipped a.s.). Then, under H_0^{run} we have $\theta_n \le \frac{1}{2}$ and

$$\mathbb{E}\Big[\frac{e_n^{\mathrm{run}}}{e_{n-1}^{\mathrm{run}}} \;\Big|\; \mathcal{F}_{n-1}\Big] = q\cdot\mathbb{E}\Big[2\int \theta'^Y(1-\theta')^{1-Y}\,\pi_n^{\mathrm{run}}(d\theta')\Big] \;+\; (1-q)\cdot 1,$$

where $Y \sim \mathrm{Ber}(\theta_n)$ on the informative event. By Lemma B.1, the bracketed expectation is ≤ 1 for $\theta_n \le 1/2$, hence the whole conditional expectation is ≤ 1. Thus $\{e_n^{\mathrm{run}}\}$ is a test supermartingale (and a martingale at $\theta_n = 1/2$).

Similarly, for the *A vs others* process,

$$\frac{e_n^{\mathrm{oth}}}{e_{n-1}^{\mathrm{oth}}} = \begin{cases} 2\int \lambda'\,\pi_n^{\mathrm{oth}}(d\lambda'), & X_n = A_{n-1}, \\ 2\int (1-\lambda')\,\pi_n^{\mathrm{oth}}(d\lambda'), & X_n \notin \{A_{n-1}, B_{n-1}\}, \\ 1, & \text{if } X_n = B_{n-1}. \end{cases}$$

Let $r_n := 1 - p_{B_{n-1}}$ and $\lambda_n := p_{A_{n-1}}/r_n$. Under H_0^{oth}, $\lambda_n \le 1/2$ and the same calculation gives

$$\mathbb{E}\Big[\frac{e_n^{\mathrm{oth}}}{e_{n-1}^{\mathrm{oth}}} \;\Big|\; \mathcal{F}_{n-1}\Big] = r\cdot\mathbb{E}\Big[2\int \lambda'^Z(1-\lambda')^{1-Z}\,\pi_n^{\mathrm{oth}}(d\lambda')\Big] \;+\; (1-r)\cdot 1 \;\le\; 1,$$

where $Z \sim \mathrm{Ber}(\lambda_n)$ on the informative event. Ville's inequality yields the stated time-uniform bounds. □

B.2 Estimation of $\widehat{\varepsilon} \geq \mathbb{P}[\hat{c}_n \neq c^\star]$

For convenience, we describe below how to compute $1 - \hat{\varepsilon}$, which provides a lower bound $1 - \hat{\varepsilon} \leq \mathbb{P}[\widehat{c}_n = c^\star]$. Before doing so, recall that if a and b denote two possible outcomes of a multinomial distribution, then

$$\mathbb{P}[p_a > p_b] = \mathbb{P}\left[\theta_{ab} = \tfrac{p_b}{p_a+p_b} < \tfrac{1}{2}\right].$$

This probability can be estimated using a Beta approximation. Assuming a Beta prior on θ_{ab} with parameters $(1,1)$, and letting N_a and N_b denote the observed counts for each outcome, we obtain

$$\mathbb{P}[\theta_{ab} < \tfrac{1}{2}] = \frac{\Gamma(N_a, N_b)}{\Gamma(N_a)\Gamma(N_b)} \int_0^{1/2} \theta^{N_a - 1}(1-\theta)^{N_b - 1}\, d\theta := I_{1/2}(N_a, N_b).$$

Therefore, we have

$$\begin{aligned} \mathbb{P}[\hat{c}_n = c^\star] &\gtrsim \min\left(\mathbb{P}(p_{\hat{c}_n} > p_{j_n^\star}), \mathbb{P}(p_{\hat{c}_n} > p_{\hat{o}_n})\right) \\ &\approx \min\left(I_{1/2}(f_n + 1, s_n + 1), I_{1/2}(o_n + 1, s_n + 1)\right). \end{aligned} \tag{13}$$

B.3 Stopping time

When the prior is of the form

$$\Pi_n(d\boldsymbol{\theta}) = \prod_{i=1}^{n} \delta_{\theta^\star}(d\theta_i)$$

the corresponding e-process is given by

$$e_n = 2^M (\theta^\star)^s (1-\theta^\star)^f.$$

If the data-generating process follows a Bernoulli distribution with parameter $\theta^\star$, then $s \approx M\theta^\star$, yielding

$$\begin{aligned} \log e_n &= M\left(\frac{s}{M}\log(2\theta^\star) + \left(1 - \frac{s}{M}\right)\log 2(1-\theta^\star)\right) \\ &\approx M\left(\theta^\star \log(2\theta^\star) + (1-\theta^\star)\log 2(1-\theta^\star)\right) \\ &= M D_{\mathrm{KL}}(\text{Ber}(\theta^\star)\|\text{Ber}(1/2)). \end{aligned}$$

Therefore, the number of informative rounds required until stopping is

$$\begin{aligned} M_\tau &= \inf\{M:\ \log e_n \geq \log(1/\varepsilon)\} \\ &= \inf\left\{M:\ M D_{\mathrm{KL}}(\text{Ber}(\theta^\star)\|\text{Ber}(1/2)) \geq \log(1/\varepsilon)\right\} \\ &\approx \frac{\log(1/\varepsilon)}{D_{\mathrm{KL}}(\text{Ber}(\theta^\star)\|\text{Ber}(1/2))}. \end{aligned}$$

Note that when $\theta^\star$ is close to $1/2$, we can approximate $D_{\mathrm{KL}}(\text{Ber}(1/2+\varepsilon)\|\text{Ber}(1/2)) \approx 2\varepsilon^2$, which leads to

$$M_{\text{lead, runner-up}} \approx \tfrac{2\left(p_{\hat{c}}+p_{j^\star}\right)^2}{\left(p_{\hat{c}}-p_{j^\star}\right)^2}\log(1/\varepsilon), \qquad M_{\text{lead,others}} \approx \tfrac{2\left(1-p_{j^\star}\right)^2}{\left(2p_{\hat{c}}+p_{j^\star}-1\right)^2}\log(1/\varepsilon).$$

Finally, since the expected number of rounds until an informative one occurs is

$$K_{\text{lead, runner-up}} = \tfrac{1}{p_{\hat{c}}+p_{j^\star}}, \qquad K_{\text{lead,others}} = \tfrac{1}{1-p_{j^\star}}.$$

due to the properties of the geometric distribution, we find that the total number of rounds required is approximately $N = K \cdot M$

$$N_{\text{lead, runner-up}} \approx \tfrac{2\left(p_{\hat{c}}+p_{j^\star}\right)}{\left(p_{\hat{c}}-p_{j^\star}\right)^2}\log(1/\varepsilon), \qquad N_{\text{lead,others}} \approx \tfrac{2\left(1-p_{j^\star}\right)}{\left(2p_{\hat{c}}+p_{j^\star}-1\right)^2}\log(1/\varepsilon).$$

Moreover, when $p_{\hat{c}} - p_{j^\star} \ll p_{j^\star}$, the ratio $\scriptstyle (p_{\hat{c}}+p_{j^\star})/(p_{\hat{c}}-p_{j^\star})^2$ is approximately $\text{SNR}(\Delta_{j^\star})^{-1}$, where $\Delta_{j^\star} = \scriptstyle \mathbf{1}\{X=\hat{c}\} - \mathbf{1}\{X=j^\star\}$.

B.4 Algorithms for truncated $\mathrm{Beta}(a,b)$ and updating point prior

We provide pseudocode for implementing the MMC stopping rule with the truncated $\mathrm{Beta}(a,b)$ shared-parameter prior (Algorithm 2) and the shared-parameter point prior presented in B.1 (Algorithm 3).

Algorithm 2 MMC stopping rule with truncated $\mathrm{Beta}(a, b)$ prior

```
Require: confidence level $\varepsilon$, budget $N_{\text{budget}}$, hyperparameters $a, b > 0$; deterministic tie-break rule
Init: $n \leftarrow 0$; for all $j \in \{1, \dots, k\}$ set label counts $N_j \leftarrow 0$; $s_0 = f_0 = o_0 \leftarrow 0$; $e_0^{\text{run}} = e_0^{\text{oth}} \leftarrow 1$
Define $\mathsf{B}_{>1/2}(a, b) = \int_{1/2}^{1} t^{a-1}(1-t)^{b-1}\,dt$
while True do
    Predictable top-2: set $A_n \leftarrow \arg\max_j N_j$, $B_n \leftarrow$ second largest (ties broken deterministically)
    Cache counts (pre-update): $\tilde{s} \leftarrow s_n$, $\tilde{f} \leftarrow f_n$, $\tilde{o} \leftarrow o_n$
    Draw a new vote: sample $X \sim \mathbb{P}[\,\cdot\,|pr]$
    Per-round ratio (A vs B):
        $$\rho_{\text{run}} = \begin{cases} 2\,\dfrac{\mathsf{B}_{>1/2}(a+\tilde{s}+1,\ b+\tilde{f})}{\mathsf{B}_{>1/2}(a+\tilde{s},\ b+\tilde{f})}, & X = A_n, \\ 2\,\dfrac{\mathsf{B}_{>1/2}(a+\tilde{s},\ b+\tilde{f}+1)}{\mathsf{B}_{>1/2}(a+\tilde{s},\ b+\tilde{f})}, & X = B_n, \\ 1, & \text{otherwise.} \end{cases}$$
    Per-round ratio (A vs others):
        $$\rho_{\text{oth}} = \begin{cases} 2\,\dfrac{\mathsf{B}_{>1/2}(a+\tilde{s}+1,\ b+\tilde{o})}{\mathsf{B}_{>1/2}(a+\tilde{s},\ b+\tilde{o})}, & X = A_n, \\ 2\,\dfrac{\mathsf{B}_{>1/2}(a+\tilde{s},\ b+\tilde{o}+1)}{\mathsf{B}_{>1/2}(a+\tilde{s},\ b+\tilde{o})}, & X \notin \{A_n, B_n\}, \\ 1, & X = B_n. \end{cases}$$
    Update $e$-values: $e_{n+1}^{\text{run}} \leftarrow e_n^{\text{run}} \cdot \rho_{\text{run}}$,  $e_{n+1}^{\text{oth}} \leftarrow e_n^{\text{oth}} \cdot \rho_{\text{oth}}$
    Update recursive counts:
        $$(s_{n+1}, f_{n+1}, o_{n+1}) = \begin{cases} (\tilde{s}+1, \tilde{f}, \tilde{o}), & X = A_n, \\ (\tilde{s}, \tilde{f}+1, \tilde{o}), & X = B_n, \\ (\tilde{s}, \tilde{f}, \tilde{o}+1), & \text{otherwise.} \end{cases}$$
    Update label counts: $N_X \leftarrow N_X + 1$; $n \leftarrow n + 1$
    Check stop: if $e_n^{\text{run}} \geq 1/\varepsilon$ and $e_n^{\text{oth}} \geq 1/\varepsilon$ then
        set $\hat{c} \leftarrow \arg\max_j N_j$; return $(\hat{c},\ \text{stopped})$
    Budget: if $n \geq N_{\text{budget}}$ then return $(\arg\max_j N_j,\ \text{abstained})$
```

B.5 General stopping rule

The proposed stopping rule exploits the fact that although the space of possible LLM outputs may be large, the true distribution $\mathbb{P}[\cdot|pr]$ (for a given prompt pr) is typically concentrated on a subset of m classes, with $m \ll k$, where $\{1, \dots, k\}$ is the total support. We further assume that the conditional distribution over these top-m classes is approximately uniform. Based on this, we design a strategy that performs pairwise comparisons: between the leader and each of the top-$(m-1)$ runner-ups, and between the leader and the remaining classes.

Similarly to the case $m = 2$, at round $n \geq 1$, *before* observing X_n, set the predictable top-m labels as

$$A_{n-1} := \widehat{c}_{n-1}, \qquad B_{n-1}^{i} := j^{\star}_{n-1,i}, \qquad B_{n-1} := \{B_{n-1}^{1}, \dots, B_{n-1}^{m-1}\},$$

which are measurable w.r.t. $\mathcal{F}_{n-1} = \sigma(X_1, \dots, X_{n-1})$ (ties broken deterministically). We maintain the following *recursive, predictable* counts

Leader hits:	$s_n = s_{n-1} + \mathbf{1}\{X_n = A_{n-1}\}, \quad s_0 = 0,$
i-th runner-up hits (for the A vs B$^{\mathbf{i}}$ test):	$f_n^{i} = f_{n-1}^{i} + \mathbf{1}\{X_n = B_{n-1}^{i}\}, \quad f_0^{i} = 0,$
Others hits (for the A vs others test):	$o_n = o_{n-1} + \mathbf{1}\{X_n \notin \{A_{n-1}, B_{n-1}\}\}, \quad o_0 = 0.$

Thus the sample sizes are

$$M_n^{i} := s_n + f_n^{i}, \qquad T_n := s_n + o_n.$$

Let $(\pi_n^{\text{run},i})_{n \geq 1}$, $i = 1, \dots, m-1$, and $(\pi_n^{\text{oth}})_{n \geq 1}$ be predictable priors (i.e., $\mathcal{F}_{n-1}$-measurable) supported on $(1/2, 1]$.

Algorithm 3 MMC stopping rule with updating point prior

```
Require: Confidence level $\varepsilon$; budget $N_{\text{budget}}$; Dirichlet smoothing $(\alpha_A, \alpha_B, \alpha_O) > 0$; clipping $\varepsilon \in (0, 10^{-3}]$; deterministic tie-break rule
Init: $n \leftarrow 0$; for all $j \in \{1, \ldots, k\}$ set label counts $N_j \leftarrow 0$; $s_0 = f_0 = o_0 \leftarrow 0$; $e_0^{\text{run}} = e_0^{\text{oth}} \leftarrow 1$
while True do
    Predictable top–2: set $A_n \leftarrow \arg\max_j N_j,\ B_n \leftarrow$ second largest (break ties deterministically)
    Predictable total counts: $L \leftarrow s_n + f_n + o_n$
    Shared multinomial plug–in (Dirichlet–smoothed):
        $$\hat{p}_A \leftarrow \frac{s_n + \alpha_A}{L + \alpha_A + \alpha_B + \alpha_O}, \quad \hat{p}_B \leftarrow \frac{f_n + \alpha_B}{L + \alpha_A + \alpha_B + \alpha_O}$$
        $$\theta_n^\star \leftarrow \text{clip}\Big(\frac{\hat{p}_A}{\hat{p}_A + \hat{p}_B},\ \tfrac{1}{2} + \varepsilon,\ 1 - \varepsilon\Big), \quad \lambda_n^\star \leftarrow \text{clip}\Big(\frac{\hat{p}_A}{1 - \hat{p}_B},\ \tfrac{1}{2} + \varepsilon,\ 1 - \varepsilon\Big)$$
    Draw a new vote: sample $X \sim \mathbb{P}[\cdot\,|pr]$
    Update recursive counts:
        $$(s_{n+1}, f_{n+1}, o_{n+1}) = \begin{cases} (s_n{+}1, f_n, o_n), & X = A_n, \\ (s_n, f_n{+}1, o_n), & X = B_n, \\ (s_n, f_n, o_n{+}1), & \text{otherwise} \end{cases}$$
    Update e–values:
        $$e_n^{\text{run}} = 2^{s_{n+1} + f_{n+1}} (\theta_n^\star)^{s_{n+1}} (1 - \theta_n^\star)^{f_{n+1}}, \qquad e_n^{\text{oth}} = 2^{s_{n+1} + o_{n+1}} (\theta_n^\star)^{s_{n+1}} (1 - \theta_n^\star)^{o_{n+1}}.$$
    Update label counts: $N_X \leftarrow N_X + 1$;   $n \leftarrow n + 1$
    Check stop: if $e_n^{\text{run}} \geq 1/\varepsilon$ and $e_n^{\text{oth}} \geq 1/\varepsilon$ then
        set $\hat{c} \leftarrow \arg\max_j N_j$; return $(\hat{c},\ \text{stopped})$
    Budget: if $n \geq N_{\text{budget}}$ then return $(\arg\max_j N_j,\ \text{abstained})$
```

We define the m mixture e-processes recursively (with optional skipping) by

$$e_n^{\text{run},i} = \begin{cases} e_{n-1}^{\text{run},i} \cdot 2 \displaystyle\int \theta \, \pi_n^{\text{run},i}(d\theta), & X_n = A_{n-1}, \\ e_{n-1}^{\text{run},i} \cdot 2 \displaystyle\int (1 - \theta) \, \pi_n^{\text{run},i}(d\theta), & X_n = B_{n-1}^i, \\ e_{n-1}^{\text{run},i}, & \text{otherwise,} \end{cases}$$

$$e_n^{\text{oth}} = \begin{cases} e_{n-1}^{\text{oth}} \cdot 2 \displaystyle\int \lambda \, \pi_n^{\text{oth}}(d\lambda), & X_n = A_{n-1}, \\ e_{n-1}^{\text{oth}} \cdot 2 \displaystyle\int (1 - \lambda) \, \pi_n^{\text{oth}}(d\lambda), & X_n \notin \{A_{n-1}, B_{n-1}\}, \\ e_{n-1}^{\text{oth}}, & \text{if } X_n = B_{n-1}, \end{cases}$$

with $e_0^{\text{run},i} = e_0^{\text{oth}} = 1$. Thanks to Theorem 3.1 $\{e_n^{\text{run},i}\}$, $i = 1, \ldots, m-1$, and $\{e_n^{\text{oth}}\}$ are non-negative test supermartingales under their respective composite nulls, and test martingales under the boundary nulls.

B.6 Analysis of the stopping rule on synthetic data

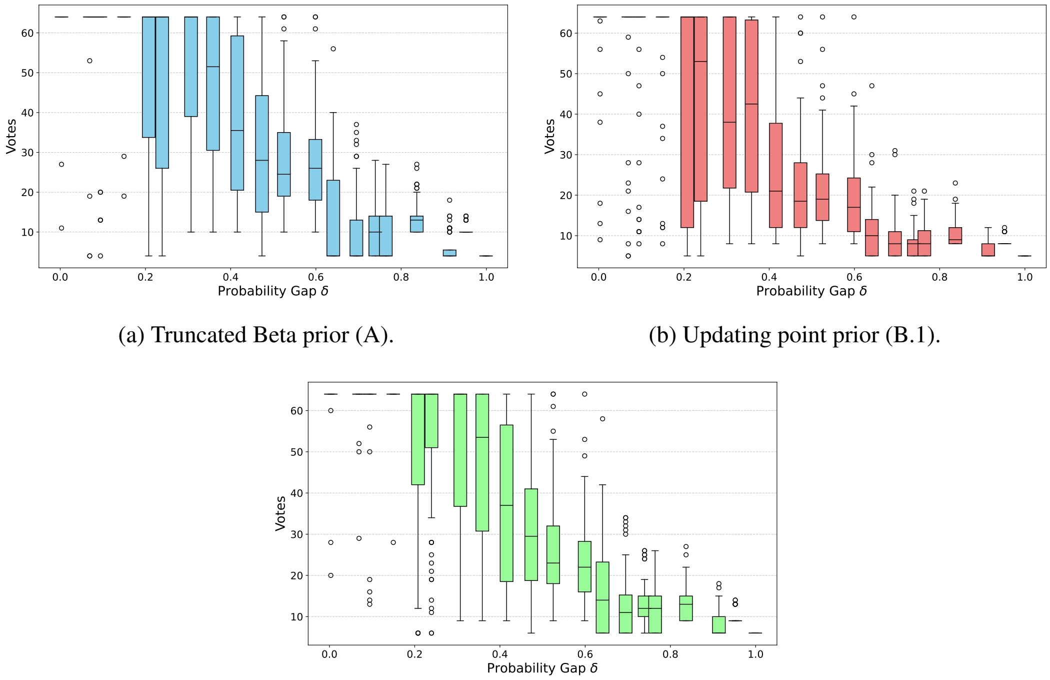
We analyse the performance of the proposed MMC stopping rule on synthetic data, focusing on the impact of the prior distribution choice. To do so, we simulate different probability distributions over $k = 26$ classes and evaluate the performance as a function of the probability gap $\delta = p_{c^\star} - p_{j^\star}$, where $c^\star$ and $j^\star$ denote the true majority vote and the runner-up, respectively.

Following Algorithm 1, we set the algorithm parameters to $\varepsilon = 0.1$ (confidence level) and $N_{\text{budget}} = 64$ (maximum budget). This ensures that, at the final iteration, either the budget is reached or the following guarantee holds

$$\mathbb{P}\left[\widehat{c}_n \neq c^\star\right] \leq \varepsilon.$$

Figure 4 presents boxplots of the number of votes required to stop under the MMC rule as a function of the probability gap δ. For small values of δ, the number of votes saturates at the maximum budget.

As δ increases, the average number of votes required to guarantee the correctness of the majority vote decreases. Comparing the three prior choices for the same value of δ, we observe that using an updating point prior with shared parameter, as presented in B.1 (Fig. 4b) results in fewer votes to achieve statistical guarantees than either a truncated Beta prior with shared parameter (Fig. 4a) or an updating point prior based on ratio updates, presented in B.2 (Fig. 4c).



(a) Truncated Beta prior (A).

(b) Updating point prior (B.1).

(c) Updating point prior (B.2).

Figure 4: Boxplots showing the distribution of the number of votes required until stopping under the MMC rule as a function of the probability gap $\delta = p_{c^\star} - p_{j^\star}$. Results are shown for $\varepsilon = 0.1$ with a maximum budget of 64 votes.

C Test-time training objectives

C.1 Test-time reinforcement learning (TTRL)

TTRL leverages majority voting over n responses $X_1, \dots, X_n$ as a proxy for the correct answer, and defines the reward function $r_n(Y_i) = \mathbf{1}\{X_i = \widehat{c}_n\}$, where $\widehat{c}_n$ is the majority vote. The regularised objective it minimises is of the form

$$L(\pi) := -\mathbb{E}_{pr\sim Q}\,\mathbb{E}_{Y\sim\pi(\cdot|pr)}[\mathbf{1}\{X = \widehat{c}_n\}] + \beta\, D_{\mathrm{KL}}(\pi(\cdot|pr)||\pi_{\mathrm{ref}}(\cdot|pr)),$$

where π is the candidate distribution and π_{ref} is a pre-trained reference model. Note that L is strictly convex and therefore admits a unique global minimiser.

Optimisation of the regularised objective. To compute the optimiser, we introduce a Lagrange multiplier λ to enforce normalisation and consider a perturbation $\pi_\varepsilon = \pi + \varepsilon\varphi$ with $\int \varphi\, d\mu = 0$. The directional derivative at $\varepsilon = 0$ is

$$\frac{d}{d\varepsilon}\left[L[\pi_\varepsilon] + \lambda \int \pi_\varepsilon\, d\mu\right]\Bigg|_{\varepsilon=0} = \int_\Omega \left[-\delta_{\widehat{c}_n} + \beta\left(1 + \log\frac{\pi}{\pi_{\text{ref}}}\right) + \lambda\right]\varphi\, d\mu.$$

Since this must vanish for all admissible φ, we obtain the pointwise stationarity condition

$$-\mathbf{1}\{x = \widehat{c}_n\} + \beta\left(1 + \log\frac{\pi(x)}{\pi_{\text{ref}}(x)}\right) + \lambda = 0.$$

Solving this yields the tilted distribution

$$\pi^\star(y|pr) \propto e^{\mathbf{1}\{x=\widehat{c}_n\}/\beta}\pi_{\text{ref}}(y|pr) = \left(1 + \mathbf{1}\{x = \widehat{c}_n\}\left(e^{\frac{1}{\beta}} - 1\right)\right)\pi_{\text{ref}}(y|pr).$$

As $\beta \to 0$, the model converges to a Dirac delta centred at $\widehat{c}_n$. For non-zero regularisation values β, the solution retains some structure from the reference model. Assuming π_{ref} is normalised and $e^{1/\beta} > 1$, we can write

$$\pi^\star(y|pr) = \frac{e^{\mathbf{1}\{x=\widehat{c}_n\}/\beta}\pi_{\text{ref}}(y|pr)}{\pi_{\text{ref}}(\widehat{c}_n|pr)e^{1/\beta} + \sum_{x' \neq \widehat{c}_n}\pi_{\text{ref}}(y'|pr)} = \frac{e^{\mathbf{1}\{x=\widehat{c}_n\}/\beta}\pi_{\text{ref}}(y|pr)}{1 + \pi_{\text{ref}}(\widehat{c}_n|pr)(e^{1/\beta} - 1)}.$$

Let $\kappa = 1/\beta$ and $p_j = \pi_{\text{ref}}(j)$. We now analyse the behaviour of $\text{SNR}(\Delta_{j^\star})$ as a function of κ. To do so, we compute its derivative with respect to κ

$$\begin{aligned}
\frac{d}{d\kappa}\text{SNR}_{\Delta_{j^\star}}(\kappa) =& \frac{d}{d\kappa}\frac{(\pi^\star_{\hat{c}} - \pi^\star_{j^\star})^2}{(\pi^\star_{\hat{c}} + \pi^\star_{j^\star}) - (\pi^\star_{\hat{c}} - \pi^\star_{j^\star})^2} \\
=& \frac{d}{d\kappa}\frac{(p_{\hat{c}}e^\kappa - p_{j^\star})^2}{(p_{\hat{c}}e^\kappa + p_{j^\star})(1 + (e^\kappa - 1)p_{\hat{c}}) - (p_{\hat{c}}e^\kappa - p_{j^\star})^2} \\
=& \frac{2(p_{\hat{c}}e^\kappa - p_{j^\star})p_{\hat{c}}e^\kappa\left[(p_{\hat{c}}e^\kappa + p_{j^\star})(1 + (e^\kappa - 1)p_{\hat{c}}) - (p_{\hat{c}}e^\kappa - p_{j^\star})^2\right]}{((p_{\hat{c}}e^\kappa + p_{j^\star})(1 + (e^\kappa - 1)p_{\hat{c}}) - (p_{\hat{c}}e^\kappa - p_{j^\star})^2)^2} \\
&- \frac{(p_{\hat{c}}e^\kappa - p_{j^\star})^2 p_{\hat{c}}e^\kappa\left[(1 + (e^\kappa - 1)p_{\hat{c}}) + (p_{\hat{c}}e^\kappa + p_{j^\star})\right]}{\left((p_{\hat{c}}e^\kappa + p_{j^\star})(1 + (e^\kappa - 1)p_{\hat{c}}) - (p_{\hat{c}}e^\kappa - p_{j^\star})^2\right)^2} \\
&+ \frac{2(p_{\hat{c}}e^\kappa - p_{j^\star})p_{\hat{c}}e^\kappa(p_{\hat{c}}e^\kappa - p_{j^\star})^2}{\left((p_{\hat{c}}e^\kappa + p_{j^\star})(1 + (e^\kappa - 1)p_{\hat{c}}) - (p_{\hat{c}}e^\kappa - p_{j^\star})^2\right)^2} \\
=& \frac{(\square)}{((p_{\hat{c}}e^\kappa + p_{j^\star})(1 + (e^\kappa - 1)p_{\hat{c}}) - (p_{\hat{c}}e^\kappa - p_{j^\star})^2)^2}.
\end{aligned}$$

The denominator is clearly positive, so we focus on the numerator. After cancelling out common terms, the numerator reduces to

$$\begin{aligned}
(\square) =& (p_{\hat{c}}e^\kappa - p_{j^\star})p_{\hat{c}}e^\kappa\Big[2(p_{\hat{c}}e^\kappa + p_{j^\star})(1 + (e^\kappa - 1)p_{\hat{c}}) - (1 + (e^\kappa - 1)p_{\hat{c}})(p_{\hat{c}}e^\kappa - p_{j^\star}) \\
&- (p_{\hat{c}}e^\kappa + p_{j^\star})(p_{\hat{c}}e^\kappa - p_{j^\star})\Big] \\
=& (p_{\hat{c}}e^\kappa - p_{j^\star})p_{\hat{c}}e^\kappa\Big[2p_{j^\star}(1 + (e^\kappa - 1)p_{\hat{c}}) + (p_{\hat{c}}e^\kappa + p_{j^\star})(1 - p_{\hat{c}} + p_{j^\star})\Big].
\end{aligned}$$

Since $\kappa \geq 0$ and $0 \leq p_{j^\star} \leq p_{\hat{c}} \leq 1$, it follows that $(\square) \geq 0$, with equality if and only if $p_{\hat{c}} = 1$. Therefore, for $0 < p_{\hat{c}} < 1$, $\frac{d}{d\kappa}\text{SNR}_{\Delta_{j^\star}}(\kappa) > 0$, which implies that $\text{SNR}_{\Delta_{j^\star}}(\kappa)$ is an increasing function of κ. This demonstrates that optimising the TTRL objective reduces the number of samples required to achieve statistical certificates.

C.2 SNR-BASED TEST-TIME RL OBJECTIVE

Let $\mathbf{X} = (X_1, \ldots, X_n)$ be a collection of answers to a given prompt corresponding to rollouts $\mathbf{Y} = (Y_1, \ldots, Y_n)$, with $\widehat{c}_n$ denoting the majority vote and $j_n^\star$ the runner-up. We propose to directly maximise $\text{SNR}(\Delta_{j_n^\star})$ by using the group-level reward function $r_n^{(1)}$ defined in Eq. (6). Our objective (without the KL-regularisation) takes the form

$$\begin{aligned}
\max_\phi \mathbb{E}_{Y_1,\ldots,Y_n \sim \pi_\phi(\cdot|pr)}\left[r_n^{(1)}(\mathbf{Y})\right] &= \max_\phi \mathbb{E}_{Y_1,\ldots,Y_n \sim \pi_\phi(\cdot|pr)}\left[\widehat{\text{SNR}}(\Delta_{j_n^\star})(\mathbf{X})\right] \\
&= \max_\phi \mathbb{E}_{Y_1,\ldots,Y_n \sim \pi_\phi(\cdot|pr)}\left[\frac{(N_{\hat{c}_n} + N_{j_n^\star})^2}{n(N_{\hat{c}_n} - N_{j_n^\star}) - (N_{\hat{c}_n} - N_{j_n^\star})^2}\right],
\end{aligned}$$

where $N_j = \sum_i \mathbf{1}\{X_i = j\}$. It is important to note that $\widehat{\mathrm{SNR}}(\Delta_{j_n^\star})(\mathbf{X})$ is a biased estimator of $\mathrm{SNR}(\Delta_{j_n^\star})$, however in the large-sample limit we obtain the approximation

$$\begin{aligned}
&\max_\phi \mathbb{E}_{Y_1,\ldots,Y_n\sim\pi_\phi(\cdot|pr)}\left[r_n^{(1)}(\mathbf{Y})\right] = \max_\phi \mathbb{E}_{Y_1,\ldots,Y_n\sim\pi_\phi(\cdot|pr)}\left[\widehat{\mathrm{SNR}}(\Delta_{j_n^\star})(\mathbf{X})\right] \\
&\approx \max_\phi \frac{(q_{\hat{c}_n} - q_{j_n^\star})^2}{q_{\hat{c}_n} + q_{j_n^\star} - (q_{\hat{c}_n} - q_{j_n^\star})^2} = \max_\phi \mathrm{SNR}(\Delta_{j_n^\star}).
\end{aligned}$$

As discussed in the main text, to reduce the variance of the gradient estimate of the group-level reward, we adopt a leave one-out control variate approach (Tang et al., 2025), resulting in the following effective advantage function for Y_i when using the REINFORCE algorithm (Williams, 1992)

$$A_i = \widehat{\mathrm{SNR}}(\Delta_{j_n^\star})(\mathbf{X}) - \widehat{\mathrm{SNR}}(\Delta_{j_n^\star})(\mathbf{X}_{-i}). \tag{14}$$

Under the GRPO algorithm (Shao et al., 2024), the effective advantage for Y_i becomes

$$\hat{A}_i = \widehat{\mathrm{SNR}}(\Delta_{j_n^\star})(\mathbf{X}) - \widehat{\mathrm{SNR}}(\Delta_{j_n^\star})(\mathbf{X}_{-i}) - \frac{1}{n}\sum_i A_i, \tag{15}$$

which further reduces the variance of the gradient estimate at the expense of introducing some bias (Tang et al., 2025). In addition, we regularise the objective with a KL term that penalises deviations from a reference model π_{ref}.

Optimisation of the regularised objective. Let $\pi_{\text{ref}} = (p_1, \ldots, p_k)$. We optimise over categorical distributions $\pi = (q_1, \ldots, q_k)$. To enforce the normalisation constraint $\sum_i q_i = 1$, we introduce a Lagrange multiplier λ, yielding the Lagrangian

$$L(\pi, \lambda) = -\frac{(q_{\hat{c}_n} - q_{j_n^\star})^2}{q_{\hat{c}_n} + q_{j_n^\star} - (q_{\hat{c}_n} - q_{j_n^\star})^2} + \beta \sum_i q_i \log \frac{q_i}{p_i} + \lambda\left(\sum_i q_i - 1\right),$$

under the large-sample limit approximation described above.

The stationary points satisfy

$$\frac{\partial L}{\partial q_i} = 0, \quad \forall i.$$

Define

$$g(x, y) = -\frac{(x-y)^2}{x + y - (x-y)^2}$$

and denote by g_x, g_y its partial derivatives with respect to x and y, respectively. The optimality conditions are given by

$$\begin{aligned}
g_x(q_{\hat{c}_n}, q_{j_n^\star}) + \beta\left(1 + \log\frac{q_{\hat{c}_n}}{p_{\hat{c}_n}}\right) + \lambda = 0, \\
g_y(q_{\hat{c}_n}, q_{j_n^\star}) + \beta\left(1 + \log\frac{q_{j_n^\star}}{p_{j_n^\star}}\right) + \lambda = 0, \\
\beta\left(1 + \log\frac{q_i}{p_i}\right) + \lambda = 0 \Longrightarrow q_i \propto p_i, \quad i \neq \hat{c}_n, j_n^\star.
\end{aligned}$$

In general, these equations do not admit a closed-form solution and must be solved numerically.

C.3 Entropy-based test-time RL objective

Let $\mathbf{X} = (X_1, \ldots, X_n)$ denote the set of i.i.d. answers to a given prompt corresponding to rollouts $\mathbf{Y} = (Y_1, \ldots, Y_n)$. Define $N_j = \sum_i \mathbf{1}\{X_i = j\}$. In the main text, we proposed a group-level reward function based on the plug-in estimator of the negative entropy

$$r_n^{(2)}(\mathbf{Y}) = \sum_{j:\, N_j>0} \frac{N_j}{n} \log\left(\frac{N_j}{n}\right).$$

This estimator is known to overestimate $\mathbb{E}[\log X]$, with an error of approximately $(k-1)/(2n)$, where k is the total number of classes of the distribution. An alternative approach is to introduce a Dirichlet prior on the class probabilities, $(p_1, \dots, p_k) \sim \text{Dir}(k, \alpha, \dots, \alpha)$. Since the data are multinomial, the posterior distribution of the probabilities is also Dirichlet. After n observations we obtain

$$(p_1, \dots, p_k)|\mathbf{Y}, \alpha \sim \text{Dir}(k, \alpha + N_1, \dots, \alpha + N_k)$$

This leads to the alternative estimator

$$\hat{r}_n^{(2)}(\mathbf{Y}) = \sum_j \frac{N_j + \alpha}{n + \alpha} \log\left(\frac{N_j + \alpha}{n + \alpha}\right).$$

Because our ensembles of voters are typically small, this Bayesian smoothing can help mitigate fluctuations, especially when prior information is available.

By using the reward functions $r_n^{(2)}(\mathbf{Y})$ or $\hat{r}_n^{(2)}(\mathbf{Y})$, the goal is to minimise the entropy of the answer distribution. In particular, our objective (without regularisation) is

$$\max_\phi \mathbb{E}_{Y_1, \dots, Y_n \sim \pi_\phi(\cdot|pr)} \left[r_n^{(2)}(\mathbf{Y})\right] = \max_\phi \mathbb{E}_{Y_1, \dots, Y_n \sim \pi_\phi(\cdot|pr)} \left[\sum_{j:\ N_j > 0} \frac{N_j}{n} \log\left(\frac{N_j}{n}\right)\right].$$

As in the previous section, $r_n^{(2)}(\mathbf{Y})$ is a biased estimator of the negative entropy $\mathbb{E}[\log X]$. However, in the large-sample limit we obtain the approximation

$$\max_\phi \mathbb{E}_{Y_1, \dots, Y_n \sim \pi_\phi(\cdot|pr)} \left[r_n^{(2)}(\mathbf{Y})\right] \approx \max_\phi \sum_{j:\ p_{j,\phi} > 0} p_{j,\phi} \log p_{j,\phi} = \max_\phi \mathbb{E}_{Y \sim \pi_\phi(\cdot|pr)}[\log X].$$

To reduce the variance of the gradient estimates of the group-level reward, we also employ the effective advantage functions introduced in (14) and (15), for the REINFORCE and GRPO algorithms, respectively.

Optimisation of the regularised objective. Let $\pi_{\text{ref}}(Y_{0:\tau})$ denote the reference distribution over reasoning trajectories with terminal variable $X = g(Y_{\tau:})$, and write $p_{\text{ref}}(x) = \pi_{\text{ref}}(X = x)$ for its induced marginal. As mentioned in the main text, the KL-regularised variational problem over the base measure reduces to one over the marginal $q(x) = \pi_\phi(x)$ alone, with the following loss

$$\begin{aligned}
L(q) &= H(q) + \beta\, D_{\text{KL}}(q||p_{\text{ref}}) \\
&= \beta\,(1/\beta\; H(q) + \beta\, D_{\text{KL}}(q||p_{\text{ref}})) \\
&\propto \mathbb{E}_{pr \sim Q}\, \mathbb{E}_{X \sim q(\cdot|pr)}\, [-1/\beta \log q(X|pr)] + D_{\text{KL}}(q(\cdot|pr)||p_{\text{ref}}(\cdot|pr)) \\
&= \mathbb{E}_{pr \sim Q}\, \mathbb{E}_{X \sim q(\cdot|pr)}\, [(1 - 1/\beta) \log q(X|pr) - \log p_{\text{ref}}(X|pr))],
\end{aligned}$$

where $\beta > 1$. Since the mapping $q \mapsto (1 - 1/\beta) \int q \log q$ is strictly convex, and the second term is linear, it follows that L is strictly convex on the space of probability distributions. Consequently, any stationary point is necessarily the unique global minimiser.

As in Section C.1, to compute the optimiser we introduce a Lagrange multiplier λ to enforce normalisation and consider a perturbation $q_\varepsilon = q + \varepsilon\varphi$ with $\int \varphi\, d\mu = 0$. The directional derivative at $\varepsilon = 0$ is

$$\frac{d}{d\varepsilon}\left[L[q_\varepsilon] + \lambda \int q_\varepsilon\, d\mu\right]\Bigg|_{\varepsilon=0} = \int_\Omega [(1 - 1/\beta)(1 + \log q) - \log p_{\text{ref}} + \lambda]\ \varphi\, d\mu.$$

Since this must vanish for all admissible φ, the pointwise stationarity condition is

$$(1 - 1/\beta)\big(1 + \log q(x)\big) - \log p_{\text{ref}}(x) + \lambda = 0.$$

Solving for q yields

$$\log q(x) = \frac{\log p_{\text{ref}}(x) - \lambda - (1 - 1/\beta)}{1 - 1/\beta} \;=\; \kappa \log p_{\text{ref}}(x) + C,$$

where $\kappa = \beta/(\beta - 1) > 1$ and $C = \left[-\lambda - (1 - 1/\beta)\right]/(1 - 1/\beta)$ is a constant. Exponentiating and renormalising gives the tempered distribution

$$q(x) = \frac{e^C p_{\text{ref}}(x)^\kappa}{\int_\Omega e^C p_{\text{ref}}(x)^\kappa \, d\mu} \; = \; \frac{p_{\text{ref}}(x)^\kappa}{Z_\beta},$$

with Z_β the normalisation constant.

Let $p_j = p_{\text{ref}}(j)$. Under the optimal distribution $q^\star$, the signal-to-noise ratio $\text{SNR}_{\Delta_{j^\star}}$ takes the form

$$\begin{aligned}
\text{SNR}_{\Delta_{j^\star}}(\kappa) =& \frac{(q^\star_{\hat{c}} - q^\star_{j^\star})^2}{(q^\star_{\hat{c}} + q^\star_{j^\star}) - (q^\star_{\hat{c}} - q^\star_{j^\star})^2} = \frac{(p^\kappa_{\hat{c}} - p^\kappa_{j^\star})^2}{(p^\kappa_{\hat{c}} + p^\kappa_{j^\star}) \sum_i p^\kappa_i - (p^\kappa_{\hat{c}} - p^\kappa_{j^\star})^2} \\
=& \frac{(p^\kappa_{\hat{c}} - p^\kappa_{j^\star})^2}{4 p^\kappa_{\hat{c}} p^\kappa_{j^\star} + (p^\kappa_{\hat{c}} + p^\kappa_{j^\star}) \sum_{i \neq \hat{c}, j^\star} p^\kappa_i} = \frac{\left(\left(\frac{p_{\hat{c}}}{p_{j^\star}}\right)^\kappa - 1\right)^2}{4\left(\frac{p_{\hat{c}}}{p_{j^\star}}\right)^\kappa + \left(\left(\frac{p_{\hat{c}}}{p_{j^\star}}\right)^\kappa + 1\right) \sum_{i \neq \hat{c}, j^\star} \left(\frac{p_i}{p_{j^\star}}\right)^\kappa}.
\end{aligned}$$

To study the behaviour of $\text{SNR}_{\Delta_{j^\star}}$ as a function of κ, we calculate its derivative with respect to κ. To do so define

$$s(\kappa) = \left(\frac{p_{\hat{c}}}{p_{j^\star}}\right)^\kappa \geq 1 \quad \text{and} \quad r(\kappa) = \sum_{i \neq \hat{c}, j^\star} \left(\frac{p_i}{p_{j^\star}}\right)^\kappa \geq 0.$$

Differentiating $\text{SNR}_{\Delta_{j^\star}}(\kappa)$ with respect to κ gives

$$\begin{aligned}
\frac{d}{d\kappa} \text{SNR}_{\Delta_{j^\star}}(\kappa) =& s'(\kappa) \frac{2(s(\kappa) - 1)\left(4s(\kappa) + (s(\kappa) + 1)\, r(\kappa)\right) - (s(\kappa) - 1)^2 (4 + r(\kappa))}{\left(4s(\kappa) + (s(\kappa) + 1)\, r(\kappa)\right)^2} \\
&- r'(\kappa) \frac{(s(\kappa) - 1)^2 (s(\kappa) + 1)}{\left(4s(\kappa) + (s(\kappa) + 1)\, r(\kappa)\right)^2} \\
=& s'(\kappa)(s(\kappa) - 1) \frac{4s(\kappa) + 3r(\kappa) + s(\kappa) r(\kappa) + 4}{\left(4s(\kappa) + (s(\kappa) + 1)\, r(\kappa)\right)^2} \\
&- r'(\kappa) \frac{(s(\kappa) - 1)^2 (s(\kappa) + 1)}{\left(4s(\kappa) + (s(\kappa) + 1)\, r(\kappa)\right)^2}
\end{aligned}$$

Since $s(\kappa) - 1 \geq 0$ and $r(\kappa) \geq 0$, it is sufficient to show that $s'(\kappa) \geq 0$ and $r'(\kappa) \leq 0$ in order to conclude that $\frac{d}{d\kappa} \text{SNR}_{\Delta_{j^\star}}(\kappa) \geq 0$. Indeed,

$$s'(\kappa) = \left(\frac{p_{\hat{c}}}{p_{j^\star}}\right)^\kappa \ln\left(\frac{p_{\hat{c}}}{p_{j^\star}}\right) \geq 0$$

and

$$r'(\kappa) = \sum_{i \neq \hat{c}, j^\star} \left(\frac{p_i}{p_{j^\star}}\right)^\kappa \ln\left(\frac{p_i}{p_{j^\star}}\right) \leq 0,$$

since $p_i \leq p_{j^\star}$ for $i \neq \hat{c}, j^\star$.

This implies that $\text{SNR}_{\Delta_{j^\star}}(\kappa)$ is non-decreasing for $\kappa \geq 1$, showing that entropy-penalising rewards reduce the number of samples required for certification.

Differences from existing entropy-penalising methods. We highlight how our approach differs from that of (Agarwal et al., 2025). Their method minimises individual rewards for a trajectory $(Y_t)_{t \geq 0}$, corresponding to an answer $X = g(Y_{\tau:})$ where τ is a random stopping time. Specifically, they define two entropy-based reward functions

- Negative trajectory-level entropy estimator. The reward for a full trajectory $(Y_t)_{t \geq 0}$ is

$$r_{\text{traj}}(Y_t) = \sum_{t=1}^{|Y^i_t|} \log \pi(Y^i_t | Y^i_{<t}).$$

- Negative token level entropy. In this case, the reward is of the form

$$r_{\text{tok}}(Y_t) = \sum_{t=1}^{|Y_t^i|} \sum_{j \in \mathcal{V}} \pi(j|Y_{<t}^i) \log \pi(j|Y_{<t}^i),$$

  where $\mathcal{V}$ denotes the vocabulary.

While both trajectory-level and token-level rewards aim to minimise entropy, they influence RL training differently: minimising trajectory entropy encourages policies with lower entropy over entire trajectories, whereas minimising token-level entropy encourages policies with low entropy at each generation step. In contrast, our group-level reward function targets the entropy of the final answer distribution, directly improving the model's confidence in its final output while allowing exploration of diverse pathways during the chain-of-thought reasoning process.

D Experimental details

D.1 Experimental setup

We adopt the data and evaluation pipeline from the TTRL codebase (Zuo et al., 2025), which is built on the VERL framework (Sheng et al., 2024). The final answer of the language model is the string inside the last $\backslash boxed\{\}$.

Implementation details. We use hyperparameters similar to those in TTRL (Zuo et al., 2025) and report them here for completeness. A cosine learning rate schedule is applied, with a peak value of 5×10^{-7}, and the AdamW optimiser is used for the policy model with a learning rate of 9×10^{-6}. The KL-regularisation parameter in the RL objective is set to 0.001.

We sample 64 responses per prompt using a temperature of 0.6 (1.0 for Qwen2.5-Math models) for voting-based label estimation, and downsample 32 responses per prompt for training. The maximum generation length is fixed at 3072 tokens for all models. The number of episodes is set to 80, 30, and 10 for AIME 2024, AMC, and MATH-500, respectively, reflecting dataset size. We also apply early stopping with a tolerance of 5×10^{-3} and a patience of 10 iterations, evaluated on both metrics (pass@1 and majority).

All other hyperparameters not explicitly mentioned here are set to their default values in the VERL framework. For the TTRL (Zuo et al., 2025) baseline, we adopt the hyperparameters reported in the paper.

Evaluation details. We also set the maximum generation length to 3072 tokens during evaluation. Following Zuo et al. (2025), we report the pass@1 score using non-zero temperature sampling. Specifically, for each prompt pr, we generate $N = 16$ responses using a temperature of 0.6 and a top-p value of 0.95. The pass@1 score is then computed as

$$\text{pass@1} = \frac{1}{QN} \sum_{pr} \sum_{i=1}^{N} \mathbf{1}\{X_i(pr) = \text{correct}\},$$

where $X_i(pr)$ denotes the i-th generated response for prompt pr and Q is the total number of prompts.

We also report majority vote accuracy, which indicates whether the most frequent answer among the $N = 16$ responses per prompt matches the ground truth

$$\text{majority} = \frac{1}{Q} \sum_{pr} \mathbf{1}\{\text{majority vote}(X_1(pr), \ldots, X_N(pr)) = \text{correct}\}.$$

Computation time. All experiments were conducted on 8×H100 Nvidia GPUs, each with 96GB of memory.

D.2 Additional results

Table 3 expands upon the results presented in Table 1. It reports the pass@1 performance for both the score and the format score before and after applying test-time training with our proposed reward functions. We observe that, for Qwen2.5 models, the improvement in score is notably larger than that in format score, suggesting that test-time training effectively uncovers latent knowledge already present in the model rather than merely correcting format errors. In contrast, for the Llama-3.1-8B model, we hypothesise that the mode of the model's final answer distribution does not coincide with the true answer, therefore, test-time training incorrectly shifts the model's output distribution. That is, the model lacks the necessary mathematical knowledge, and our test-time training strategies serve to reveal rather than create new knowledge. Table 4 presents analogous results for majority vote accuracy, leading to similar conclusions.

Table 5 provides evidence that the model becomes more confident in its outputs after applying test-time training strategies. Specifically, the required number of samples for the MMC stopping rule, denoted as N_{adaptive} is lower after test-time training compared to the pre-trained model.

The relationship between N_{adaptive} and N_{budget} can be accurately modelled by a linear regression of the form $N_{\text{adaptive}} = \alpha + \beta N_{\text{budget}}$ with a coefficient of determination R^2 very close to 1. We therefore report the estimated value of β obtained via least squares fitting. Since $0 \leq N_{\text{adaptive}} \leq N_{\text{budget}}$, it follows that $\beta \leq 1$.

Table 3: Comparison of pass@1 performance for the score and format score (using 16 samples per prompt) before and after applying test-time training.

	AIME		**AMC**		**Math-500**	
	Score	**Format score**	**Score**	**Format score**	**Score**	**Format score**
Qwen2.5-7B	9.4	84.6	31.2	84.6	59.1	90.2
SNR	23.3 +13.9	100.0 +15.4	51.8 +20.6	99.5 +14.9	80.3 +21.2	98.9 +8.7
Entropy	20.0 +10.6	100.0 +15.4	49.2 +18.0	99.5 +14.9	77.6 +18.5	100.0 +9.8
Llama-3.1-8B	4.4	60.0	21.8	72.0	48.2	83.8
SNR	13.4 +9.0	99.6 +39.6	29.3 +7.5	100.0 +28.0	59.2 +11.0	100.0 +16.2
Entropy	13.3 +8.9	99.8 +39.8	27.0 +5.2	100.0 +28.0	55.4 +7.2	100.0 +16.2
Qwen2.5-Math-7B	10.6	73.5	31.0	85.4	47.1	90.2
SNR	36.7 +26.1	85.4 +11.9	65.0 +34.0	88.8 +3.4	84.5 +37.4	97.5 +7.3
Entropy	38.3 +27.7	97.5 +24.0	65.4 +34.4	99.9 +14.5	82.4 +35.3	99.3 +9.1
Qwen2.5-Math-1.5B	7.1	74.2	28.1	80.1	31.4	66.4
SNR	16.3 +9.2	91.9 +17.7	45.4 +17.3	92.2 +12.1	72.0 +40.6	97.7 +11.3
Entropy	15.6 +8.5	88.3 +14.1	45.9 +17.8	96.2 +16.1	70.8 +39.4	98.1 +11.7

Table 4: Comparison of majority vote accuracy for the score and format score (using 16 samples per prompt) before and after applying test-time training.

	AIME		**AMC**		**Math-500**	
	Score	**Format score**	**Score**	**Format score**	**Score**	**Format score**
Qwen2.5-7B	16.7	72.8	41.8	79.6	73.5	93.4
SNR	23.3 +6.6	100.0 +27.2	51.2 +9.4	100.0 +20.4	81.0 +7.5	98.9 +5.5
Entropy	20.0 +3.3	100.0 +27.2	49.4 +7.6	100.0 +20.4	79.0 +5.5	100.0 +6.6
Llama-3.1-8B	4.6	26.8	27.4	53.4	57.7	77.2
SNR	13.3 +8.7	99.8 +73.0	28.6 +1.2	100.0 +46.6	60.3 +2.6	100.0 +22.8
Entropy	13.3 +8.7	100.0 +73.2	29.3 +1.9	100.0 +46.6	57.6 -0.1	100.0 +22.8
Qwen2.5-Math-7B	16.5	56.3	41.5	78.4	59.5	87.8
SNR	37.8 +21.3	77.9 +21.6	67.2 +25.7	87.9 +9.5	85.7 +26.2	99.5 +11.7
Entropy	36.7 +20.2	97.3 +41.0	66.1 +24.6	100.0 +22.6	84.3 +24.8	99.5 +11.7
Qwen2.5-Math-1.5B	11.7	60.0	37.2	70.8	36.5	57.4
SNR	23.7 +12.0	90.5 +30.5	53.3 +16.1	90.9 +20.1	78.9 +42.4	97.8 +40.4
Entropy	23.2 +11.5	81.3 +21.3	52.8 +15.6	95.7 +24.9	77.4 +40.9	98.2 +40.8

We observe that the estimated slope for the pre-trained model, $\hat{\beta}_{\text{pre}}$, is larger than that of the test-time trained model, $\hat{\beta}_{\text{post}}$. This reduction is particularly pronounced for the smaller 1.5B model, suggesting that larger models experience diminishing returns from test-time training.

These results are consistent with the larger increase in the estimated $\text{SNR}(\Delta_{j_n^\star})$ observed during training. Recall from (5) that the required number of samples for the MMC stopping rule is approximately inversely proportional to the $\text{SNR}(\Delta_{j_n^\star})$. Figure 5 shows the evolution of the estimated $\text{SNR}(\Delta_{j_n^\star})$ when using SNR-based rewards, as well as the negative entropy when training with entropy-based rewards, measured on the training dataset. We also include the evolution of the pass@1 performance on the validation dataset.

Finally, Figures 6-9 provide a detailed analysis, for each difficulty level in the MATH-500 dataset, of the distributions of the estimated lower bound on the probability $\mathbb{P}[\widehat{c}_n = c^\star]$, as well as the estimated $\text{SNR}(\Delta_{j_n^\star})$ when applying the MMC adaptive sampling scheme under two confidence levels, $\varepsilon = 0.1$ and 0.4. The lower bound estimates of $\mathbb{P}[\widehat{c}_n = c^\star]$ (Figures 6, 7) are computed using a Beta approximation (see Appendix B.2 for details). Results are reported after test-time training with SNR-based rewards. The SNR plots (Figures 8, 9) further illustrate how SNR can serve as a label-free estimator of problem difficulty.

Table 5: Regression coefficients from fitting the required number of samples under the MMC stopping rule as a function of the budget, $N_{\text{adaptive}} = \alpha + \beta N_{\text{budget}}$, for $\varepsilon = 0.1$ and 0.4. Results contrast the pre-trained model with the model after test-time training using SNR-based rewards.

	Qwen2.5-Math-7B		**Qwen2.5-Math-1.5B**		**Qwen2.5-7B**		**Llama-3.1-8B**	
ε	0.1	0.4	0.1	0.4	0.1	0.4	0.1	0.4
$\hat{\boldsymbol{\beta}}_{\text{pre}}$ pre-trained model	0.725	0.711	0.848	0.798	0.627	0.589	0.645	0.590
$\hat{\boldsymbol{\beta}}_{\text{post}}$ test-time trained model	0.631	0.568	0.570	0.533	0.472	0.392	0.564	0.488
$\nabla = \hat{\boldsymbol{\beta}}_{\text{pre}} - \hat{\boldsymbol{\beta}}_{\text{post}}$	0.094	0.143	0.237	0.265	0.155	0.197	0.081	0.102

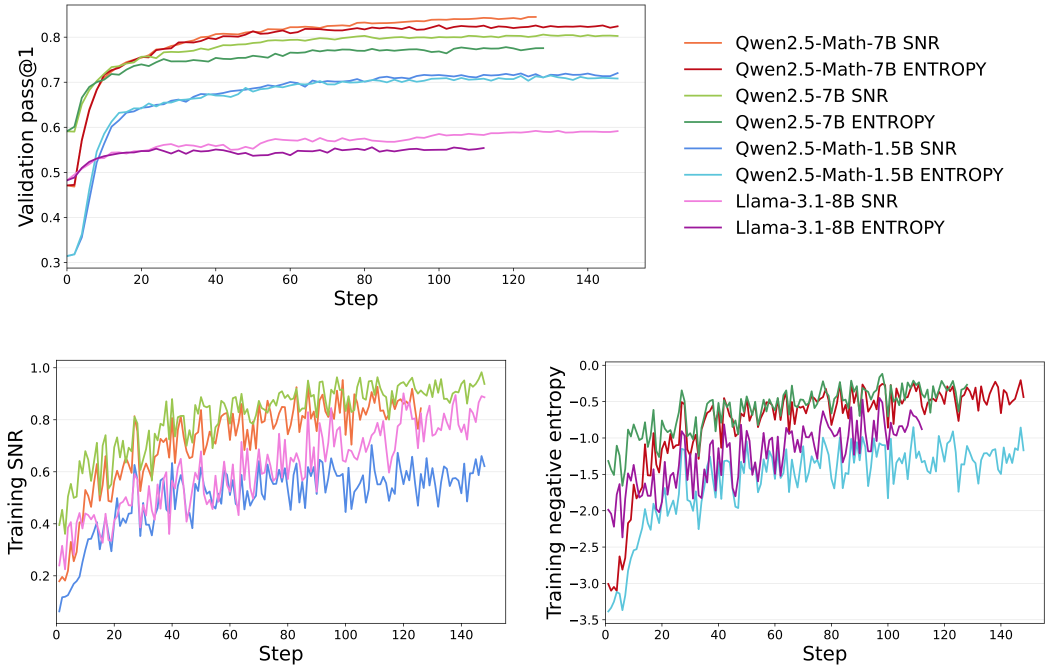
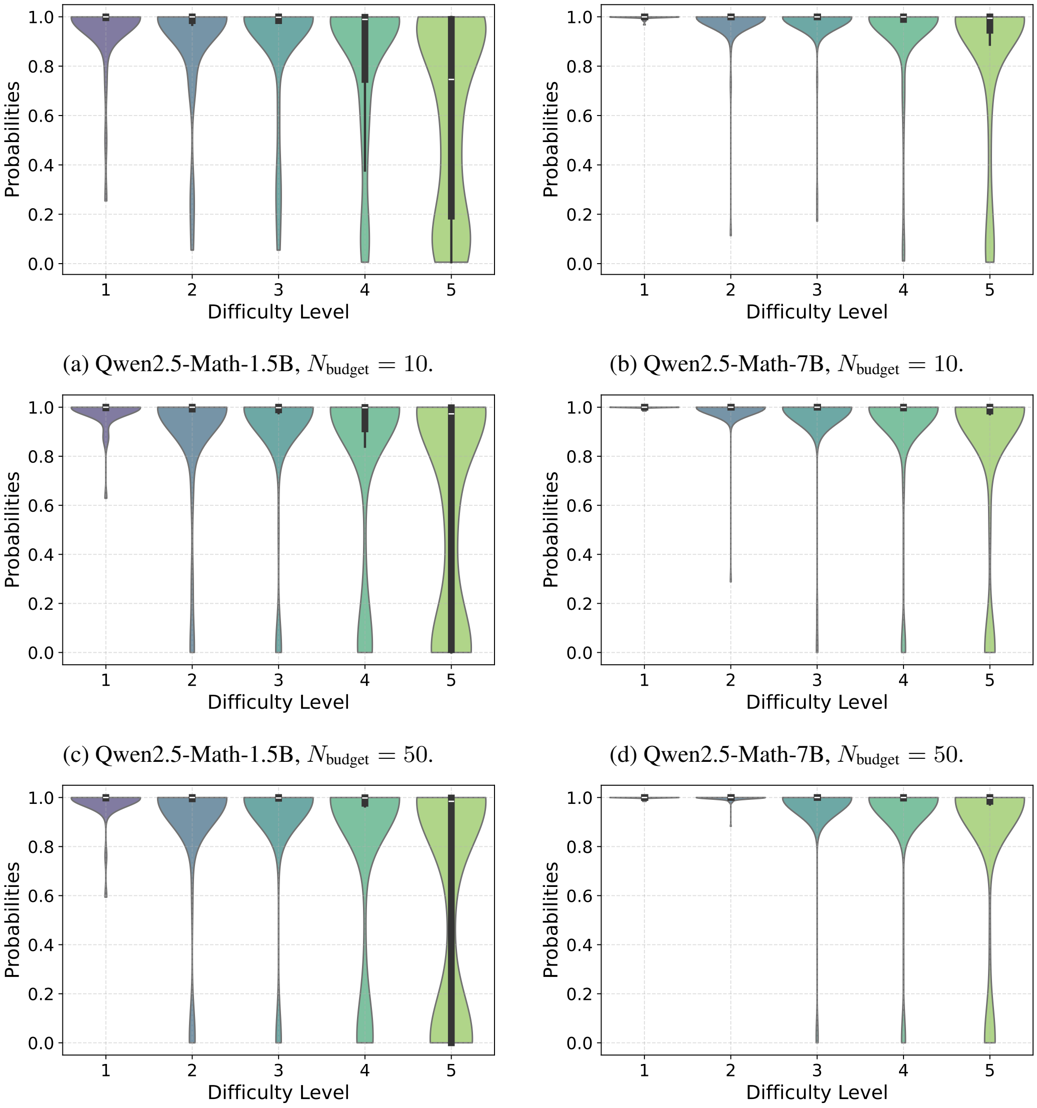


Figure 5: Evolution of different training and validation metrics on the MATH-500 dataset.



(a) Qwen2.5-Math-1.5B, $N_{\text{budget}} = 10$.

(b) Qwen2.5-Math-7B, $N_{\text{budget}} = 10$.

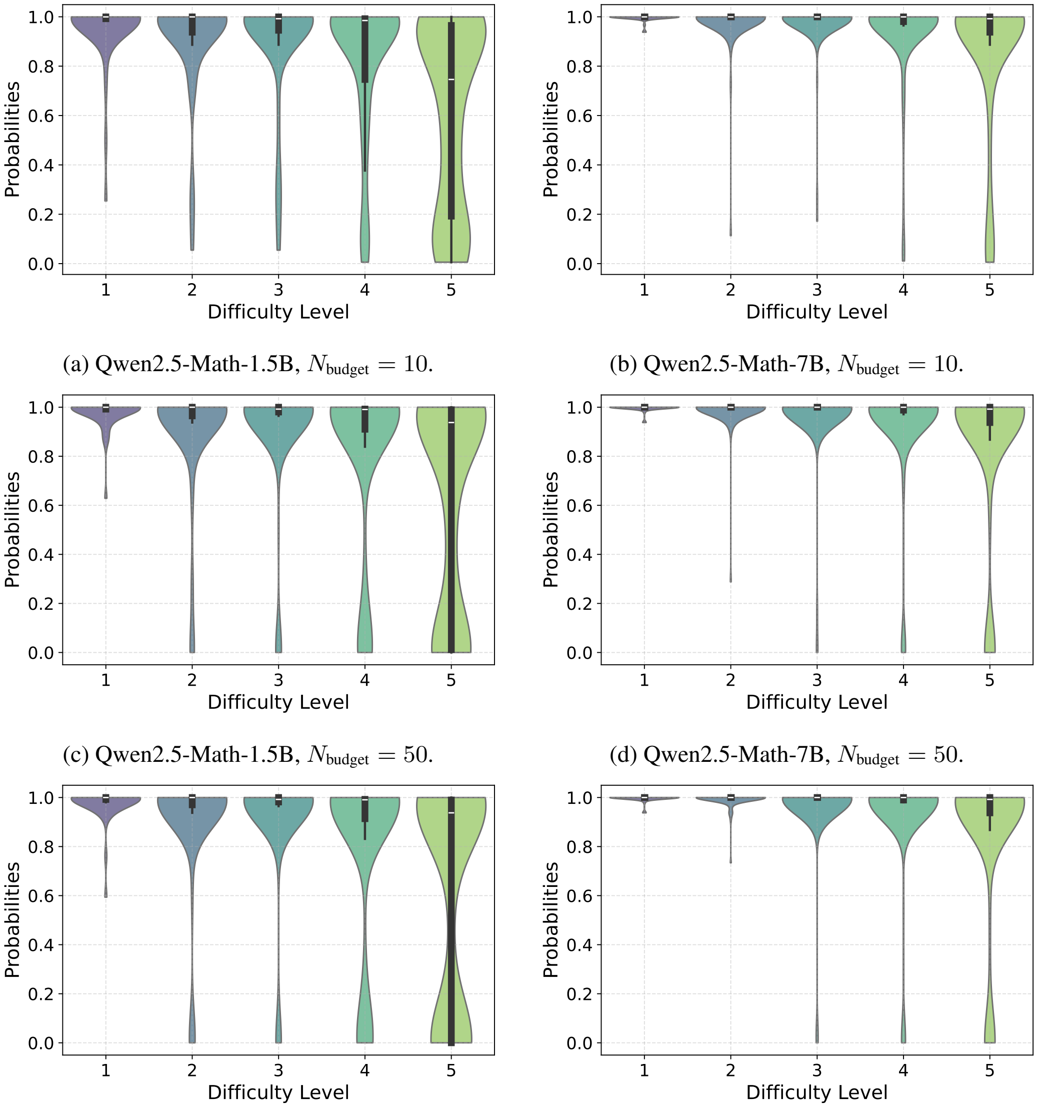
(c) Qwen2.5-Math-1.5B, $N_{\text{budget}} = 50$.

(d) Qwen2.5-Math-7B, $N_{\text{budget}} = 50$.

(e) Qwen2.5-Math-1.5B, $N_{\text{budget}} = 100$.

(f) Qwen2.5-Math-7B, $N_{\text{budget}} = 100$.

Figure 6: Violin plots illustrating the distribution of the estimated lower bound on the probability $\mathbb{P}[\widehat{c}_n = c^\star]$ when applying Martingale Majority Certificate stopping rule with $\varepsilon = 0.1$ across different budget values N_{budget}. Results are obtained after test-time training with SNR-based rewards on the MATH-500 dataset.



(a) Qwen2.5-Math-1.5B, $N_{\text{budget}} = 10$.

(b) Qwen2.5-Math-7B, $N_{\text{budget}} = 10$.

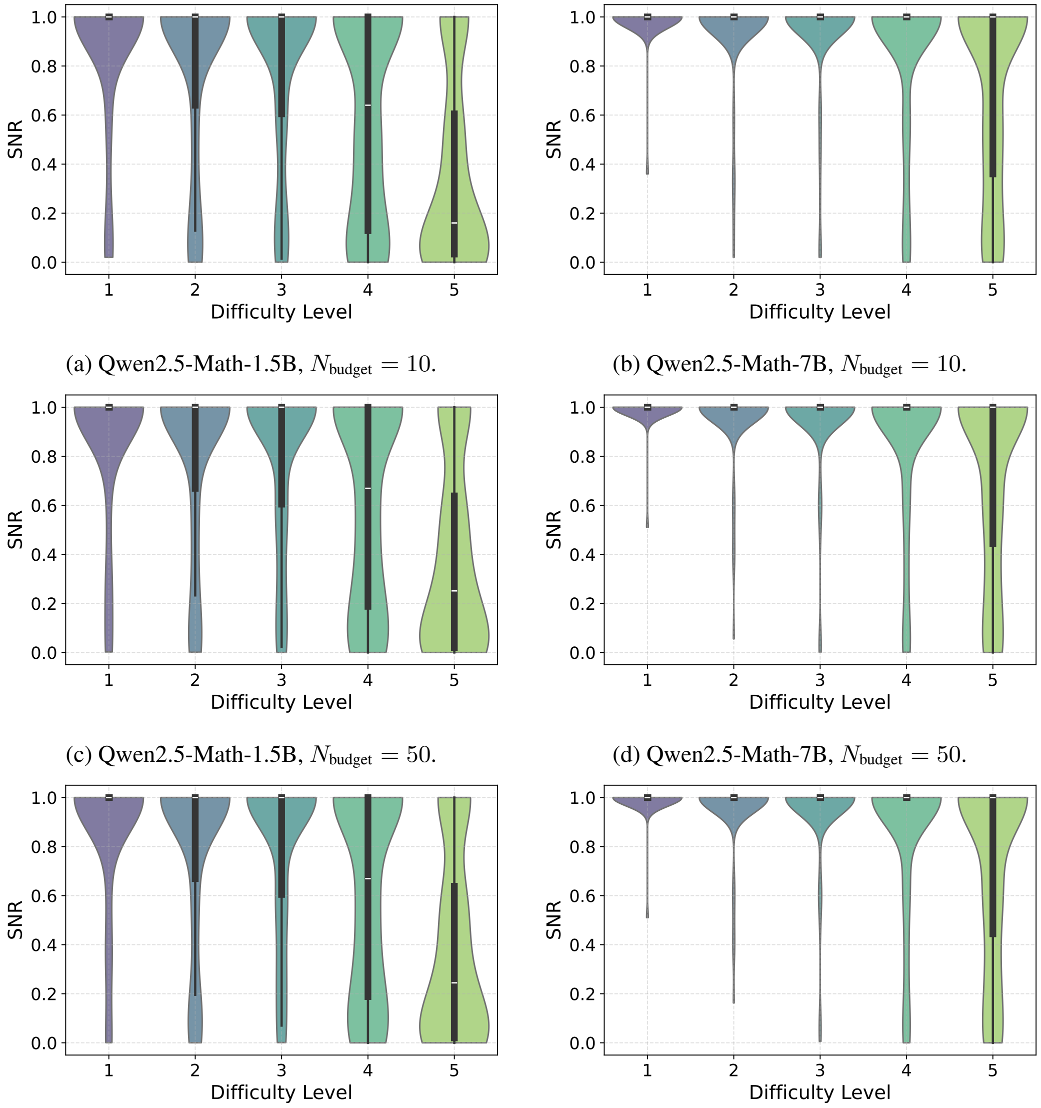
(c) Qwen2.5-Math-1.5B, $N_{\text{budget}} = 50$.

(d) Qwen2.5-Math-7B, $N_{\text{budget}} = 50$.

(e) Qwen2.5-Math-1.5B, $N_{\text{budget}} = 100$.

(f) Qwen2.5-Math-7B, $N_{\text{budget}} = 100$.

Figure 7: Violin plots illustrating the distribution of the estimated lower bound on the probability $\mathbb{P}[\widehat{c}_n = c^\star]$ when applying Martingale Majority Certificate stopping rule with $\varepsilon = 0.4$ across different budget values N_{budget}. Results are obtained after test-time training with SNR-based rewards on the MATH-500 dataset.



(a) Qwen2.5-Math-1.5B, $N_{\text{budget}} = 10$.

(b) Qwen2.5-Math-7B, $N_{\text{budget}} = 10$.

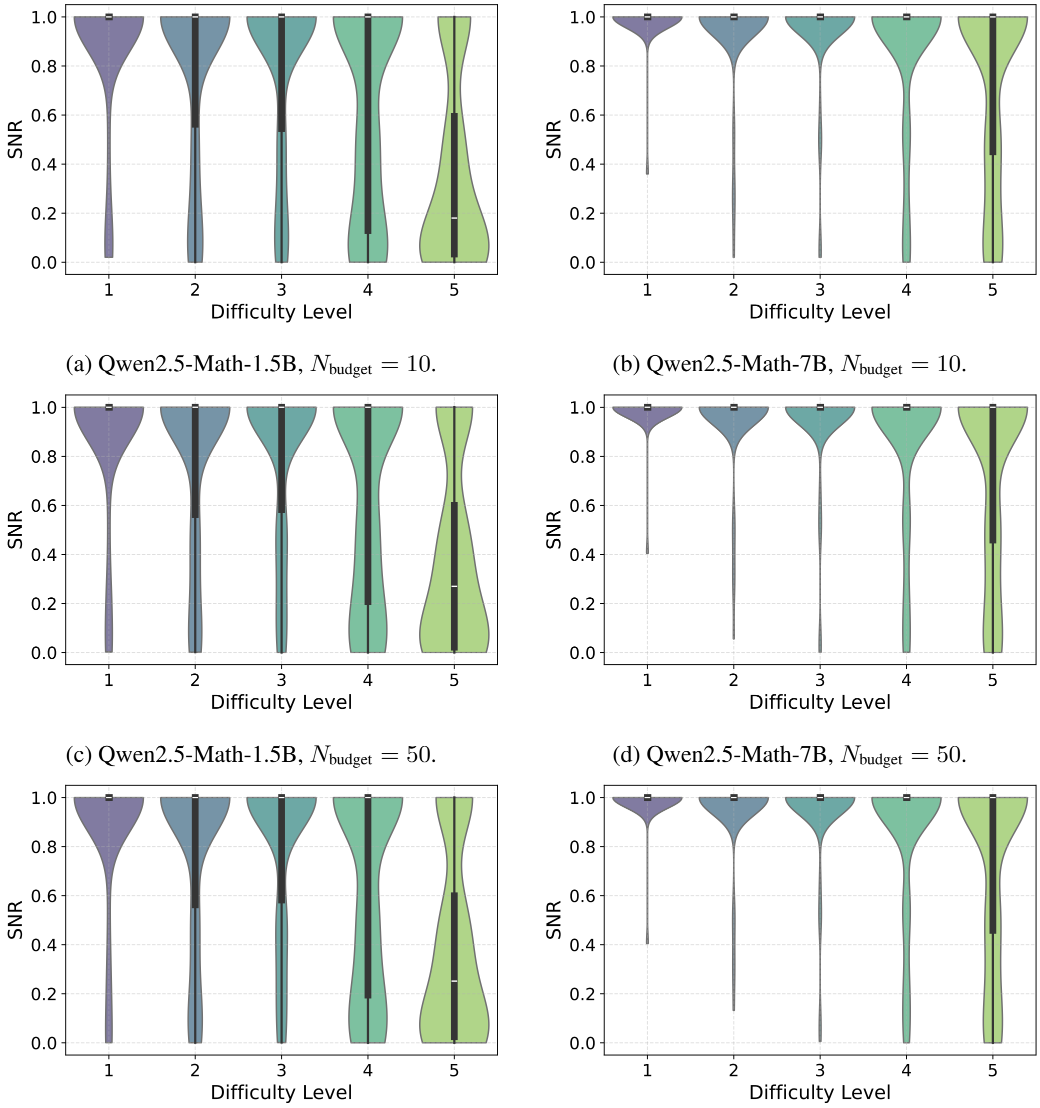
(c) Qwen2.5-Math-1.5B, $N_{\text{budget}} = 50$.

(d) Qwen2.5-Math-7B, $N_{\text{budget}} = 50$.

(e) Qwen2.5-Math-1.5B, $N_{\text{budget}} = 100$.

(f) Qwen2.5-Math-7B, $N_{\text{budget}} = 100$.

Figure 8: Violin plots showing the distribution of the estimated signal-to-noise ratio between the leader and runner-up, $\text{SNR}(\Delta_{j_n^\star})$, when using Martingale Majority Certificate stopping rule with $\varepsilon = 0.1$ across different budget values N_{budget}. Results are obtained after applying test-time training with SNR-based rewards on the MATH-500 dataset.



(a) Qwen2.5-Math-1.5B, $N_{\text{budget}} = 10$.

(b) Qwen2.5-Math-7B, $N_{\text{budget}} = 10$.

(c) Qwen2.5-Math-1.5B, $N_{\text{budget}} = 50$.

(d) Qwen2.5-Math-7B, $N_{\text{budget}} = 50$.

(e) Qwen2.5-Math-1.5B, $N_{\text{budget}} = 100$.

(f) Qwen2.5-Math-7B, $N_{\text{budget}} = 100$.

Figure 9: Violin plots showing the distribution of the estimated signal-to-noise ratio between the leader and runner-up, $\text{SNR}(\Delta_{j_n^\star})$, when using Martingale Majority Certificate stopping rule with $\varepsilon = 0.4$ across different budget values N_{budget}. Results are obtained after applying test-time training with SNR-based rewards on the MATH-500 dataset.