Ranking Reasoning LLMs under Test-Time Scaling

Mohsen Hariri[1], Michael Hinczewski[2], Jing Ma[1], Vipin Chaudhary[1],
[1]Department of Computer and Data Sciences, [2]Department of Physics
Case Western Reserve University, Cleveland, OH, USA
mohsen.hariri@case.edu

Abstract

Test-time scaling evaluates reasoning LLMs by sampling multiple outputs per prompt, but ranking models in this regime remains underexplored. We formalize dense benchmark ranking under test-time scaling and introduce `Scorio`, a library that implements statistical ranking methods such as paired-comparison models, item response theory (IRT) models, voting rules, and graph- and spectral-based methods. Across 20 reasoning models on four Olympiad-style math benchmarks (AIME'24, AIME'25, HMMT'25, and BrUMO'25; up to $N = 80$ trials), most full-trial rankings agree closely with the Bayesian gold standard $\mathrm{Bayes}_{\mathcal{U}}@80$ (mean Kendall's $\tau_b = 0.93$–0.95), and 19–34 methods recover exactly the same ordering. In the single-trial regime, the best methods reach $\tau_b \approx 0.86$. Using greedy decoding as an empirical prior ($\mathrm{Bayes}_{\mathbf{R}_0}@N$) reduces variance at $N = 1$ by 16–52%, but can bias rankings when greedy and stochastic sampling disagree. These results identify reliable ranking methods for both high- and low-budget test-time scaling. We release `Scorio` as an open-source library at `Scorio`[1].

1 Introduction

Large language models (LLMs) are increasingly used as general-purpose reasoning systems for tasks such as programming and mathematical problem solving (Chen et al., 2021; Wang et al., 2023). Reliable evaluation is therefore essential. In many settings, what matters is not only an absolute score but also a *ranking* that supports model selection, deployment, and scientific comparison. This need is amplified by *test-time scaling*, which allocates additional inference compute by sampling multiple outputs per prompt and aggregating them, turning evaluation into a repeated-sampling problem (Wang et al., 2023; Snell et al., 2024; Zeng et al., 2025).

Statistical ranking methods underpin two common LLM workflows. First, preference-based learning and alignment pipelines rely on human or model preferences over alternative responses, where the primitive observations are paired comparisons and downstream optimization depends on how those preferences are modeled and aggregated (Christiano et al., 2017; Rafailov et al., 2023). Second, model comparisons are often communicated through leaderboards. Crowdsourced paired-comparison platforms such as Chatbot Arena collect head-to-head judgments and fit rating or paired-comparison models to produce public rankings (Chiang et al., 2024), while benchmark-style evaluations rank models by task performance metrics such as Pass@k (Chen et al., 2021). Recent work has revisited the statistical foundations of LLM ranking in both preference-based settings (Ameli et al., 2025) and benchmark settings, including IRT-style benchmarking (Zhou et al., 2025). Different ranking methods can produce noticeably different model orderings, and their agreement can vary with benchmark difficulty (Fig. 1).

A key practical distinction between these regimes is the *representation* of the data used for ranking. Preference-based evaluation typically yields a sparse and evolving comparison graph because only a subset of model pairs are compared and the model pool changes over time (Chiang et al., 2024). In contrast, benchmark evaluations produce dense outcomes for every model–question pair. For a fixed set of L models and M questions, we observe an outcome for every pair. Under test-time scaling, each model–question pair is evaluated with N independent trials, producing a response tensor $\mathbf{R} \in \{0, 1\}^{L \times M \times N}$. This dense repeated-trial setting raises new methodological questions: Which ranking rule should be used when N is small? How quickly do different ranking methods

[1] https://github.com/mohsenhariri/scorio. See Appendix J for API documentation.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 33437–33478

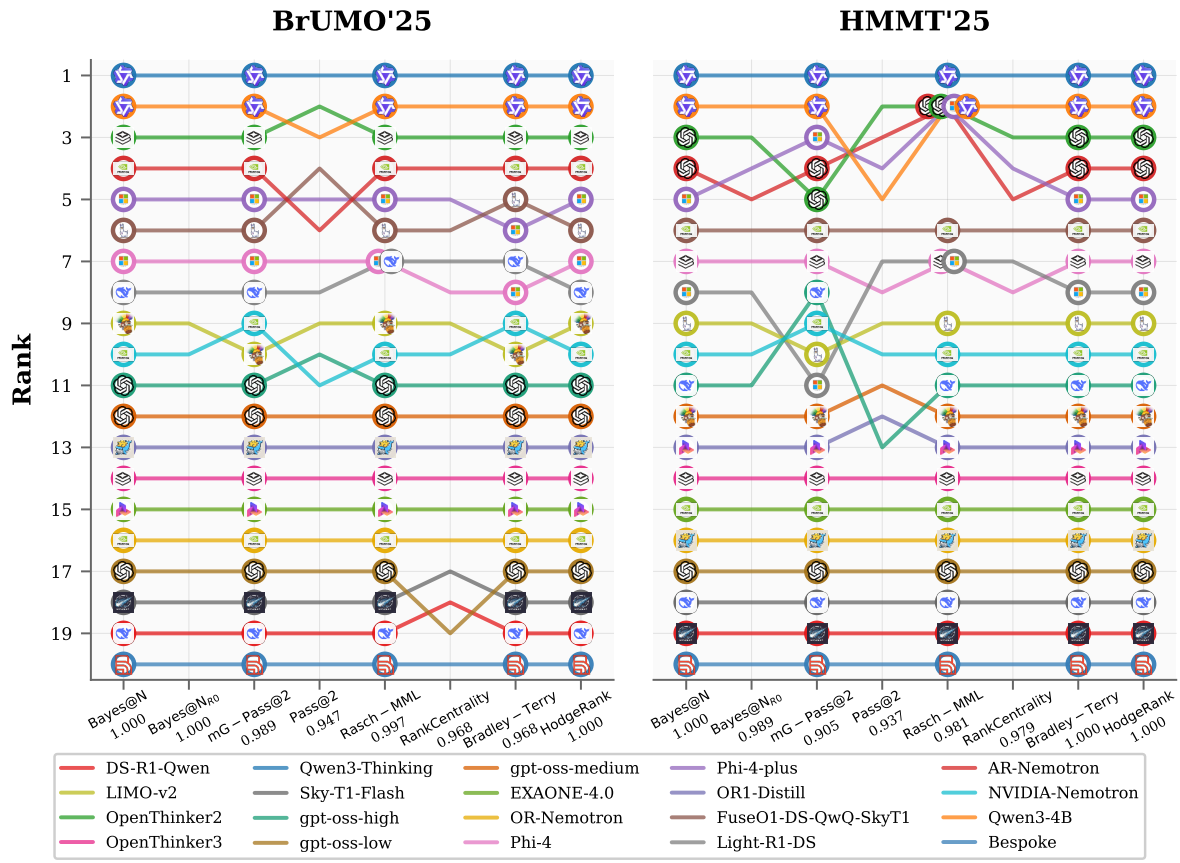

Figure 1: Agreement between each method's full-trial ranking and the gold standard. Kendall's τ_b is computed between each method's ranking (at $N = 80$ trials) and $\text{Bayes}_{\mathcal{U}}@80$ on an easier benchmark (BrUMO'25, left) and the hardest benchmark (HMMT'25, right). On BrUMO'25, multiple methods achieve near-perfect or perfect agreement: $\text{Bayes}_{\mathbf{R}_0}@N$ and HodgeRank reach $\tau_b = 1.0$, while Rasch MML achieves 0.997. On HMMT'25, Bradley–Terry and HodgeRank maintain perfect agreement ($\tau_b = 1.0$), but $\text{Bayes}_{\mathbf{R}_0}@N$ drops to 0.989 and Pass@2 falls to 0.937. This divergence is consistent with the lower greedy–sampling alignment observed on harder benchmarks (Section 3.4).

stabilize as N grows? How do priors and uncertainty estimates affect ranking robustness?

In this work, we study performance-based ranking under test-time scaling. We formalize the dense benchmark setting through the response tensor $\mathbf{R}$, evaluate ranking methods by their low-budget stability and convergence as the test-time budget increases, and implement the studied methods in `Scorio`.

We summarize our contributions as follows:

- We formalize dense benchmark ranking under test-time scaling via $\mathbf{R} \in \{0,1\}^{L\times M\times N}$ and connect common ranking families through pointwise, pairwise, and setwise transformations of $\mathbf{R}$.
- We propose an evaluation protocol based on low-budget stability (agreement between rankings computed from subsampled trials and reference rankings) and convergence with increasing numbers of trials.
- We compare a broad suite of ranking methods across 20 reasoning models and four Olympiad-style math benchmarks (up to $N = 80$ trials), characterizing where method families agree and where they diverge.
- We analyze Bayesian and uncertainty-aware ranking choices, including priors and conservative (quantile-based) scoring, and quantify their bias–variance trade-offs in low-trial regimes.
- We release `Scorio`, a library implementing the ranking methods and Bayesian options.

2 Ranking Problem and Test-time Scaling

In classical statistical settings, there is no canonical theoretical ground truth or empirical gold stan-

dard against which competing ranking rules can be judged. Choosing among methods therefore usually requires additional modeling assumptions. Test-time scaling offers a useful alternative: because each model–question pair can be sampled repeatedly, it lets us evaluate ranking methods by how stable they are in low-budget settings and how quickly they converge as more trials are observed.

Statistical ranking methods are widely used in domains such as sports competitions (e.g., paired-comparison models and rating systems for head-to-head games) (Bradley and Terry, 1952; Elo, 1978; Glickman, 1999) and voting or collective decision-making (de Borda, 1781; Condorcet, 1785; Arrow, 1951). In such settings, there are L entities to be ranked (e.g., players, items, or models) over M tasks (e.g., matches, questions, or instances). Test-time scaling adds a third dimension: N, the number of i.i.d. samples generated for a fixed question $m \in \{1, \dots, M\}$. Repeated sampling lets us study two complementary properties. First, *low-budget stability* asks whether a ranking computed from a small number of trials agrees with a high-budget reference ranking. In our experiments, the low-budget case is $N = 1$: we subsample one trial per question, compute the ranking, repeat this over the available single-trial draws, and compare each ranking either with an empirical gold standard or with the same method's full-trial ranking. Second, *convergence* asks how quickly rankings computed from n trials approach the full-trial ordering as n increases from 1 to N.

2.1 Gold Standard Rankings

Evaluation metrics widely used in test-time scaling, such as Pass@k and Bayes@N, can be analyzed through statistical properties such as bias. For instance, Chen et al. (2021) derive an unbiased estimator for Pass@k. As the number of trials N grows, empirical estimates of these metrics concentrate around their population values, making metric-based rankings increasingly stable. In particular, for binary outcomes, $\mathrm{Bayes}_{\mathcal{U}}@N$ is order-equivalent to mean accuracy avg@N (Hariri et al., 2026), which motivates our use of the full-trial $\mathrm{Bayes}_{\mathcal{U}}@N$ ranking as an empirical accuracy-based gold standard.

This reasoning does not extend automatically to all ranking methods. Even as the number of questions M or trials N increases, different ranking methods need not converge to a unique limiting ordering, such as the one induced by average accuracy (Appendix C.1). Unlike evaluation metrics, ranking algorithms can emphasize different aspects of performance across tasks, players, or items. In Section 3, we show that rankings induced by probabilistic models (e.g., Bradley–Terry) can differ from those induced by expected-performance metrics (e.g., mean accuracy or Bayesian estimates).

Given the absence of a universal gold standard for ranking methods, we use two target rankings for comparison. First, we define an empirical gold standard based on average performance over all trials with a large sample size (e.g., $N = 80$). This target captures aggregate performance across tasks and trials while allowing ties. This choice is justified for several reasons: (a) the ranking induced by average performance is order-equivalent to the ranking induced by Bayesian estimation with a uniform prior ($\mathrm{Bayes}_{\mathcal{U}}@N$); (b) when N is large, average performance is among the most stable ranking rules relative to the alternatives (Section 3.1); and (c) it is easy to interpret, widely used in practice, and yields absolute performance values.

The second target ranking is the ordering produced by a method itself (method@80) when all available trials are aggregated. This target lets us assess a method's self-consistency and convergence as more data become available.

2.2 Representation

We consider L models evaluated on a benchmark of M questions under test-time scaling, generating N i.i.d. trials per model–question pair. Let $\mathcal{L} = \{1, \dots, L\}$ index models and $\mathcal{Q} = \{1, \dots, M\}$ questions; for each question we observe N independent trials indexed by $n \in \{1, \dots, N\}$. For each $(l, m, n) \in \mathcal{L} \times \mathcal{Q} \times \{1, \dots, N\}$ we observe a binary outcome

$$R_{lmn} \in \{0, 1\}, \tag{1}$$

where $R_{lmn} = 1$ if model l solves question m on trial n. We collect these outcomes in a response tensor $\mathbf{R} \in \{0, 1\}^{L \times M \times N}$. When $N = 1$, this reduces to the standard single-run benchmark setting. Unlike crowdsourced paired-comparison datasets (e.g., Chatbot Arena (Chiang et al., 2024)), where the primitive observations are model–model outcomes on a possibly sparse comparison graph, our benchmark setting produces outcomes for every model–question pair. We discuss the Arena sparse-comparison counterpart in Appendix G. We therefore take $\mathbf{R}$ as the primitive object; all ranking

methods we study use $\mathbf{R}$ as input, but they differ in the representations on which they operate after transforming or aggregating it.

Pointwise (model–question) representation. Define the per-question solve rate

$$\widehat{p}_{lm} := \frac{1}{N} \sum_{n=1}^{N} R_{lmn}, \tag{2}$$

and the overall mean accuracy $\widehat{p}_l := \frac{1}{M} \sum_{m=1}^{M} \widehat{p}_{lm}$. Pointwise and IRT-style methods operate on the matrix $\widehat{\mathbf{P}} = [\widehat{p}_{lm}] \in [0,1]^{L \times M}$ (or on its row means), optionally reweighting questions (e.g., inverse-difficulty weighting (Gotou et al., 2020)). Classical IRT models infer latent abilities from this representation (Rasch, 1960; Birnbaum, 1968), and have recently been applied to LLM benchmarking (Zhou et al., 2025). When $N > 1$, the trial axis corresponds to repeated Bernoulli observations; likelihood-based models (including IRT) can equivalently work with the sufficient statistic $k_{lm} := \sum_n R_{lmn}$, yielding a binomial-response formulation (McCullagh and Nelder, 1989; De Boeck and Wilson, 2004). Related repeated-measures and longitudinal IRT extensions are also well studied (Verhelst and Glas, 1993; Wang and Nydick, 2020). Evaluation-metric rankings (e.g., Pass@k and Bayes@N) additionally use the per-question trial multiset $\{R_{lm1}, \ldots, R_{lmN}\}$ (equivalently the count $\sum_n R_{lmn}$) to compute per-question metrics before aggregating across m (Chen et al., 2021).

Pairwise (win/tie) representation. Many classical ranking methods reduce $\mathbf{R}$ to pairwise outcomes. For a pair of models $(i, j) \in \mathcal{L}^2$ we define win and tie counts

$$W_{ij} := \sum_{m=1}^{M} \sum_{n=1}^{N} \mathbf{1}\{R_{imn} = 1,\ R_{jmn} = 0\}, \tag{3}$$

$$T_{ij} := \sum_{m=1}^{M} \sum_{n=1}^{N} \mathbf{1}\{R_{imn} = R_{jmn}\}, \tag{4}$$

so that, in our fully observed setting, $W_{ij} + W_{ji} + T_{ij} = MN$ for all $i \neq j$. Equivalently, we can form an undirected comparison graph $G = (V, E)$ with vertex set $V = \mathcal{L}$ and edge set $E = \{\{i, j\} : W_{ij} + W_{ji} + T_{ij} > 0\}$, and store (W_{ij}, W_{ji}, T_{ij}) on each edge. In our benchmark setting E is the complete graph (every pair is compared MN times), whereas in interactive evaluation settings E is typically sparse and one assumes G is connected. The matrices $\mathbf{W} = [W_{ij}]$ and $\mathbf{T} = [T_{ij}]$ define a weighted comparison graph over models. Probabilistic paired-comparison models (e.g., Bradley–Terry and tie extensions (Bradley and Terry, 1952; Rao and Kupper, 1967; Davidson, 1970)) and voting rules (e.g., Borda and Copeland (de Borda, 1781; Brandt et al., 2016)) use these aggregated counts; graph- and spectral-based methods (e.g., PageRank, Rank Centrality, HodgeRank, SerialRank, AlphaRank, and Nash-based ranking (Page et al., 1999; Negahban et al., 2017; Jiang et al., 2011; Fogel et al., 2016; Omidshafiei et al., 2019; Balduzzi et al., 2019)) further transform $(\mathbf{W}, \mathbf{T})$ into Markov chains or skew-symmetric edge flows, typically via edge weights based on empirical win rates such as $\widehat{P}_{i \succ j} = (W_{ij} + \frac{1}{2} T_{ij})/(W_{ij} + W_{ji} + T_{ij})$. Sequential rating systems (e.g., Elo and TrueSkill (Elo, 1978; Herbrich et al., 2006)) instead process the underlying stream of pairwise "matches" induced by each question–trial (m, n).

Listwise or setwise representation. For each question–trial (m, n) we define the winning set $U_{mn} := \{l \in \mathcal{L} : R_{lmn} = 1\}$ and the losing set $\mathcal{L} \setminus U_{mn}$, which induces a two-level partial order: all winners tie above all losers. Setwise or listwise models (e.g., Plackett–Luce (Plackett, 1975; Luce, 1959) and Davidson–Luce (Firth et al., 2019)) operate directly on the collection of events $\{(U_{mn}, \mathcal{L} \setminus U_{mn})\}_{m,n}$, discarding degenerate events with $U_{mn} = \emptyset$ or $U_{mn} = \mathcal{L}$. In our binary two-level setting, Plackett–Luce likelihoods collapse to functions of pairwise win counts (cf. the MM formulation for generalized Bradley–Terry and Plackett–Luce likelihoods (Hunter, 2004)), whereas Davidson–Luce explicitly models within-set ties.

2.3 Bayesian Approaches in Ranking

Many ranking methods can be viewed as probabilistic models with latent parameters θ (e.g., model strength and, optionally, question difficulty). Given observations $\mathbf{R}$ (or derived representations such as pairwise counts; Section 2.2), inference reduces to estimating θ from a likelihood $p(\mathbf{R} \mid \theta)$. We consider maximum likelihood estimation (MLE), maximum a posteriori (MAP), and expected a posteriori (EAP), and discuss how uncertainty can be propagated to rankings (Gelman et al., 2013). Although MLE is not Bayesian, we include it as a standard baseline for likelihood-based ranking models.

Maximum likelihood estimation (MLE). The maximum likelihood estimate is

$$\hat{\theta}_{\mathrm{MLE}} \in \arg\max_{\theta} \; p(\mathbf{R} \mid \theta), \tag{5}$$

which yields a point estimate without requiring a prior. MLE is attractive for its simplicity, but in paired-comparison and IRT-like models it can be unstable under (near-)separation or weak identification, which motivates priors in MAP and EAP.

Maximum a posteriori (MAP). MAP incorporates prior information $p(\theta)$ and estimates the posterior mode:

$$\hat{\theta}_{\mathrm{MAP}} \in \arg\max_{\theta} \; p(\mathbf{R} \mid \theta)\, p(\theta). \tag{6}$$

Equivalently, MAP is a penalized MLE in which $-\log p(\theta)$ acts as a regularizer; priors can improve stability in paired-comparison and IRT-style models (Caron and Doucet, 2012; Mislevy, 1986). We can also construct *empirical* priors from auxiliary evaluation runs. For example, a prior outcome tensor $\mathbf{R}_0$ (e.g., one greedy decode per question) can be used to regularize stochastic trials (`EmpiricalPrior` in `Scorio`) (Hariri et al., 2026).

Expected a posteriori (EAP). EAP uses the posterior mean as the point estimate:

$$\hat{\theta}_{\mathrm{EAP}} := \mathbb{E}[\theta \mid \mathbf{R}], \tag{7}$$

which is Bayes-optimal under squared-error loss (Gelman et al., 2013). Compared with MAP, EAP accounts for posterior mass beyond the mode and typically requires approximation or sampling. EAP is common in latent-trait settings such as IRT and adaptive testing (Chen et al., 1998).

Interval estimates and conservative ranking. Bayesian methods naturally yield *credible intervals* (posterior quantiles) for each θ_l, while frequentist analyses can produce approximate *confidence intervals* for $\hat{\theta}_{\mathrm{MLE}}$ via bootstrap resampling of questions or trials. Interval estimates are especially useful because ranking is sensitive to near ties: rather than ranking by point estimates alone, one can rank conservatively using a lower credible or confidence bound (LCB), or report pairwise superiority probabilities $\Pr(\theta_i > \theta_j \mid \mathbf{R})$. Metric-level Bayesian estimators such as Bayes@N provide both a posterior mean and uncertainty, enabling rankings by posterior mean or by a chosen posterior quantile. Bayes@N also supports incorporating prior outcomes $\mathbf{R}_0$ (e.g., one greedy decode per question) as pseudo-counts in the posterior, which is complementary to using $\mathbf{R}_0$ to define empirical priors for MAP in parametric ranking models. Our implementation in `Scorio` supports both credible-interval ranking via Bayes@N and empirical priors via `EmpiricalPrior` for MAP estimation.

3 Experiments

We evaluate 72 ranking methods (Appendix J.2) on four Olympiad-style math benchmarks: AIME'24, AIME'25, HMMT'25, and BrUMO'25, each with $M = 30$ questions. We use $L = 20$ reasoning LLMs (full list in Table 23). For each model–question pair, we collect $N = 80$ independent trials via top-p sampling, yielding a response tensor $\mathbf{R} \in \{0,1\}^{20\times30\times80}$. We also collect a single greedy-decoding output per question ($\mathbf{R}_0$) to serve as an empirical prior. Detailed generation, sampling, and reproducibility settings appear in Appendix I; the library API is documented in Appendix J.

3.1 Gold Standard Ranking

Following Section 2.1, we define the gold-standard ranking as $\mathrm{Bayes}_{\mathcal{U}}@80$, the Bayesian posterior-mean estimator with a uniform prior computed from all $N = 80$ trials. This choice is order-equivalent to avg@80 (mean correctness over all M questions and all $N = 80$ trials, with ties allowed) and yields an interpretable accuracy-based target. Empirically, when each of our 72 ranking methods is computed using all 80 trials, the resulting orderings agree closely with $\mathrm{Bayes}_{\mathcal{U}}@80$ (Table 1): across benchmarks, the average Kendall's τ_b between $\mathrm{Bayes}_{\mathcal{U}}@80$ and the other methods is 0.93–0.95 (median 0.95–0.99), and 19–34 methods recover exactly the same ordering ($\tau_b = 1$). The largest deviations come from a small set of voting rules (e.g., minimax and Nanson variants) and difficulty-weighted baselines, with minimum τ_b values of 0.68–0.79 depending on the benchmark. Although $\mathrm{Bayes}_{\mathcal{U}}@N$ is order-equivalent to avg@N, we prefer the Bayesian formulation because it supports priors (e.g., $\mathrm{Bayes}_{\mathbf{R}_0}@N$) and uncertainty estimates.

3.2 Ranking-Method Stability

To compare ranking methods in the low-budget regime, we set $N = 1$ by subsampling one of the 80 trials per question and recomputing the rankings. For each method, we report Kendall's τ_b averaged over the 80 single-trial draws (mean $\pm$ std). Since

Table 1: Agreement between the gold-standard ranking ($\text{Bayes}_{\mathcal{U}}$@80) and each other ranking method, measured by Kendall's τ_b, when all methods are computed from the full $N = 80$ trials. Statistics are computed over the other 71 methods; "Combined" pools all benchmarks.

Benchmark	Mean	Median	Min	#($\tau_b = 1$)	#($\tau_b \geq 0.95$)
AIME'24	0.941	0.989	0.682	20	40
AIME'25	0.934	0.947	0.771	19	29
HMMT'25	0.950	0.989	0.758	34	44
BrUMO'25	0.954	0.968	0.789	26	49
Combined	0.962	0.989	0.748	22	53

the Pass@k family requires at least two trials to differ from mean accuracy, the $N = 1$ comparisons below cover the remaining 69 methods.

Gold-standard agreement. We first rank methods by agreement with the empirical gold standard ($\text{Bayes}_{\mathcal{U}}$@80). Across AIME'24, AIME'25, and BrUMO'25, $\text{Bayes}_{\mathbf{R}_0}@N$ performs best, achieving $\tau_b = 0.779 \pm 0.034$, 0.798 ± 0.045, and 0.858 ± 0.028, respectively (Table 2). On HMMT'25, the hardest benchmark (see Appendix B), the greedy prior no longer helps, and the best score is shared by a 21-method equivalence class ($\text{Bayes}_{\mathcal{U}}@N$ and several graph- and voting-based methods), with $\tau_b = 0.790 \pm 0.053$. When all benchmarks are pooled (Combined), the same 21-method class attains $\tau_b = 0.865 \pm 0.049$, while $\text{Bayes}_{\mathbf{R}_0}@N$ drops to $\tau_b = 0.786 \pm 0.031$ (Table 18).

Self-consistency and convergence. Next, we evaluate each method against its own full-trial ranking (method@80), which summarizes convergence from $N = 1$ to $N = 80$. Rasch MML with LCB scoring is the most self-consistent on AIME'24, AIME'25, and HMMT'25, with $\tau_b = 0.804 \pm 0.051$, 0.834 ± 0.054, and 0.810 ± 0.056 (Table 2); BrUMO'25 again favors $\text{Bayes}_{\mathbf{R}_0}@N$ (0.858 ± 0.028). On the Combined benchmark, the most self-consistent method is Nanson's rule with tie averaging (0.892 ± 0.050), followed by Rasch MML (LCB) (0.883 ± 0.037), whereas several minimax variants are among the least self-consistent (down to 0.765 ± 0.045; Table 19). High self-consistency does not imply strong agreement with the gold standard: Nanson (avg ties) ranks first in self-consistency on Combined but has substantially lower gold-standard agreement (0.807 ± 0.036; Table 18).

3.3 Bootstrapped Model-Pool Robustness

The preceding $N = 1$ results use the full set of 20 models. To test whether those conclusions depend on the evaluation pool, we repeat the low-budget analysis on bootstrapped model pools of size 5, 10, and 15. For each bootstrap subset, we recompute the full-trial rankings, use the subset-specific avg@80 ordering as the gold-standard target, and compare each method's 80 single-trial rankings against two references: (i) the subset-specific avg@80 ordering and (ii) its own subset-specific full-trial ranking (method@80). We aggregate 1000 bootstrap subsets for each benchmark–size setting.

Easy and medium benchmarks preserve the original winner. On AIME'24, AIME'25, and BrUMO'25, $\text{Bayes}_{\mathbf{R}_0}@N$ remains the best representative method under both targets at all three model-pool sizes (Table 3). The mean score changes only slightly with pool size: on AIME'24, gold-standard agreement moves from 0.769 to 0.780 and self-consistency from 0.773 to 0.785 as the pool size increases from 5 to 15 models; on AIME'25, the corresponding ranges are 0.797–0.802 and 0.803–0.809; on BrUMO'25, $\text{Bayes}_{\mathbf{R}_0}@N$ stays near 0.854–0.858 for both targets. On BrUMO'25, this advantage also becomes more decisive as the pool grows: the fraction of subsets where $\text{Bayes}_{\mathbf{R}_0}@N$ is the top-scoring method rises from about 0.69 at $k = 5$ to 0.98–0.99 at $k = 15$.

Harder benchmarks remain tie-rich. The harder settings behave differently. On HMMT'25 and on the Combined benchmark, the top score is not unique: for agreement with avg@80, an equivalence class of 29–30 methods shares the best mean, while for method@80 the tied class still contains 13–14 methods. We report avg (avg@N, order-equivalent to $\text{Bayes}_{\mathcal{U}}@N$) as a representative member of these tied classes. The tied optimum is essentially flat across pool size, staying near 0.788–0.790 on HMMT'25 and 0.863–0.866 on Combined. This mirrors the full-model analysis in Section 3.2: once the benchmark is difficult or pooled across heterogeneous tasks, many pointwise, voting, and graph-based methods become empirically indistinguishable.

Larger pools mainly reduce between-subset variance. The primary effect of increasing the model-pool size is to reduce dispersion across subsets

Table 2: Best-performing ranking methods in the low-budget regime ($N = 1$) under two targets: (i) agreement with the gold standard ($\text{Bayes}_{\mathcal{U}}$@80) and (ii) self-consistency with each method's own full-trial ranking (method@80). Kendall's τ_b is averaged over 80 single-trial draws; † denotes a 21-way tie for best gold-standard agreement (see Table 18). Pass@k variants are excluded at $N = 1$ because they require $N \geq 2$. Method identifiers correspond to the APIs listed in Section J.2.

Benchmark	Best vs. gold standard	τ_b	Best self-consistency (vs. `method@80`)	τ_b
AIME'24	$\text{Bayes}_{\mathbf{R}_0}$@1	0.779 ± 0.034	Rasch MML LCB (`rasch_mml_credible`)	0.804 ± 0.051
AIME'25	$\text{Bayes}_{\mathbf{R}_0}$@1	0.798 ± 0.045	Rasch MML LCB (`rasch_mml_credible`)	0.834 ± 0.054
HMMT'25	Bayes@1 †	0.790 ± 0.053	Rasch MML LCB (`rasch_mml_credible`)	0.810 ± 0.056
BrUMO'25	$\text{Bayes}_{\mathbf{R}_0}$@1	0.858 ± 0.028	$\text{Bayes}_{\mathbf{R}_0}$@1	0.858 ± 0.028
Combined	Bayes@1 †	0.865 ± 0.049	Nanson avg ties (`nanson_rank_ties_average`)	0.892 ± 0.050

Table 3: Bootstrapped model-pool results in the low-budget regime ($N = 1$). For each model-pool subset, we compute Kendall's τ_b over the 80 single-trial rankings against two targets: the subset-specific gold standard `avg@80` and each method's own subset-specific full-trial ranking (`method@80`). The table reports the mean and standard deviation of the subset-level mean score across bootstrap model pools for each subset size.

Benchmark	Pool	Best Method	τ_b vs Target	
			`avg@80`	`method@80`
AIME'24	5	$\text{Bayes}_{\mathbf{R}_0}$@1	0.769 ± 0.209	0.773 ± 0.207
	10		0.776 ± 0.107	0.781 ± 0.105
	15		0.780 ± 0.057	0.785 ± 0.057
AIME'25	5	$\text{Bayes}_{\mathbf{R}_0}$@1	0.802 ± 0.144	0.809 ± 0.144
	10		0.797 ± 0.071	0.803 ± 0.073
	15		0.798 ± 0.038	0.804 ± 0.040
HMMT'25	5	Bayes@1	0.788 ± 0.114	0.788 ± 0.114
	10		0.789 ± 0.059	0.789 ± 0.059
	15		0.790 ± 0.033	0.790 ± 0.033
BrUMO'25	5	$\text{Bayes}_{\mathbf{R}_0}$@1	0.854 ± 0.136	0.854 ± 0.136
	10		0.856 ± 0.062	0.856 ± 0.062
	15		0.858 ± 0.032	0.858 ± 0.032
Combined	5	Bayes@1	0.863 ± 0.084	0.863 ± 0.084
	10		0.866 ± 0.042	0.866 ± 0.042
	15		0.864 ± 0.023	0.864 ± 0.023

rather than to shift the mean systematically (Table 3). For the best method under the avg@80 target, the across-subset standard deviation falls from 0.209 to 0.057 on AIME'24, from 0.144 to 0.038 on AIME'25, from 0.114 to 0.033 on HMMT'25, from 0.136 to 0.032 on BrUMO'25, and from 0.084 to 0.023 on Combined when moving from 5 to 15 models. Thus, the qualitative recommendation is stable under moderate changes to the model pool: larger pools mainly make the same conclusion more certain.

3.4 Effect of Empirical Priors

Empirical priors use auxiliary evaluation signals to stabilize low-budget rankings. In our setting, the signal is a single greedy decode, $\mathbf{R}_0$. We incorporate $\mathbf{R}_0$ into Bayes@N, yielding $\text{Bayes}_{\mathbf{R}_0}$@$N$, and compare it with the uniform-prior variant $\text{Bayes}_{\mathcal{U}}$@$N$. We evaluate both variants by their agreement with the gold-standard ranking $\text{Bayes}_{\mathcal{U}}$@80. For each N, we compute Kendall's τ_b between the induced model ranking and $\text{Bayes}_{\mathcal{U}}$@80 and report the mean and standard deviation over 50 resampled datasets.

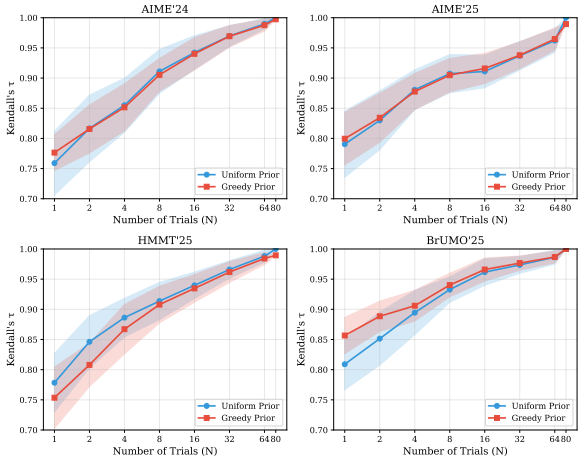

Figure 2: Gold-standard agreement of $\text{Bayes}_{\mathcal{U}}$@$N$ (blue) and $\text{Bayes}_{\mathbf{R}_0}$@$N$ (red) as a function of N across benchmarks. Shaded regions show ± 1 standard deviation over 50 resampled datasets.

Empirical priors reduce variance at low N. Across all benchmarks, $\text{Bayes}_{\mathbf{R}_0}$@$N$ yields more stable low-N rankings than $\text{Bayes}_{\mathcal{U}}$@$N$. At $N = 1$, the standard deviation of τ_b decreases by 16–52% depending on the benchmark (Table 4 and Fig. 7). This advantage shrinks quickly as N increases (Fig. 2), consistent with the prior contributing only $O(1)$ pseudo-counts per question.

The mean effect depends on greedy–sampling alignment. Variance reduction does not guarantee improved agreement with $\text{Bayes}_{\mathcal{U}}$@80. The

Table 4: Dataset difficulty (mean accuracy), greedy–sampling alignment ($\tau_{\text{G-S}}$), and the effect of the greedy empirical prior at $N = 1$. $\Delta\tau$ is the difference in gold-standard agreement (greedy minus uniform), and Std. Red. is the relative reduction in the standard deviation of τ_b.

Benchmark	Difficulty	$\tau_{\text{G-S}}$	$\Delta\tau$	Std. Red.
AIME'24	0.620	0.739	+0.020	42%
AIME'25	0.533	0.660	+0.008	17%
HMMT'25	**0.333**	**0.635**	**−0.022**	16%
BrUMO'25	0.588	**0.768**	**+0.049**	52%

greedy prior increases mean τ_b on AIME'24, AIME'25, and BrUMO'25, but decreases it on HMMT'25 (Table 4). At $N = 1$, when all benchmarks are pooled, this negative shift is substantially larger (Table 18), indicating that an empirical prior can introduce systematic bias when greedy and sampling behave differently across datasets.

We summarize this diagnostic via *greedy–sampling alignment* $\tau_{\text{G-S}}$, defined as Kendall's τ_b between the model rankings induced by greedy decoding and by stochastic sampling at $N = 80$. In our results, higher $\tau_{\text{G-S}}$ coincides with a more positive $\Delta\tau$ (Appendix E and Fig. 6), suggesting that the empirical prior is most likely to help when greedy is a faithful proxy for the sampling-induced ordering. While this evidence is limited to four benchmarks, the trend is consistent with $\text{Bayes}_{\mathbf{R}_0}@N$ acting as shrinkage toward the greedy ordering.

Implications. $\text{Bayes}_{\mathbf{R}_0}@N$ behaves as a shrinkage estimator toward the greedy ordering: it is helpful when greedy decoding is a faithful proxy for the sampling-induced ranking, and harmful when the two disagree. Because $\mathbf{R}_0$ is generated under a different decoding policy, incorporating it effectively biases the estimate toward greedy behavior. This can be desirable for variance reduction, but it changes the implied evaluation target. A plausible source of disagreement is that greedy decoding may under-explore on hard instances, while stochastic sampling can recover alternative successful reasoning paths. In practice, empirical priors are most attractive when N is very small and greedy–sampling alignment has been checked on a small pilot sample; otherwise, $\text{Bayes}_{\mathcal{U}}@N$ provides a safer default.

Bias–variance trade-off. Figure 7 visualizes the trade-off induced by empirical priors: in our benchmarks, the greedy prior reduces variability (narrower distributions) but can introduce bias (shifted means), with the net effect governed by greedy–sampling alignment.

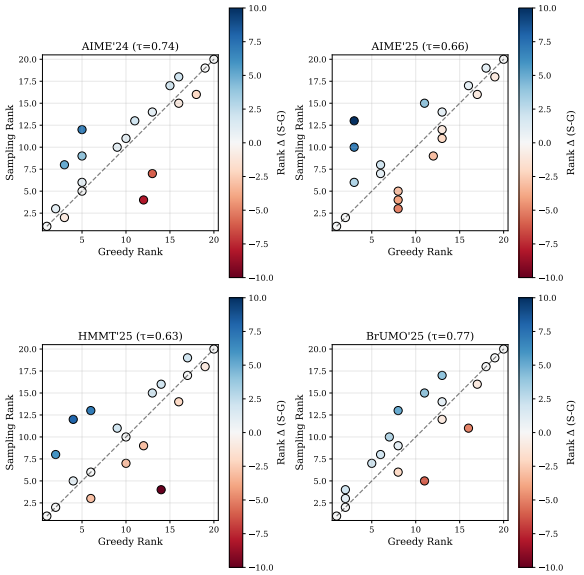

Figure 3: Model-level ranks under greedy decoding versus stochastic sampling ($N = 80$) for each benchmark. Points on the diagonal indicate perfect alignment; color shows rank displacement (Δ).

3.5 Categorical Ranking

We extend the Bayesian framework to *categorical outcomes*: each completion is mapped to one of $C + 1$ ordered categories based on signals such as answer format (boxed vs. unboxed), model confidence (completion bits per token), token efficiency, and external verifier judgments. Each scheme defines a categorical mapping and a utility weight vector $\mathbf{w} = (w_0, \ldots, w_C)$; Bayesian estimation then proceeds with a Dirichlet–multinomial model rather than a Beta–binomial model (details and scheme definitions are given in Appendix F).

We select eight non-redundant representative schemes. Using the $N = 1$ subsampling protocol on the Combined benchmark (the first $L = 11$ models of Table 23, $M = 120$ questions pooled across all four datasets), we measure Kendall's τ_b against three references (Table 5).

Self-consistency vs. gold-standard trade-off. Signal-rich schemes achieve the highest self-consistency: Verifier-only ($\tau_{\text{Self}} = 0.897$) and OOD-robust (0.892) rank first and second (Fig. 4). Yet these schemes have the lowest agreement with the gold standard ($\tau_{\text{GS}} = 0.824$ and 0.840, respec-

Table 5: Categorical ranking at $N=1$ on the combined benchmark ($L=11$, the first 11 models from Table 23, $M=120$). Eight representative schemes are ordered by agreement with the gold standard (τ_{GS}, vs. $\text{Bayes}_{\mathcal{U}}@80$). Self: τ_b vs. Scheme@80; Greedy: τ_b vs. $\text{Bayes}_{\mathbf{R}_0}@80$. Values are mean $\pm$ std over 80 draws.

Scheme	τ_{GS}	τ_{Self}	τ_{Greedy}
Conservative	0.856 ± 0.076	0.861 ± 0.066	0.858 ± 0.074
Efficiency-adj.	0.850 ± 0.070	0.875 ± 0.057	0.859 ± 0.071
Format-aware	0.849 ± 0.071	0.881 ± 0.064	0.869 ± 0.069
Balanced comp.	0.843 ± 0.075	0.877 ± 0.067	0.862 ± 0.073
OOD-robust	0.840 ± 0.071	0.892 ± 0.063	0.870 ± 0.066
Rare-event	0.838 ± 0.073	0.888 ± 0.065	0.867 ± 0.069
Verifier-calib.	0.832 ± 0.076	0.877 ± 0.067	0.855 ± 0.073
Verifier-only	0.824 ± 0.071	0.897 ± 0.068	0.870 ± 0.071

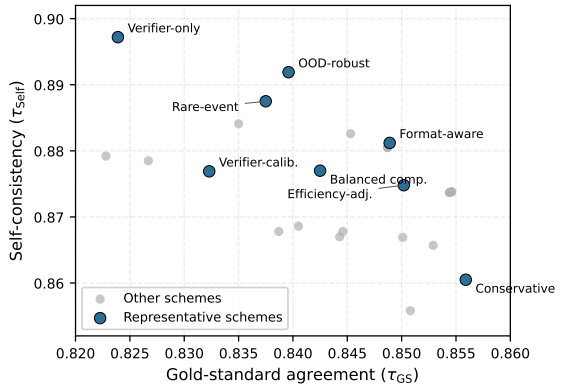

Figure 4: Gold-standard agreement vs. self-consistency for 25 categorical schemes at $N = 1$ on the Combined benchmark. Blue markers indicate the 8 representative schemes; gray markers show the remaining 17. Schemes in the upper-left are self-consistent but deviate from $\text{Bayes}_{\mathcal{U}}@80$; those in the lower-right closely track the gold standard but are less stable across single-trial draws.

tively), extending the finding from Section 3.2 that high self-consistency does not imply closeness to the gold standard. The negative correlation between τ_{GS} and τ_{Self} across schemes (Fig. 4) suggests that auxiliary signals introduce systematic biases away from the correctness-based ordering while stabilizing single-trial rankings.

Greedy-prior alignment. All eight schemes correlate more strongly with $\text{Bayes}_{\mathbf{R}_0}@80$ than with $\text{Bayes}_{\mathcal{U}}@80$; the gap is largest for Verifier-only ($\Delta\tau = +0.046$) and OOD-robust ($+0.031$), consistent with the mechanism in Section 3.4: verifier and OOD signals encode information partially aligned with greedy-decoding behavior. Per-dataset results (Appendix F) show that scheme differentiation widens on harder benchmarks (HMMT'25, BrUMO'25), where Verifier-only drops to $\tau_{\text{GS}} = 0.753$ and 0.734, while correctness-driven schemes remain stable ($\tau_{\text{GS}} \geq 0.80$).

4 Related Work

Test-time scaling samples multiple solutions per prompt and aggregates them (Wang et al., 2023; Snell et al., 2024; Zeng et al., 2025); because stochastic reasoning varies across runs (Liu et al., 2025), we study how this variability affects *rankings* as budget changes. Preference evaluation and alignment learn from paired comparisons (Christiano et al., 2017; Rafailov et al., 2023) and underpin leaderboards such as Chatbot Arena (Chiang et al., 2024; Ameli et al., 2025). Benchmark leaderboards often rank models by task metrics such as Pass@k (Chen et al., 2021), but item-level difficulty and discrimination affect reliability (Rodriguez et al., 2021); recent work adds Bayesian uncertainty and IRT-style modeling (Hariri et al., 2026; Zhou et al., 2025). We extend this literature to dense repeated-trial benchmarks and compare ranking methods by stability and convergence; Appendix H gives background.

5 Conclusion & Future Directions

Test-time scaling turns LLM benchmarking into a repeated-sampling problem, so model rankings must be estimated from stochastic trials rather than from a single run. We formalize this setting and compare a broad collection of ranking methods within a common framework. When many trials are available, most reasonable ranking families induce nearly identical orderings, making $\text{Bayes}_{\mathcal{U}}@N$ a simple and interpretable default. The main differences appear in the low-budget regime. There, uncertainty-aware estimators can improve stability, and the greedy prior $\text{Bayes}_{\mathbf{R}_0}@N$ acts as a shrinkage estimator: it reduces variance when greedy and stochastic sampling align, but can bias rankings when they diverge.

In practice, $\text{Bayes}_{\mathcal{U}}@N$ is a strong default, whereas $\text{Bayes}_{\mathbf{R}_0}@N$ is best used after checking greedy–sampling alignment on a small pilot sample. Our experiments focus on binary correctness; extending the analysis to partial credit, rubric-based scoring, and other categorical evaluation settings is a natural next step.

Limitations

Our experiments focus on mathematical reasoning benchmarks. We do not evaluate partial credit, or open-ended outputs, where outcome categories are less clear and annotation or verification noise may be larger. More generally, when informative priors are used—especially priors derived from auxiliary signals other than greedy decoding—the prior source and specification should be reported explicitly, since the prior can introduce systematic bias if it is misaligned with the stochastic evaluation regime.

Acknowledgments

This research was supported in part by NSF awards 2117439 and 2320952.

References

Marah Abdin, Sahaj Agarwal, Ahmed Awadallah, Vidhisha Balachandran, Harkirat Behl, Lingjiao Chen, Gustavo de Rosa, Suriya Gunasekar, Mojan Javaheripi, Neel Joshi, Piero Kauffmann, Yash Lara, Caio César Teodoro Mendes, Arindam Mitra, Besmira Nushi, Dimitris Papailiopoulos, Olli Saarikivi, Shital Shah, Vaishnavi Shrivastava, and 4 others. 2025. Phi-4-reasoning technical report. *Preprint*, arXiv:2504.21318.

Siavash Ameli, Siyuan Zhuang, Ion Stoica, and Michael W. Mahoney. 2025. A statistical framework for ranking LLM-based chatbots. In *International Conference on Learning Representations*.

Kenneth J. Arrow. 1951. *Social Choice and Individual Values*. John Wiley & Sons, New York.

David Balduzzi, Marta Garnelo, Yoram Bachrach, Wojciech Czarnecki, Julien Pérolat, Max Jaderberg, and Thore Graepel. 2019. Open-ended learning in symmetric zero-sum games. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 434–443. PMLR.

J. M. Baldwin. 1926. The technique of the nanson preferential majority system of election. *Proceedings of the Royal Society of Victoria, New Series*, 39(1):42–52.

Michel Balinski and Rida Laraki. 2011. *Majority Judgment: Measuring, Ranking, and Electing*. The MIT Press.

Bespoke Labs. 2025. Bespoke-stratos: The unreasonable effectiveness of reasoning distillation. Accessed: 2025-01-22.

Allan Birnbaum. 1968. Some latent trait models and their use in inferring an examinee's ability. In Frederic M. Lord and Melvin R. Novick, editors, *Statistical Theories of Mental Test Scores*, pages 396–479. Addison-Wesley, Reading, MA.

R. Darrell Bock and Murray Aitkin. 1981. Marginal maximum likelihood estimation of item parameters: Application of an EM algorithm. *Psychometrika*, 46(4):443–459.

Ralph Allan Bradley and Milton E. Terry. 1952. Rank analysis of incomplete block designs: The method of paired comparisons. *Biometrika*, 39(3-4):324–345.

Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D. Procaccia, editors. 2016. *Handbook of Computational Social Choice*. Cambridge University Press.

Brown University Math Olympiad Organizers. 2025. Brown university math olympiad (BrUMO). Official BrUMO website with tournament information (Apr 4–5, 2025); accessed 2025-09-25.

François Caron and Arnaud Doucet. 2012. Efficient bayesian inference for generalized bradley–terry models. *Journal of Computational and Graphical Statistics*, 21(1):174–196.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, and 39 others. 2021. Evaluating large language models trained on code. *Preprint*, arXiv:2107.03374.

Ssu-Kuang Chen, Liling Hou, and Barbara G. Dodd. 1998. A comparison of maximum likelihood estimation and expected a posteriori estimation in CAT using the partial credit model. *Educational and Psychological Measurement*, 58(4):569–595.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Banghua Zhu, Hao Zhang, Michael Jordan, Joseph E. Gonzalez, and Ion Stoica. 2024. Chatbot arena: An open platform for evaluating LLMs by human preference. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 8359–8388. PMLR.

Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems*, volume 30, pages 4299–4307.

Marquis de Condorcet, Marie Jean Antoine Nicolas Caritat. 1785. *Essai sur l'application de l'analyse 'a la probabilité des décisions rendues 'a la pluralité des voix*. Imprimerie Royale, Paris.

Arthur H. Copeland. 1951. A reasonable social welfare function. Seminar on Applications of Mathematics to Social Sciences. University of Michigan, Ann Arbor. Mimeographed notes.

Tri Dao. 2023. Flashattention-2: Faster attention with better parallelism and work partitioning. *arXiv preprint arXiv:2307.08691*.

Roger R. Davidson. 1970. On extending the bradley–terry model to accommodate ties in paired comparison experiments. *Journal of the American Statistical Association*, 65(329):317–328.

Paul De Boeck and Mark Wilson, editors. 2004. *Explanatory Item Response Models*. Springer.

Jean-Charles de Borda. 1781. Mémoire sur les élections au scrutin. Histoire de l'Académie Royale des Sciences, Paris. Often cited as appearing in the 1781 volume (issued in 1784) of the Histoire/Mémoires of the Académie.

Arpad E. Elo. 1978. *The Rating of Chessplayers, Past and Present*. Arco Publishing.

David Firth, Ioannis Kosmidis, and Heather Turner. 2019. Davidson–luce model for multi-item choice with ties. *Preprint*, arXiv:1909.07123.

Fajwel Fogel, Alexandre d'Aspremont, and Milan Vojnovic. 2016. Spectral ranking using seriation. *Journal of Machine Learning Research*, 17:88:1–88:45.

FuseAI. 2025. FuseO1-DeepSeekR1-QwQ-SkyT1-Flash-32B-Preview. Model card; accessed 2026-03-09.

Andrew Gelman, John B. Carlin, Hal S. Stern, David B. Dunson, Aki Vehtari, and Donald B. Rubin. 2013. *Bayesian Data Analysis*, 3 edition. CRC Press.

Mark E. Glickman. 1999. Parameter estimation in large dynamic paired comparison experiments. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 48(3):377–394.

Takumi Gotou, Ryo Nagata, Masato Mita, and Kazuaki Hanawa. 2020. Taking the correction difficulty into account in grammatical error correction evaluation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2085–2095.

Sylvain Gugger, Lysandre Debut, Thomas Wolf, Philipp Schmid, Zachary Mueller, Sourab Mangrulkar, Marc Sun, and Benjamin Bossan. 2022. Accelerate: Training and inference at scale made simple, efficient and adaptable.

Etash Guha, Ryan Marten, Sedrick Keh, Negin Raoof, Georgios Smyrnis, Hritik Bansal, Marianna Nezhurina, Jean Mercat, Trung Vu, Zayne Sprague, Ashima Suvarna, Benjamin Feuer, Liangyu Chen, Zaid Khan, Eric Frankel, Sachin Grover, Caroline Choi, Niklas Muennighoff, Shiye Su, and 31 others. 2025. OpenThoughts: Data Recipes for Reasoning Models. *Preprint*, arXiv:2506.04178.

D. Guo, D. Yang, H. Zhang, J. Song, P. Wang, Q. Zhu, R. Xu, R. Zhang, S. Ma, X. Bi, X. Zhang, X. Yu, Y. Wu, Z. F. Wu, Z. Gou, Z. Shao, Z. Li, Z. Gao, A. Liu, and 175 others. 2025. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638.

Mohsen Hariri, Amirhossein Samandar, Michael Hinczewski, and Vipin Chaudhary. 2026. Don't pass@k: A bayesian framework for large language model evaluation. In *Proceedings of the 14th International Conference on Learning Representations (ICLR 2026)*.

Harvard–MIT Mathematics Tournament. 2025. Hmmt february 2025 archive (problems and solutions). Official HMMT archive page for February 2025 competition; accessed 2025-09-25.

W. K. Hastings. 1970. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109.

Ralf Herbrich, Tom Minka, and Thore Graepel. 2006. TrueSkill: A bayesian skill rating system. In *Advances in Neural Information Processing Systems*, volume 19, pages 569–576. MIT Press.

Hugging Face. 2025. Open-R1: A fully open reproduction of DeepSeek-R1.

David R. Hunter. 2004. MM algorithms for generalized bradley–terry models. *The Annals of Statistics*, 32(1):384–406.

Xiaoye Jiang, Lek-Heng Lim, Yuan Yao, and Yinyu Ye. 2011. Statistical ranking and combinatorial hodge theory. *Mathematical Programming*, 127(1):203–244.

John G. Kemeny. 1959. Mathematics without numbers. *Daedalus*, 88(4):577–591.

M. G. Kendall. 1938. A new measure of rank correlation. *Biometrika*, 30(1-2):81–93.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, pages 611–626. ACM.

LG AI Research. 2025. EXAONE 4.0: Unified large language models integrating non-reasoning and reasoning modes. *Preprint*, arXiv:2507.11407.

Junnan Liu, Hongwei Liu, Linchen Xiao, Ziyi Wang, Kuikun Liu, Songyang Gao, Wenwei Zhang, Songyang Zhang, and Kai Chen. 2025. Are your LLMs capable of stable reasoning? In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 17594–17632. Association for Computational Linguistics.

Zihan Liu, Zhuolin Yang, Yang Chen, Chankyu Lee, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. 2026. Acereason-nemotron 1.1: Advancing math and code reasoning through SFT and RL synergy. In *International Conference on Learning Representations*.

R. Duncan Luce. 1959. *Individual Choice Behavior: A Theoretical Analysis*. John Wiley & Sons.

Mathematical Association of America. 2024. American invitational mathematics examination (AIME). Official MAA page for the AIME competition (covers AIME 2024); accessed 2025-09-25.

Mathematical Association of America. 2025. American invitational mathematics examination (AIME). Official MAA page for the AIME competition (covers AIME 2025); accessed 2025-09-25.

P. McCullagh and J. A. Nelder. 1989. *Generalized Linear Models*. Springer.

Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. 1953. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092.

Robert J. Mislevy. 1986. Bayes modal estimation in item response models. *Psychometrika*, 51(2):177–195.

E. J. Nanson. 1883. Methods of election. *Transactions and Proceedings of the Royal Society of Victoria*, 19:197–240. Often cited as 1882 in secondary sources.

Sahand Negahban, Sewoong Oh, and Devavrat Shah. 2017. Rank centrality: Ranking from pairwise comparisons. *Operations Research*, 65(1):266–287.

NovaSky Team. 2025. Think less, achieve more: Cut reasoning costs by 50% without sacrificing accuracy. Accessed: 2025-01-23.

NVIDIA. 2025a. NVIDIA nemotron nano 2: An accurate and efficient hybrid mamba-transformer reasoning model. *Preprint*, arXiv:2508.14444.

NVIDIA. 2025b. OpenReasoning-Nemotron-1.5B. Model card; accessed 2026-03-09.

Shayegan Omidshafiei, Christos Papadimitriou, Georgios Piliouras, Karl Tuyls, Mark Rowland, Jean-Baptiste Lespiau, Wojciech M. Czarnecki, Julien Pérolat, and Rémi Munos. 2019. α-rank: Multi-agent evaluation by evolution. *Scientific Reports*, 9(1):9937.

OpenAI. 2025. gpt-oss-120b & gpt-oss-20b Model Card. *Preprint*, arXiv:2508.10925.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab. Previous number: SIDL-WP-1999-0120.

R. L. Plackett. 1975. The analysis of permutations. *Applied Statistics*, 24(2):193–202.

Qwen Team. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, volume 36, pages 53728–53741.

P. V. Rao and L. L. Kupper. 1967. Ties in paired-comparison experiments: A generalization of the bradley–terry model. *Journal of the American Statistical Association*, 62(317):194–204.

Georg Rasch. 1960. *Probabilistic Models for Some Intelligence and Attainment Tests*. Danish Institute for Educational Research, Copenhagen.

Pedro Rodriguez, Joe Barrow, Alexander Hoyle, John P. Lalor, Robin Jia, and Jordan Boyd-Graber. 2021. Evaluation examples are not equally informative: How should that change NLP leaderboards? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4486–4503, Online. Association for Computational Linguistics.

Daniel J. Russo, Benjamin Van Roy, Abbas Kazerouni, Ian Osband, and Zheng Wen. 2018. A tutorial on thompson sampling. *Foundations and Trends in Machine Learning*, 11(1):1–96.

Markus Schulze. 2011. A new monotonic, clone-independent, reversal symmetric, and Condorcet-consistent single-winner election method. *Social Choice and Welfare*, 36(2):267–303.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. 2024. Scaling LLM test-time compute optimally can be more effective than scaling model parameters. *Preprint*, arXiv:2408.03314.

William R. Thompson. 1933. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3-4):285–294.

T. N. Tideman. 1987. Independence of clones as a criterion for voting rules. *Social Choice and Welfare*, 4(3):185–206.

Norman D. Verhelst and Cees A. W. Glas. 1993. A dynamic generalization of the rasch model. *Psychometrika*, 58(3):395–415.

Chun Wang and Steven W. Nydick. 2020. On longitudinal item response theory models: A didactic. *Journal of Educational and Behavioral Statistics*, 45(3):339–368.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations*.

Liang Wen, Yunke Cai, Fenrui Xiao, Xin He, Qi An, Zhenyu Duan, Yimin Du, Junchen Liu, Lifu Tang, Xiaowei Lv, Haosheng Zou, Yongchao Deng, Shousheng Jia, and Xiangzheng Zhang. 2025. Light-r1: Curriculum SFT, DPO and RL for long COT from scratch and beyond. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track)*, pages 318–327. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and 1 others. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. 2025. τ-bench: A benchmark for tool-agent-user interaction in real-world domains. In *International Conference on Learning Representations*.

Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. 2025. LIMO: Less is more for reasoning. In *Second Conference on Language Modeling*.

H. P. Young. 1977. Extending Condorcet's rule. *Journal of Economic Theory*, 16(2):335–353.

Zhiyuan Zeng, Qingyuan Chen, Zhangyue Yin, Yunhua Zhou, and Xipeng Qiu. 2025. Revisiting the test-time scaling of o1-like models: Do they truly possess test-time scaling capabilities? In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4651–4665.

Tianyi Zhang, Mohsen Hariri, Shaochen Zhong, Vipin Chaudhary, Yang Sui, Xia Hu, and Anshumali Shrivastava. 2025. 70% size, 100% accuracy: Lossless llm compression for efficient gpu inference via dynamic-length float (DFloat11). In *Advances in Neural Information Processing Systems*.

Hongli Zhou, Hui Huang, Ziqing Zhao, Lvyuan Han, Huicheng Wang, Kehai Chen, Muyun Yang, Wei Bao, Jian Dong, Bing Xu, Conghui Zhu, Hailong Cao, and Tiejun Zhao. 2025. Lost in benchmarks? rethinking large language model benchmarking with item response theory. *Preprint*, arXiv:2505.15055.

Contents

A Notation and Definitions

The following notation is used throughout the paper.

A.1 Data and Basic Quantities

- L: number of models being ranked.
- M: number of questions in a benchmark.
- N: number of independent stochastic trials per model–question pair under test-time scaling.
- $\mathbf{R} \in \{0,1\}^{L\times M\times N}$: response tensor, where $R_{lmn} = 1$ if model l solves question m on trial n.
- $\mathbf{R}_0$: optional prior outcomes used by Bayesian estimators. In this paper, greedy decoding yields a shared prior matrix $\mathbf{R}_0 \in \{0,1\}^{M\times D}$ with $D = 1$, but the notation can also accommodate model-specific prior tensors.
- $\widehat{p}_{lm} := \frac{1}{N}\sum_{n=1}^{N} R_{lmn}$: per-question solve rate for model l on question m.
- $k_{lm} := \sum_{n=1}^{N} R_{lmn}$: number of successful trials for model l on question m.

A.2 Metric Shorthand

- **Bayes@N**: Bayesian posterior-mean estimate at N trials under a specified prior.
- $\mathrm{Bayes}_{\mathcal{U}}@N$: Bayesian estimate with a uniform Dirichlet prior, denoted $\mathrm{Bayes}_{\mathcal{U}}@N$.
- $\mathrm{Bayes}_{\mathbf{R}_0}@N$: Bayesian estimate with a greedy empirical prior, denoted $\mathrm{Bayes}_{\mathbf{R}_0}@N$.
- **Pass@k**: probability that at least one of k sampled completions is correct.
- **avg@N**: mean accuracy over all M questions and N trials. For binary outcomes, it is order-equivalent to $\mathrm{Bayes}_{\mathcal{U}}@N$.

A.3 Ranking-Method Families

- **Pointwise methods**: aggregate per-question performance to produce model scores (e.g., mean accuracy, inverse-difficulty weighting).
- **Pairwise methods**: transform outcomes into win/tie counts between model pairs and fit paired-comparison models (e.g., Bradley–Terry, Elo, Glicko).
- **Listwise, setwise methods**: operate on winner and loser sets for each question–trial (e.g., Plackett–Luce, Davidson–Luce).
- **Voting rules**: treat questions as voters that rank models and then aggregate those preferences (e.g., Borda, Copeland, Schulze, Kemeny–Young).
- **Graph/spectral methods**: construct comparison graphs and compute centrality- or flow-based scores (e.g., PageRank, Rank Centrality, HodgeRank, α-Rank).
- **IRT-inspired methods**: estimate latent model abilities and item difficulties (e.g., Rasch, 2PL, 3PL, dynamic IRT).

A.4 Evaluation Criteria

- **Kendall's τ_b**: rank-correlation coefficient that accounts for ties; it ranges from -1 (perfect disagreement) to $+1$ (perfect agreement).
- **Gold-standard agreement**: agreement between a low-budget ranking and the empirical gold standard, typically $\mathrm{Bayes}_{\mathcal{U}}@80$ in this paper.
- **Self-consistency**: agreement between a low-budget ranking and the same method's all-trial ranking.
- **Convergence**: the rate at which a method's ranking approaches its full-trial ordering as the number of trials increases.
- **Greedy–sampling alignment** ($\tau_{\text{G-S}}$): Kendall's τ_b between the ranking induced by greedy decoding and the ranking induced by stochastic sampling at high budget.

A.5 Inference Terminology

- **MLE** (maximum likelihood estimation): point estimate that maximizes $p(\mathbf{R} \mid \theta)$.
- **MAP** (maximum a posteriori): point estimate that maximizes $p(\mathbf{R} \mid \theta)p(\theta)$.
- **EAP** (expected a posteriori): posterior mean estimate $\mathbb{E}[\theta \mid \mathbf{R}]$.
- **MML** (marginal maximum likelihood): likelihood-based estimation that integrates over a latent population distribution, commonly used in IRT.

- **Credible intervals (CrI)**: Bayesian posterior intervals used for uncertainty quantification; we use lower credible bounds (LCBs) for conservative ranking.

B Accuracy of Models

Tables 6 to 9 report detailed accuracy statistics for all $L = 20$ models, including greedy accuracy and stochastic-sampling statistics (minimum, mean, maximum, and standard deviation) over $N = 80$ trials. HMMT'25 is the most difficult benchmark (mean accuracies 0.080–0.554), whereas AIME'24 and BrUMO'25 are less difficult. Figure 5 visualizes these distributions across benchmarks and highlights the heterogeneity in model performance and sampling variance that motivates our ranking-stability analysis.

C Gold Standard Agreement

To justify our use of $\mathrm{Bayes}_{\mathcal{U}}@80$ as the gold standard, we compare the full-trial rankings produced by all methods at $N = 80$. Table 1 summarizes Kendall's τ_b between $\mathrm{Bayes}_{\mathcal{U}}@80$ and each competing method. These results indicate that $\mathrm{Bayes}_{\mathcal{U}}@80$ is also a high-consensus ordering: by average agreement with all other methods, it ranks first on AIME'25, HMMT'25, and the Combined benchmark and second on AIME'24 and BrUMO'25 within 5×10^{-4} of the best (Table 10). Dataset-level consensus tables appear in Tables 12 to 16. Many methods recover the same ordering exactly (Table 17), and the remaining disagreement is concentrated in a small low-agreement tail (Table 11).

C.1 Convergence of Ranking Methods

As the number of trials N (or questions M) increases, evaluation metrics such as avg@N, Bayes@N, and Pass@k need not induce the same limiting ordering as ranking methods. The reason is that they target different population quantities.

We illustrate this distinction for two canonical choices used throughout the paper: the average-accuracy ranking and the Bradley–Terry (BT) model.

C.2 Large-Budget Limits: Each Method Converges, but Generally to a Different Target

To discuss $M \to \infty$ (or $N \to \infty$) formally, we introduce an i.i.d. sampling model at the level of question–trial pairs. Assume $(X_{mn})_{m\in[M],n\in[N]}$ are i.i.d. draws from some distribution P on $\{0,1\}^L$. Let

$$p_\ell := \mathbb{P}_{X\sim P}(X_\ell = 1)$$
$$w_{ij} := \mathbb{P}_{X\sim P}(X_i = 1,\ X_j = 0).$$

Here, p_ℓ depends only on the marginal of model ℓ, whereas w_{ij} depends on the *joint* distribution of (X_i, X_j).

Average targets marginal accuracy. By the law of large numbers,

$$\widehat{p}_\ell^{\,\mathrm{avg}}(R) \xrightarrow[MN\to\infty]{\text{a.s.}} p_\ell.$$

Likewise, $\mathrm{Bayes}_{\mathcal{U}}@N$ converges to the same p_ℓ; for binary outcomes it differs from $\widehat{p}_\ell^{\,\mathrm{avg}}$ only by $O((MN)^{-1})$ smoothing.

Bradley–Terry targets a pairwise decisive-win functional. The empirical win frequencies converge:

$$\frac{1}{MN} W_{ij}(R) \xrightarrow[MN\to\infty]{\text{a.s.}} w_{ij}.$$

Define the BT log-likelihood

$$\ell(\pi; W) := \sum_{i\neq j} W_{ij} \Big(\log \pi_i - \log(\pi_i + \pi_j) \Big). \quad (8)$$

Then the BT-ML estimator is an M-estimator: maximizing (8) with W_{ij} is equivalent to maximizing the scaled objective $(MN)^{-1}\ell(\pi; W)$. Under mild regularity and connectivity conditions (ensuring strict concavity in $\log \pi$ and uniqueness up to scale), $\widehat{\pi}$ converges to the unique (up to scale) maximizer of the *population* objective

$$\pi^\star \in \arg\max_{\pi>0} \sum_{i\neq j} w_{ij} \Big(\log \pi_i - \log(\pi_i + \pi_j) \Big). \quad (9)$$

The limiting objects $(p_\ell)_{\ell=1}^L$ and $\pi^\star$ are generally *not* linked by any monotone transform: p_ℓ depends only on marginal correctness, while $\pi^\star$ depends on the full matrix $(w_{ij})_{i\neq j}$. Therefore, without additional assumptions on P (e.g., that P is generated by a BT choice model at the level of decisive comparisons), there is no reason to expect the induced orderings to coincide as $MN \to \infty$. The following counterexample demonstrates this non-equivalence.

Table 6: Accuracy on AIME'24.

Model	Greedy	Top-p			
	Acc.	Min	Mean	Max	Std
DS-R1-Qwen	0.200	0.167	0.297	0.433	0.055
LIMO-v2	0.600	0.467	0.619	0.733	0.059
OpenThinker2	0.767	0.600	0.722	0.833	0.048
OpenThinker3	0.333	0.400	0.517	0.667	0.059
Qwen3-Thinking	0.867	0.767	0.875	0.933	0.038
Sky-T1-Flash	0.400	0.167	0.310	0.400	0.050
gpt-oss-high	0.700	0.633	0.747	0.833	0.053
gpt-oss-low	0.700	0.333	0.675	0.867	0.130
gpt-oss-medium	0.800	0.533	0.755	0.867	0.054
EXAONE-4.0	0.500	0.433	0.570	0.733	0.055
OR-Nemotron	0.433	0.367	0.490	0.667	0.064
Phi-4	0.667	0.567	0.705	0.800	0.050
Phi-4-plus	0.533	0.633	0.753	0.867	0.049
OR1-Distill	0.400	0.400	0.547	0.700	0.066
FuseO1-DS-QwQ-SkyT1	0.500	0.633	0.728	0.800	0.042
Light-R1-DS	0.700	0.600	0.734	0.833	0.060
AR-Nemotron	0.700	0.600	0.709	0.800	0.043
NVIDIA-Nemotron	0.633	0.567	0.676	0.833	0.059
Qwen3-4B	0.767	0.667	0.772	0.900	0.052
Bespoke	0.167	0.100	0.197	0.267	0.043

Table 7: Accuracy on HMMT'25.

Model	Greedy	Top-p			
	Acc.	Min	Mean	Max	Std
DS-R1-Qwen	0.133	0.067	0.135	0.233	0.040
LIMO-v2	0.433	0.233	0.347	0.467	0.048
OpenThinker2	0.333	0.233	0.382	0.500	0.057
OpenThinker3	0.200	0.200	0.297	0.467	0.047
Qwen3-Thinking	0.500	0.467	0.554	0.633	0.037
Sky-T1-Flash	0.167	0.033	0.106	0.200	0.034
gpt-oss-high	0.233	0.267	0.449	0.633	0.069
gpt-oss-low	0.167	0.100	0.203	0.333	0.051
gpt-oss-medium	0.400	0.333	0.455	0.600	0.056
EXAONE-4.0	0.400	0.200	0.335	0.433	0.060
OR-Nemotron	0.267	0.167	0.283	0.400	0.049
Phi-4	0.467	0.267	0.378	0.533	0.056
Phi-4-plus	0.433	0.333	0.447	0.633	0.056
OR1-Distill	0.233	0.133	0.251	0.333	0.042
FuseO1-DS-QwQ-SkyT1	0.300	0.233	0.363	0.467	0.045
Light-R1-DS	0.367	0.233	0.356	0.433	0.045
AR-Nemotron	0.400	0.333	0.408	0.500	0.042
NVIDIA-Nemotron	0.333	0.267	0.362	0.467	0.048
Qwen3-4B	0.467	0.367	0.464	0.567	0.046
Bespoke	0.000	0.000	0.080	0.167	0.035

Table 8: Accuracy on AIME'25.

Model	Greedy	Top-p			
	Acc.	Min	Mean	Max	Std
DS-R1-Qwen	0.133	0.133	0.236	0.333	0.046
LIMO-v2	0.633	0.333	0.541	0.700	0.068
OpenThinker2	0.500	0.467	0.595	0.733	0.060
OpenThinker3	0.367	0.333	0.425	0.600	0.057
Qwen3-Thinking	0.733	0.733	0.804	0.900	0.037
Sky-T1-Flash	0.267	0.133	0.220	0.333	0.041
gpt-oss-high	0.567	0.467	0.690	0.833	0.063
gpt-oss-low	0.600	0.267	0.598	0.800	0.145
gpt-oss-medium	0.567	0.500	0.689	0.833	0.065
EXAONE-4.0	0.467	0.300	0.441	0.567	0.054
OR-Nemotron	0.433	0.300	0.425	0.533	0.054
Phi-4	0.600	0.400	0.599	0.767	0.072
Phi-4-plus	0.567	0.533	0.683	0.800	0.058
OR1-Distill	0.533	0.300	0.426	0.567	0.059
FuseO1-DS-QwQ-SkyT1	0.467	0.433	0.585	0.733	0.064
Light-R1-DS	0.633	0.467	0.589	0.700	0.056
AR-Nemotron	0.633	0.567	0.651	0.733	0.045
NVIDIA-Nemotron	0.467	0.433	0.546	0.667	0.050
Qwen3-4B	0.700	0.600	0.729	0.800	0.044
Bespoke	0.100	0.067	0.193	0.300	0.050

Table 9: Accuracy on BrUMO'25.

Model	Greedy	Top-p			
	Acc.	Min	Mean	Max	Std
DS-R1-Qwen	0.267	0.167	0.344	0.500	0.062
LIMO-v2	0.567	0.500	0.651	0.800	0.065
OpenThinker2	0.767	0.600	0.738	0.900	0.061
OpenThinker3	0.500	0.400	0.512	0.667	0.055
Qwen3-Thinking	0.867	0.733	0.838	0.933	0.038
Sky-T1-Flash	0.333	0.233	0.372	0.500	0.059
gpt-oss-high	0.433	0.533	0.628	0.767	0.053
gpt-oss-low	0.500	0.300	0.393	0.500	0.053
gpt-oss-medium	0.500	0.500	0.610	0.733	0.052
EXAONE-4.0	0.533	0.333	0.484	0.633	0.059
OR-Nemotron	0.400	0.333	0.469	0.600	0.054
Phi-4	0.733	0.533	0.692	0.800	0.052
Phi-4-plus	0.533	0.533	0.711	0.800	0.048
OR1-Distill	0.567	0.367	0.538	0.667	0.057
FuseO1-DS-QwQ-SkyT1	0.567	0.567	0.710	0.900	0.056
Light-R1-DS	0.700	0.600	0.690	0.833	0.049
AR-Nemotron	0.767	0.633	0.714	0.867	0.044
NVIDIA-Nemotron	0.633	0.533	0.649	0.800	0.048
Qwen3-4B	0.767	0.633	0.744	0.833	0.049
Bespoke	0.167	0.167	0.265	0.367	0.053

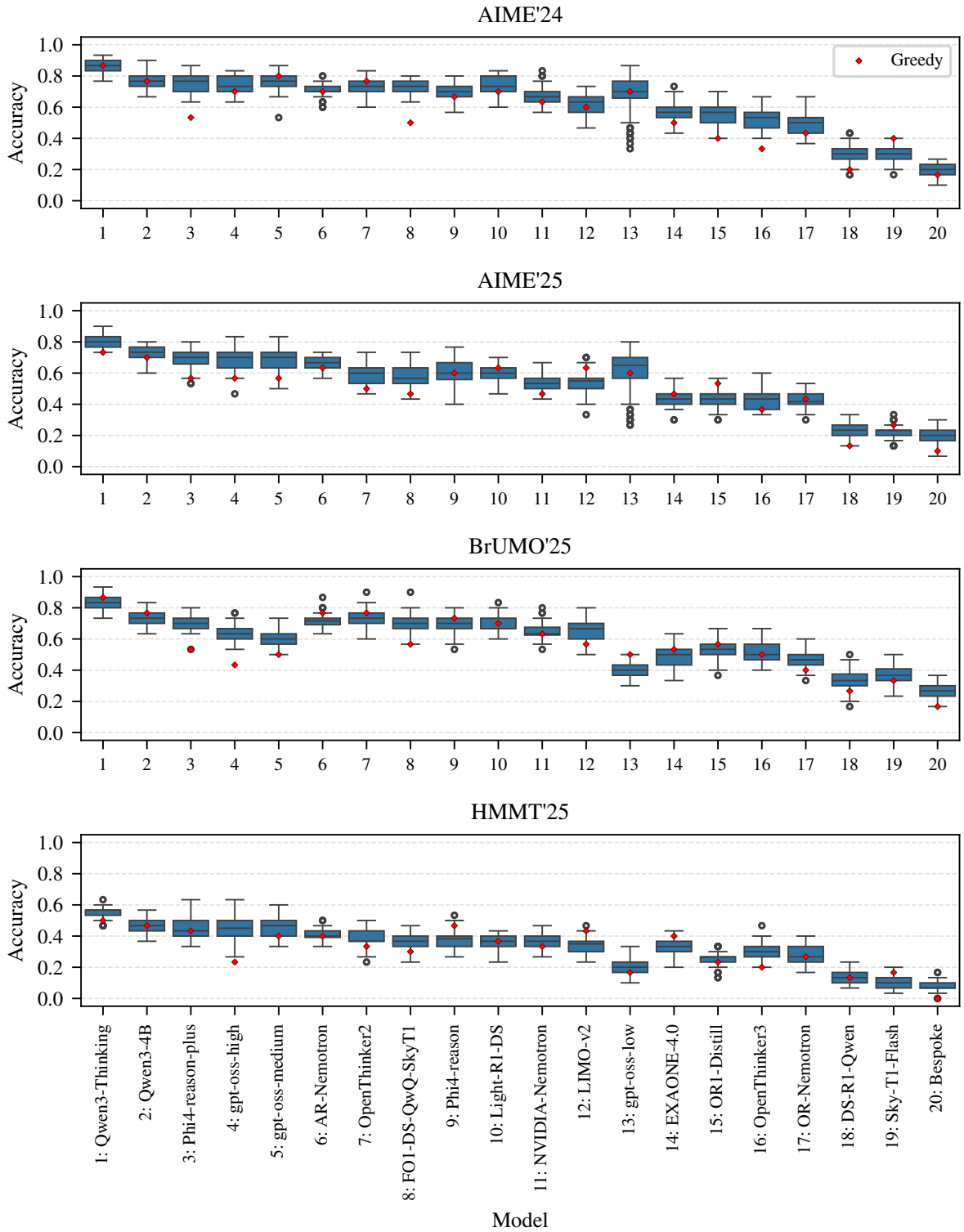

Figure 5: Overview of model accuracies across all four benchmarks. Each panel shows each model's mean accuracy under stochastic sampling (over $N = 80$ trials), together with greedy accuracy (markers). Error bars denote one standard deviation across trials and illustrate the variability introduced by test-time scaling. Models are color-coded consistently across benchmarks. The figure shows substantial heterogeneity in both absolute performance and sampling variance, with HMMT'25 notably more difficult than the other three benchmarks.

A Counterexample: Average accuracy and Bradley–Terry disagree even at infinite budget

We construct a distribution P (equivalently, a finite pattern that can be repeated) for which the aver-

Table 10: $\text{Bayes}_{\mathcal{U}}$@80 as a consensus ranking. "Consensus rank" sorts methods by their average Kendall's τ_b agreement with all other methods (computed at $N = 80$; ties broken by lower std).

Benchmark	Mean rank	$\text{Bayes}_{\mathcal{U}}$@80 avg.	Best method	Best avg.	Gap
AIME'24	2	0.9414	rasch_mml	0.9417	0.0003
AIME'25	1	0.9344	avg (tie)	0.9344	0.0000
HMMT'25	1	0.9499	avg (tie)	0.9499	0.0000
BrUMO'25	2	0.9542	rasch_mml	0.9547	0.0005
Combined	1	0.9616	avg (tie)	0.9616	0.0000

Table 11: Low-agreement tail: methods whose full-trial rankings have Kendall's $\tau_b < 0.85$ relative to $\text{Bayes}_{\mathcal{U}}$@80 (computed at $N = 80$).

Method	τ_b
AIME'24	
minimax_variant_margin_tie_ignore	0.682
minimax_variant_margin_tie_half	0.682
minimax_variant_winning_votes_tie_half	0.682
minimax_variant_winning_votes_tie_ignore	0.693
nanson_rank_ties_average	0.798
nanson_rank_ties_max	0.802
dynamic_irt_growth	0.821
majority_judgment	0.842
rasch_3pl	0.842
rasch_3pl_map	0.842
AIME'25	
minimax_variant_winning_votes_tie_ignore	0.771
majority_judgment	0.779
minimax_variant_margin_tie_ignore	0.819
minimax_variant_margin_tie_half	0.819
minimax_variant_winning_votes_tie_half	0.819
nanson_rank_ties_max	0.840
nanson_rank_ties_average	0.849
HMMT'25	
nanson_rank_ties_max	0.758
inverse_difficulty	0.811
nanson_rank_ties_average	0.818
minimax_variant_margin_tie_ignore	0.831
minimax_variant_margin_tie_half	0.831
minimax_variant_winning_votes_tie_half	0.831
baldwin_rank_ties_max	0.850
BrUMO'25	
rasch_3pl	0.789
rasch_3pl_map	0.789
minimax_variant_margin_tie_ignore	0.814
minimax_variant_margin_tie_half	0.814
minimax_variant_winning_votes_tie_half	0.814
inverse_difficulty	0.821
Combined	
minimax_variant_winning_votes_tie_ignore	0.748
minimax_variant_margin_tie_ignore	0.825
minimax_variant_margin_tie_half	0.825
minimax_variant_winning_votes_tie_half	0.825
nanson_rank_ties_max	0.843

age ranking and the BT-ML ranking disagree. The construction uses $L = 3$ models. For notational convenience, we label them $0, 1, 2$.

Outcome patterns. Consider the following three outcome vectors in $\{0, 1\}^3$:

$$\text{Type A: } (0, 1, 1),$$
$$\text{Type B: } (1, 0, 0),$$
$$\text{Type C: } (1, 1, 0).$$

Let P place mass

$$\mathbb{P}(\text{A}) = \frac{2}{8}, \qquad \mathbb{P}(\text{B}) = \frac{3}{8}, \qquad \mathbb{P}(\text{C}) = \frac{3}{8}.$$

Equivalently, one may take a deterministic dataset with $M = 8$ questions and $N = 1$ trial, containing exactly 2 questions of Type A, 3 of Type B, and 3 of Type C; repeating this block preserves both rankings, as established by the derivation.

The marginal success probabilities are

$$p_0 = \frac{6}{8} = \frac{3}{4}, \qquad p_1 = \frac{5}{8}, \qquad p_2 = \frac{2}{8} = \frac{1}{4},$$

so the average method ranks

$$0 \;>\; 1 \;>\; 2.$$

For these three types, the decisive-win probabilities $w_{ij} = \mathbb{P}(X_i = 1, X_j = 0)$ are:

$$w_{01} = \frac{3}{8}, \quad w_{10} = \frac{2}{8},$$
$$w_{02} = \frac{6}{8}, \quad w_{20} = \frac{2}{8},$$
$$w_{12} = \frac{3}{8}, \quad w_{21} = 0.$$

For the finite $M = 8, N = 1$ realization, the corresponding win counts are $W_{ij} = 8w_{ij}$, i.e.,

$$W = \begin{pmatrix} 0 & 3 & 6 \\ 2 & 0 & 3 \\ 2 & 0 & 0 \end{pmatrix}. \tag{10}$$

It remains to show that BT-ML ranks $1 > 0 > 2$ for (10), thereby disagreeing with the average ranking.

A convenient characterization of the BT-ML optimum is the standard first-order condition equating observed wins to model-implied expected wins: for each i,

$$\sum_{j \neq i} W_{ij} \;=\; \sum_{j \neq i} (W_{ij} + W_{ji}) \cdot \frac{\pi_i}{\pi_i + \pi_j}. \tag{11}$$

(These equations follow by differentiating (8) with respect to $\log \pi_i$.)

Because BT strengths are identifiable only up to a global scale factor, fix $\pi_2 = 1$ and write $\pi_0 = a$, $\pi_1 = b$. Plugging (10) into (11) yields two independent equations:

$$9 = 5 \cdot \frac{a}{a+b} \;+\; 8 \cdot \frac{a}{a+1}, \tag{12}$$
$$5 = 5 \cdot \frac{b}{a+b} \;+\; 3 \cdot \frac{b}{b+1}. \tag{13}$$

Table 12: Consensus ranking on AIME'24 by average Kendall's τ_b agreement with all other methods at $N = 80$ (higher is better). Method variants with identical (Avg., Std.) are collapsed; we show the top 10 and bottom 5 groups.

Rank	Method(s)	Avg.	Std.
1	rasch_mml	0.9417	0.0799
2	alpharank, bayes, bayes_ci, bradley_terry_davidson, bradley_terry_davidson_map, dynamic_irt_linear, glicko_tie_draw, hodge_rank_binary_decisive, hodge_rank_binary_total, hodge_rank_binary_uniform, avg, nash_advantage_vs_equilibrium, nash_vs_equilibrium, pagerank, rank_centrality_tie_half, rasch, rasch_map, serial_rank_prob_diff, serial_rank_sign, spectral, thompson	0.9414	0.0815
3	bayes_greedy	0.9407	0.0817
4	bayesian_mcmc, bradley_terry, bradley_terry_map, elo_tie_skip, glicko_tie_correct_draw_only, glicko_tie_skip, mg_pass_at_k_2, pass_hat_k_2, plackett_luce, plackett_luce_map, trueskill	0.9403	0.0817
5	hodge_rank_log_odds_decisive, hodge_rank_log_odds_total, hodge_rank_log_odds_uniform, rank_centrality_tie_ignore, rao_kupper, rao_kupper_map	0.9349	0.0816
6	borda	0.9179	0.0729
7	baldwin_rank_ties_max	0.9163	0.0754
8	elo_tie_correct_draw_only, elo_tie_draw	0.9141	0.0689
9	copeland	0.9108	0.0728
10	schulze_tie_half	0.9045	0.0722
	Omitted ranks 11–22.		
23	nanson_rank_ties_max	0.7943	0.0492
24	nanson_rank_ties_average	0.7904	0.0431
25	dynamic_irt_growth	0.7897	0.0763
26	minimax_variant_winning_votes_tie_ignore	0.6887	0.0691
27	minimax_variant_margin_tie_half, minimax_variant_margin_tie_ignore, minimax_variant_winning_votes_tie_half	0.6777	0.0763

Table 13: Consensus ranking on AIME'25 by average Kendall's τ_b agreement with all other methods at $N = 80$ (higher is better). Method variants with identical (Avg., Std.) are collapsed; we show the top 10 and bottom 5 groups.

Rank	Method(s)	Avg.	Std.
1	alpharank, bayes, bayes_ci, bradley_terry_davidson, bradley_terry_davidson_map, dynamic_irt_linear, hodge_rank_binary_decisive, hodge_rank_binary_total, hodge_rank_binary_uniform, avg, nash_advantage_vs_equilibrium, nash_vs_equilibrium, pagerank, rank_centrality_tie_half, rasch, rasch_map, serial_rank_prob_diff, serial_rank_sign, spectral, thompson	0.9344	0.0595
2	glicko_tie_draw	0.9331	0.0486
3	bayes_greedy	0.9306	0.0577
4	rasch_mml	0.9293	0.0349
5	hodge_rank_log_odds_decisive, hodge_rank_log_odds_total, hodge_rank_log_odds_uniform, rao_kupper, rao_kupper_map	0.9285	0.0541
6	glicko_tie_correct_draw_only	0.9285	0.0522
7	rasch_mml_credible	0.9249	0.0216
8	rasch_3pl, rasch_3pl_map	0.9172	0.0495
9	rasch_2pl, rasch_2pl_map	0.9162	0.0507
10	bradley_terry, bradley_terry_map, plackett_luce, plackett_luce_map	0.9156	0.0486
	Omitted ranks 11–26.		
27	inverse_difficulty	0.8447	0.0455
28	nanson_rank_ties_max	0.8353	0.0256
29	minimax_variant_margin_tie_half, minimax_variant_margin_tie_ignore, minimax_variant_winning_votes_tie_half	0.8280	0.0377
30	majority_judgment	0.8029	0.0322
31	minimax_variant_winning_votes_tie_ignore	0.7923	0.0386

Step 1: solve a in terms of b. From (13),

$$5 - 3 \cdot \frac{b}{b+1} = 5 \cdot \frac{b}{a+b}.$$

The left-hand side simplifies:

$$5 - 3 \cdot \frac{b}{b+1} = \frac{5(b+1) - 3b}{b+1} = \frac{2b+5}{b+1}.$$

Thus,

$$\frac{5b}{a+b} = \frac{2b+5}{b+1} \tag{14}$$

$$\implies \quad a + b = \frac{5b(b+1)}{2b+5} \tag{15}$$

$$\implies \quad a = \frac{3b^2}{2b+5}. \tag{16}$$

Step 2: determine b from a one-dimensional equation. Substitute (16) into (12). The substitu-

Table 14: Consensus ranking on HMMT'25 by average Kendall's τ_b agreement with all other methods at $N = 80$ (higher is better). Method variants with identical (Avg., Std.) are collapsed; we show the top 10 and bottom 5 groups.

Rank	Method(s)	Avg.	Std.
1	alpharank, bayes, bayes_ci, bradley_terry, bradley_terry_davidson, bradley_terry_davidson_map, bradley_terry_map, dynamic_irt_linear, elo_tie_correct_draw_only, elo_tie_skip, glicko_tie_correct_draw_only, glicko_tie_draw, glicko_tie_skip, hodge_rank_binary_decisive, hodge_rank_binary_total, hodge_rank_binary_uniform, hodge_rank_log_odds_decisive, hodge_rank_log_odds_total, hodge_rank_log_odds_uniform, avg, nash_advantage_vs_equilibrium, nash_vs_equilibrium, pagerank, plackett_luce, plackett_luce_map, rank_centrality_tie_half, rao_kupper, rao_kupper_map, rasch, rasch_map, serial_rank_prob_diff, serial_rank_sign, spectral, thompson, trueskill	0.9499	0.0631
2	rasch_mml	0.9494	0.0442
3	bayes_greedy	0.9468	0.0556
4	rasch_3pl, rasch_3pl_map	0.9420	0.0624
5	rank_centrality_tie_ignore	0.9415	0.0511
6	elo_tie_draw	0.9356	0.0561
7	bayesian_mcmc	0.9335	0.0495
8	dynamic_irt_growth	0.9287	0.0434
9	pass_at_k_2	0.9169	0.0325
10	rasch_2pl, rasch_2pl_map	0.9161	0.0629
	Omitted ranks 11–22.		
23	majority_judgment	0.8636	0.0276
24	minimax_variant_margin_tie_half, minimax_variant_margin_tie_ignore, minimax_variant_winning_votes_tie_half	0.8391	0.0388
25	inverse_difficulty	0.8319	0.0458
26	nanson_rank_ties_average	0.8184	0.0156
27	nanson_rank_ties_max	0.7763	0.0346

tion gives

$$\frac{a}{a+b} = \frac{\frac{3b^2}{2b+5}}{\frac{5b(b+1)}{2b+5}} = \frac{3b}{5(b+1)},$$

so the first term in (12) becomes $5 \cdot \frac{a}{a+b} = \frac{3b}{b+1}$. Also,

$$\frac{a}{a+1} = \frac{\frac{3b^2}{2b+5}}{\frac{3b^2}{2b+5}+1} = \frac{3b^2}{3b^2+2b+5}.$$

Therefore (12) is equivalent to

$$9 = \frac{3b}{b+1} + 8 \cdot \frac{3b^2}{3b^2+2b+5}.$$

Simplifying gives the cubic equation

$$2b^3 - 5b^2 - 16b - 15 = 0. \tag{17}$$

Let $f(b) = 2b^3 - 5b^2 - 16b - 15$. We have

$$f(4) = 128 - 80 - 64 - 15 = -31 < 0,$$
$$f(5) = 250 - 125 - 80 - 15 = 30 > 0,$$

so there exists a root $b^\star \in (4, 5)$. Moreover,

$$f'(b) = 6b^2 - 10b - 16 = 2(3b^2 - 5b - 8),$$

whose positive root is $b = \frac{5+\sqrt{121}}{6} = \frac{16}{6} = \frac{8}{3}$. Hence f is strictly increasing for all $b > \frac{8}{3}$, implying the root $b^\star \in (4, 5)$ is unique. We therefore conclude that the BT-ML solution (under $\pi_2 = 1$) satisfies $b = b^\star \in (4, 5)$ and $a = \frac{3(b^\star)^2}{2b^\star+5}$.

Step 3: show $b > a > 1$, hence BT ranks $1 > 0 > 2$. Using (16),

$$\frac{b}{a} = \frac{b}{\frac{3b^2}{2b+5}} = \frac{2b+5}{3b}.$$

Thus, $b > a$ holds exactly when $(2b+5)/(3b) > 1$, i.e., when $b < 5$. Since $b^\star \in (4, 5)$, we have $b^\star > a$.

It remains to show $a > 1$. Suppose for contradiction that $a \le 1$. We have already established $b > a$, so $b \ge a$. Then $\frac{a}{a+b} \le \frac{a}{2a} = \frac{1}{2}$ and $\frac{a}{a+1} \le \frac{1}{2}$. Plugging into (12) gives

$$9 = 5 \cdot \frac{a}{a+b} + 8 \cdot \frac{a}{a+1} \le 5 \cdot \frac{1}{2} + 8 \cdot \frac{1}{2} = \frac{13}{2} < 9,$$

a contradiction. Hence $a > 1 = \pi_2$. Putting these inequalities together yields

$$\pi_1 = b^\star \ > \ \pi_0 = a \ > \ \pi_2 = 1,$$

so BT ranks

$$1 \ > \ 0 \ > \ 2.$$

This contradicts the average ranking $0 > 1 > 2$, establishing that the two methods can induce different orderings even in the absence of sampling noise.

From a Finite Counterexample to "No Convergence" as $M \to \infty$ or $N \to \infty$. The counterexample rules out a general theorem forcing

Table 15: Consensus ranking on BrUMO'25 by average Kendall's τ_b agreement with all other methods at $N = 80$ (higher is better). Method variants with identical (Avg., Std.) are collapsed; we show the top 10 and bottom 5 groups.

Rank	Method(s)	Avg.	Std.
1	rasch_mml	0.9547	0.0581
2	alpharank, bayes, bayes_ci, bayes_greedy, bradley_terry_davidson, bradley_terry_davidson_map, dynamic_irt_linear, glicko_tie_draw, hodge_rank_binary_decisive, hodge_rank_binary_total, hodge_rank_binary_uniform, hodge_rank_log_odds_decisive, hodge_rank_log_odds_total, hodge_rank_log_odds_uniform, avg, nash_advantage_vs_equilibrium, nash_vs_equilibrium, pagerank, rank_centrality_tie_half, rao_kupper, rao_kupper_map, rasch, rasch_map, serial_rank_prob_diff, serial_rank_sign, spectral, thompson	0.9542	0.0588
3	glicko_tie_correct_draw_only	0.9501	0.0575
4	mg_pass_at_k_2, pass_hat_k_2	0.9490	0.0582
5	elo_tie_draw	0.9423	0.0589
6	borda, win_rate	0.9399	0.0470
7	bayesian_mcmc, bradley_terry, bradley_terry_map, elo_tie_correct_draw_only, elo_tie_skip, glicko_tie_skip, plackett_luce, plackett_luce_map, trueskill	0.9382	0.0592
8	copeland	0.9376	0.0516
9	bradley_terry_luce, bradley_terry_luce_map	0.9350	0.0579
10	dynamic_irt_growth	0.9318	0.0505
	Omitted ranks 11–19.		
20	nanson_rank_ties_average, nanson_rank_ties_max	0.8437	0.0294
21	majority_judgment	0.8267	0.0393
22	inverse_difficulty	0.8156	0.0273
23	minimax_variant_margin_tie_half, minimax_variant_margin_tie_ignore, minimax_variant_winning_votes_tie_half	0.8113	0.0446
24	rasch_3pl, rasch_3pl_map	0.7893	0.0396

average and BT rankings to coincide in the large-budget limit. To connect it directly to $M \to \infty$ or $N \to \infty$, it suffices that both methods are invariant under replication.

Replication invariance (deterministic construction). Let R be any fixed tensor. For an integer $k \geq 1$, define: (i) *question replication* $R^{(k,M)}$ by repeating the M questions k times (so $M' = kM$ and $N' = N$), and (ii) *trial replication* $R^{(k,N)}$ by repeating the N trials k times (so $M' = M$ and $N' = kN$). Then:

1. Average scores are unchanged:
$$\widehat{p}_\ell^{\,\mathrm{avg}}(R^{(k,M)}) = \widehat{p}_\ell^{\,\mathrm{avg}}(R^{(k,N)}) = \widehat{p}_\ell^{\,\mathrm{avg}}(R).$$
2. The decisive-win matrix scales linearly:
$$W(R^{(k,M)}) = k\,W(R)$$
$$W(R^{(k,N)}) = k\,W(R).$$
3. The BT-ML maximizer is unchanged, because the log-likelihood scales as
$$\ell(\pi; kW) = k\,\ell(\pi; W),$$
and therefore has the same maximizer.

Therefore, if two methods disagree on R, they disagree on $R^{(k,M)}$ for arbitrarily large M and on $R^{(k,N)}$ for arbitrarily large N. Applied to the $M = 8, N = 1$ tensor corresponding to (10), this yields an explicit sequence with $M \to \infty$ (or $N \to \infty$) for which the average and BT rankings remain different at every budget.

Stochastic formulation (i.i.d. construction). Alternatively, under the i.i.d. model of Section C.2, the same discrepancy appears at the population level. For the distribution P in Section C.2, the limiting average ranking is determined by (p_ℓ) and yields $0 > 1 > 2$, while the limiting BT ranking is determined by the maximizer of (9) and yields $1 > 0 > 2$. Thus, even with independent sampling and $MN \to \infty$, the two rankings can converge to different limits.

C.3 Implications and Support for the Gold-Standard Definition

This analysis has a direct implication for benchmarking ranking methods: there is no method-independent guarantee that all reasonable procedures converge to the same ordering as the evaluation budget grows. Different ranking procedures correspond to different statistical targets.

Why this happens. Average-based ranking targets the marginal success probabilities $p_\ell = \mathbb{P}(X_\ell = 1)$. BT instead targets the latent strengths that best explain the decisive pairwise win rates $w_{ij} = \mathbb{P}(X_i = 1, X_j = 0)$ through a logistic choice model. These are different summaries of the same joint outcome distribution P. The counterexample in Section C.2 isolates the mechanism: a model can have higher marginal accuracy while assigning less decisive-win mass against another model, which shifts the BT optimum.

Table 16: Consensus ranking on the combined benchmark by average Kendall's τ_b agreement with all other methods at $N = 80$ (higher is better). Method variants with identical (Avg., Std.) are collapsed; we show the top 10 and bottom 5 groups.

Rank	Method(s)	Avg.	Std.
1	alpharank, bayes, bayes_ci, bradley_terry_davidson, bradley_terry_davidson_map, dynamic_irt_linear, glicko_tie_draw, hodge_rank_binary_decisive, hodge_rank_binary_total, hodge_rank_binary_uniform, kemeny_young_tie_half, kemeny_young_tie_ignore, avg, nash_advantage_vs_equilibrium, nash_vs_equilibrium, pagerank, rank_centrality_tie_half, rasch, rasch_map, serial_rank_prob_diff, serial_rank_sign, spectral, thompson	0.9616	0.0559
2	copeland, ranked_pairs_strength_margin_tie_half, ranked_pairs_strength_margin_tie_ignore, ranked_pairs_strength_winning_votes_tie_half, ranked_pairs_strength_winning_votes_tie_ignore, schulze_tie_half, schulze_tie_ignore	0.9602	0.0546
3	hodge_rank_log_odds_decisive, hodge_rank_log_odds_total, hodge_rank_log_odds_uniform, rao_kupper, rao_kupper_map	0.9567	0.0516
4	glicko_tie_correct_draw_only	0.9566	0.0522
5	baldwin_rank_ties_average	0.9558	0.0495
6	borda	0.9557	0.0516
7	rank_centrality_tie_ignore	0.9554	0.0505
8	bayesian_mcmc	0.9512	0.0493
9	bayes_greedy	0.9508	0.0473
10	rasch_2pl	0.9502	0.0486
	Omitted ranks 11–26.		
27	nanson_rank_ties_average	0.8562	0.0263
28	elo_tie_draw	0.8552	0.0239
29	minimax_variant_margin_tie_half, minimax_variant_margin_tie_ignore, minimax_variant_winning_votes_tie_half	0.8339	0.0324
30	nanson_rank_ties_max	0.8333	0.0302
31	minimax_variant_winning_votes_tie_ignore	0.7665	0.0390

Why a gold standard is needed. Because ranking methods need not share a common asymptotic ordering, claims about "distance to the truth" require a specified target ordering. Otherwise even statements such as "method A converges faster than method B" are ambiguous.

Our choice: $\mathrm{Bayes}_{\mathcal{U}}@N$. We define the gold-standard ordering as $\mathrm{Bayes}_{\mathcal{U}}@N$ (with $N = 80$ in our experiments). This definition is supported by three considerations:

1. **Interpretability and decision relevance.** $\mathrm{Bayes}_{\mathcal{U}}@N$ estimates the probability that a model solves a randomly drawn benchmark item under the sampling policy. This is an accuracy-like quantity with a direct operational meaning.

2. **Minimal modeling assumptions.** $\mathrm{Bayes}_{\mathcal{U}}@N$ (and avg@N) depend only on marginal correctness and do not impose a parametric pairwise-choice model. Methods such as BT are useful when the pairwise-choice model is appropriate, but their induced ordering is not, in general, a refinement of accuracy.

3. **Consistency under increasing budget.** Under i.i.d. sampling of (m, n) pairs, $\mathrm{Bayes}_{\mathcal{U}}@N$ converges to p_ℓ as $MN \to \infty$, making it a natural "infinite-budget" reference for accuracy-based evaluation.

Relationship to self-consistency. This non-convergence result does *not* argue against BT or other rankers. It instead clarifies that two evaluations are complementary: agreement with an explicit accuracy-based target, and *self-consistency*, i.e., how quickly a method stabilizes toward its own full-budget ordering. The former asks whether a method matches the chosen reference; the latter asks how stable the method itself becomes as trials accumulate. The counterexample shows why these questions are not interchangeable.

C.4 Minimality of the eight-question construction

The counterexample in Section C.2 uses $M = 8$ questions. The same setting ($L = 3$, $N = 1$, and BT-ML fit from decisive wins) also yields a minimality fact: there is no *strict* disagreement example with fewer than eight questions.

Proposition (minimal M for strict disagreement; verified by exhaustive enumeration). Assume $L = 3$ and $N = 1$. Assume moreover that the average ranking is strict (all three average scores are distinct), and that BT-ML is well-defined and finite (equivalently, the directed win graph with an edge $i \to j$ whenever $W_{ij} > 0$ is strongly connected, which ensures a unique BT-ML maximizer up to

Table 17: Methods that induce *exactly* the same ranking as $\mathrm{Bayes}_{\mathcal{U}}$@80 ($\tau_b = 1$) when computed on the full $N = 80$ trials (excluding avg itself).

Benchmark	Count	Methods
AIME'24	20	alpharank, bayes, bayes_ci, bradley_terry_davidson, bradley_terry_davidson_map, dynamic_irt_linear, glicko_tie_draw, hodge_rank_binary_decisive, hodge_rank_binary_total, hodge_rank_binary_uniform, nash_advantage_vs_equilibrium, nash_vs_equilibrium, pagerank, rank_centrality_tie_half, rasch, rasch_map, serial_rank_prob_diff, serial_rank_sign, spectral, thompson
AIME'25	19	alpharank, bayes, bayes_ci, bradley_terry_davidson, bradley_terry_davidson_map, dynamic_irt_linear, hodge_rank_binary_decisive, hodge_rank_binary_total, hodge_rank_binary_uniform, nash_advantage_vs_equilibrium, nash_vs_equilibrium, pagerank, rank_centrality_tie_half, rasch, rasch_map, serial_rank_prob_diff, serial_rank_sign, spectral, thompson
HMMT'25	34	alpharank, bayes, bayes_ci, bradley_terry, bradley_terry_davidson, bradley_terry_davidson_map, bradley_terry_map, dynamic_irt_linear, elo_tie_correct_draw_only, elo_tie_skip, glicko_tie_correct_draw_only, glicko_tie_draw, glicko_tie_skip, hodge_rank_binary_decisive, hodge_rank_binary_total, hodge_rank_binary_uniform, hodge_rank_log_odds_decisive, hodge_rank_log_odds_total, hodge_rank_log_odds_uniform, nash_advantage_vs_equilibrium, nash_vs_equilibrium, pagerank, plackett_luce, plackett_luce_map, rank_centrality_tie_half, rao_kupper, rao_kupper_map, rasch, rasch_map, serial_rank_prob_diff, serial_rank_sign, spectral, thompson, trueskill
BrUMO'25	26	alpharank, bayes, bayes_ci, bayes_greedy, bradley_terry_davidson, bradley_terry_davidson_map, dynamic_irt_linear, glicko_tie_draw, hodge_rank_binary_decisive, hodge_rank_binary_total, hodge_rank_binary_uniform, hodge_rank_log_odds_decisive, hodge_rank_log_odds_total, hodge_rank_log_odds_uniform, nash_advantage_vs_equilibrium, nash_vs_equilibrium, pagerank, rank_centrality_tie_half, rao_kupper, rao_kupper_map, rasch, rasch_map, serial_rank_prob_diff, serial_rank_sign, spectral, thompson
Combined	22	alpharank, bayes, bayes_ci, bradley_terry_davidson, bradley_terry_davidson_map, dynamic_irt_linear, glicko_tie_draw, hodge_rank_binary_decisive, hodge_rank_binary_total, hodge_rank_binary_uniform, kemeny_young_tie_half, kemeny_young_tie_ignore, nash_advantage_vs_equilibrium, nash_vs_equilibrium, pagerank, rank_centrality_tie_half, rasch, rasch_map, serial_rank_prob_diff, serial_rank_sign, spectral, thompson

global scale). If BT-ML disagrees with the average ranking, then $M \geq 8$.

Verification. With $N = 1$, each question produces an outcome pattern in $\{0, 1\}^3$. Hence, up to permutation of questions, any dataset with M questions is determined by the count vector $c = (c_x)_{x \in \{0,1\}^3} \in \mathbb{N}^8$ with $\sum_x c_x = M$. For fixed M, there are $\binom{M+7}{7}$ such vectors; thus the total number of datasets with $M \leq 7$ is

$$\sum_{M=1}^{7} \binom{M+7}{7} = 6434.$$

For each such dataset, we compute the induced average ordering and the BT-ML ordering (obtained by maximizing (8), equivalently solving (11)). Restricting to datasets with (i) strict average ordering and (ii) strong connectivity (so the BT-ML maximizer is unique up to scale), an exhaustive enumeration yields 1506 instances for $M \leq 7$; in all of them the BT-ML ordering agrees with the average ordering. Therefore, no strict-disagreement example exists for $M \leq 7$.

Section C.2 exhibits a strict-disagreement dataset at $M = 8$, so $M_{\min} = 8$.

D Ranking-Method Stability at $N = 1$

We provide additional details for the $N = 1$ stability analyses in Section 3.2. Method rankings on the Combined benchmark are reported for (i) gold-standard agreement (method@1 vs. $\mathrm{Bayes}_{\mathcal{U}}$@80) and (ii) self-consistency (method@1 vs. method@80), collapsing method variants with identical mean and standard deviation across the 80 single-trial draws.

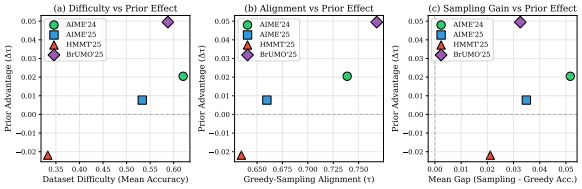

Figure 6: Across our four benchmarks, the prior advantage is not monotonically related to difficulty (a), but it is associated with greedy–sampling alignment (b). The sampling–greedy accuracy gap (c) shows no clear relationship.

As a pair-level diagnostic, we compute *gap-conditional stability*: pooled over benchmarks, $\mathrm{Bayes}_{\mathcal{U}}$@1 reversals concentrate among near-tied pairs under the $\mathrm{Bayes}_{\mathcal{U}}$@80 gap $g_{ij} = |\mu_i - \mu_j|$ ($g_{ij} \leq 0.02$: reversal 0.350, tie 0.151, pairwise correctness 0.566), while well-separated pairs are ordered almost perfectly ($g_{ij} \geq 0.10$: correctness 0.985; $g_{ij} \geq 0.20$: correctness 0.999).

E Additional Prior Diagnostics

Supplementary diagnostics for the empirical-prior analysis in Section 3.4 are shown in Figs. 6 and 7.

Table 18: Gold-standard agreement at $N = 1$ on the combined benchmark, measured as Kendall's τ_b between each method's single-trial ranking and the gold standard ($\text{Bayes}_{\mathcal{U}}$@80). Statistics are computed over 80 single-trial draws. Methods with identical mean/std. values are collapsed; we show the top 10 and bottom 5 groups.

Rank	Method(s)	Mean	Std.
1	`baldwin_rank_ties_average`, `bayes`, `bayes_ci`, `borda`, `copeland`, `majority_judgment`, `avg`, `minimax_variant_margin_tie_half`, `minimax_variant_margin_tie_ignore`, `minimax_variant_winning_votes_tie_half`, `nash_advantage_vs_equilibrium`, `nash_vs_equilibrium`, `pagerank`, `rank_centrality_tie_half`, `ranked_pairs_strength_margin_tie_half`, `ranked_pairs_strength_margin_tie_ignore`, `ranked_pairs_strength_winning_votes_tie_half`, `ranked_pairs_strength_winning_votes_tie_ignore`, `schulze_tie_half`, `schulze_tie_ignore`, `spectral`	0.8647	0.0486
2	`alpharank`	0.8646	0.0486
3	`rasch_mml_credible`	0.8642	0.0351
4	`hodge_rank_binary_uniform`	0.8623	0.0491
5	`hodge_rank_binary_decisive`	0.8623	0.0484
6	`hodge_rank_binary_total`	0.8616	0.0493
7	`serial_rank_sign`	0.8615	0.0503
8	`hodge_rank_log_odds_total`, `hodge_rank_log_odds_uniform`	0.8603	0.0482
9	`rao_kupper_map`	0.8603	0.0483
10	`rao_kupper`	0.8601	0.0484
	Omitted ranks 11–38.		
39	`nanson_rank_ties_average`	0.8067	0.0363
40	`bradley_terry_luce_map`	0.8064	0.0556
41	`bradley_terry_luce`	0.8058	0.0554
42	`bayes_greedy`	0.7856	0.0309
43	`nanson_rank_ties_max`	0.7825	0.0394

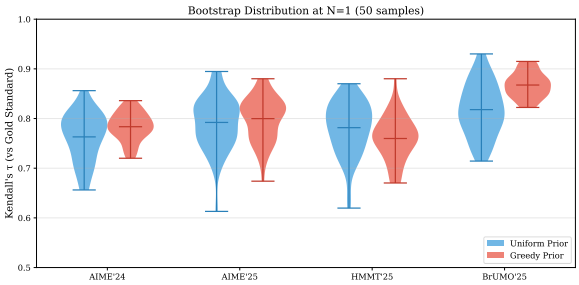

Figure 7: Bootstrap distributions of Kendall's τ_b at $N = 1$ (50 samples). Violin plots show the full distribution; the greedy prior (red) yields narrower distributions but can shift the mean negatively (HMMT'25) or positively (BrUMO'25).

F Categorical Ranking

We report the experimental setup and per-dataset results for the categorical-ranking experiments summarized in Section 3.5.

F.1 Setup

The binary Bayesian estimator (Section 2.3) models each trial outcome as $R_{lmn} \in \{0, 1\}$ and places a Beta prior on the per-question solve rate. The categorical extension maps each completion to one of $C + 1$ categories, yielding outcomes $R_{lmn} \in \{0, \dots, C\}$ defined by auxiliary signals extracted during generation. A categorical *scheme* s specifies:

1. a categorical mapping ϕ_s: completion features $\to \{0, \dots, C_s\}$, which assigns each completion to a category based on predicates over the base signals (Table 20), and
2. a utility weight vector $\mathbf{w}_s = (w_0, \dots, w_{C_s}) \in \mathbb{R}^{C_s+1}$, encoding the relative value of each category.

Bayesian estimation replaces the Beta–binomial model with a Dirichlet–multinomial model: for each model–question pair, a symmetric Dirichlet prior is placed on the $C + 1$ category probabilities $\boldsymbol{\theta} = (\theta_0, \dots, \theta_C)$, and the posterior mean of the weighted utility $\sum_{k=0}^{C} w_k \hat{\theta}_k$ is computed. Model-level scores are then aggregated across questions, as in the binary case.

Base signals. For each of the $L = 11$ models in the categorical cohort, we extract 9 features per completion (Table 20). These features span five domains: answer format (`has_box`), correctness (`is_correct`), generation cost (`token_ratio`, `repeated_pattern`), decoding confidence (`prompt_bpt`, `completion_bpt`), and external verification via CompassVerifier (`compass_A/B/C`). The feature tensors have shape $(N, M, 9)$ per model, with $N = 80$ trials and $M = 30$ questions per benchmark.

`CompassVerifier-3B` provides the external verification signals. Its scores on completions generated by the other models define the verifier-based categorical schemes. Verifier inference uses `Transformers` (Wolf et al., 2019) and `Accelerate` (Gugger et al., 2022), with FlashAttention kernels (Dao, 2023) and the `DFloat11` for-

Table 19: Self-consistency at $N = 1$ on the combined benchmark, measured as Kendall's τ_b between each method's single-trial ranking and its own full-trial ranking (method@80). Statistics are computed over 80 single-trial draws. Methods with identical (Mean, Std.) are collapsed; we show the top 10 and bottom 5 groups.

Rank	Method(s)	Mean	Std.
1	nanson_rank_ties_average	0.8925	0.0497
2	rasch_mml_credible	0.8831	0.0370
3	nanson_rank_ties_max	0.8669	0.0589
4	baldwin_rank_ties_average	0.8664	0.0492
5	copeland, ranked_pairs_strength_margin_tie_half, ranked_pairs_strength_margin_tie_ignore, ranked_pairs_strength_winning_votes_tie_half, ranked_pairs_strength_winning_votes_tie_ignore, schulze_tie_half, schulze_tie_ignore	0.8654	0.0489
6	rasch_mml	0.8648	0.0417
7	bayes, bayes_ci, avg, nash_advantage_vs_equilibrium, nash_vs_equilibrium, pagerank, rank_centrality_tie_half, spectral	0.8647	0.0486
8	alpharank	0.8646	0.0486
9	borda	0.8646	0.0499
10	hodge_rank_binary_uniform	0.8623	0.0491
	Omitted ranks 11–44.		
45	elo_tie_correct_draw_only	0.8074	0.0507
46	bayes_greedy	0.8064	0.0309
47	elo_tie_draw	0.8063	0.0507
48	minimax_variant_margin_tie_half, minimax_variant_margin_tie_ignore, minimax_variant_winning_votes_tie_half	0.7963	0.0454
49	minimax_variant_winning_votes_tie_ignore	0.7655	0.0455

Table 20: Nine base signals extracted per completion for the categorical ranking experiments. Each model–question–trial entry produces a vector in $\mathbb{R}^9$.

#	Signal	Description
1	has_box	Boxed final answer present (0/1)
2	is_correct	Exact-match correctness (0/1)
3	token_ratio	Completion tokens / 32768
4	repeated_pattern	Non-stop finish reason (0/1)
5	prompt_bpt	Prompt bits-per-token
6	completion_bpt	Completion bits-per-token
7	compass_A	Verifier $P(\text{correct})$
8	compass_B	Verifier $P(\text{wrong})$
9	compass_C	Verifier $P(\text{irrelevant})$

mat (Zhang et al., 2025) for throughput.

Derived predicates and thresholds. Several predicates are shared across schemes. All thresholds are computed per-model from the available samples:

- **Invalid**: repeated_pattern $= 1$ or compass_C ≥ 0.5.
- **Confidence**: High confidence := completion_bpt $\leq P_{40}$(completion_bpt); wrong–high-confidence := wrong and completion_bpt $\leq P_{60}$(completion_bpt | wrong).
- **Prompt OOD**: prompt_bpt $\geq P_{90}$(prompt_bpt).
- **Efficiency bands**: Economical/moderate/verbose based on P_{33} and P_{66} of token_ratio.
- **Verifier**: CompassVerifier dominant label is $\arg\max(A, B, C)$; verifier-high := $A \geq 0.6$.

Scheme definitions. The experiments include 25 categorical schemes spanning correctness-only baselines (A, H, S, Y), confidence-aware (C, I, J, V), format-aware (B, P, T), efficiency-aware (F, G, M), verifier-based (D, K, O, U, Z), OOD-aware (E, N, W), abstention-aware (L, Q), and composite (R) variants. Several schemes are metric-level near-duplicates (e.g., A $\equiv$ S, H $\equiv$ Y, L $\equiv$ Q), so the reported comparison uses 8 non-redundant representative schemes covering distinct design axes (Table 21).

Evaluation protocol. For each scheme, we apply the same $N = 1$ subsampling protocol as in Section 3.2: one of the $N = 80$ trials is subsampled per question, the scheme's categorical ranking is computed, and Kendall's τ_b is measured against three references:

1. **Gold-standard** (τ_{GS}): agreement with the binary $\text{Bayes}_{\mathcal{U}}$@80 ranking, which treats outcomes as correct/wrong with a uniform Dirichlet prior.
2. **Self-consistency** (τ_{Self}): agreement with the scheme's own all-80-trial ranking (Scheme@80).
3. **Greedy-prior** (τ_{Greedy}): agreement with $\text{Bayes}_{\mathbf{R}_0}$@80, the binary Bayes ranking incorporating a greedy-decoding empirical prior.

Statistics (mean and standard deviation) are computed over the 80 single-trial draws. Combined results aggregate the four benchmarks ($M = 120$ questions) and are reported in Table 5; per-dataset results are reported in Table 22.

Table 21: Eight representative categorical schemes used in Section 3.5. Each scheme maps a completion to one of $C + 1$ categories using the base signals in Table 20 and scores the result with a utility weight vector $\mathbf{w}$. Category 0 is always *Invalid* ($w_0 = 0$) unless otherwise noted.

Scheme	Intent	Categories (k)	Weights $\mathbf{w}$
Conservative	Penalize confidently-wrong	1: Wrong $\wedge$ HighConf 2: Wrong $\wedge$ LowConf 3: Correct	$(0, -0.10, 0.05, 1.00)$
Efficiency-adj.	Discount verbose correct	1–3: Wrong $\times$ {Econ., Mod., Verb.} 4–6: Correct $\times$ {Econ., Mod., Verb.}	$(0, 0.10, 0.07, 0.03, 1, 0.92, 0.85)$
Format-aware	Reward boxed correct	1: Wrong $\wedge$ Unboxed; 2: Wrong $\wedge$ Boxed 3: Correct; 4: Correct $\wedge$ Boxed	$(0, 0.10, 0.05, 0.90, 1)$
Balanced comp.	Format $\times$ confidence	1: Wrong $\wedge$ Unboxed; 2–3: Wrong $\wedge$ Boxed $\times$ Conf 4–7: Correct $\times$ {Un/Boxed} $\times$ Conf	$(0, 0.10, 0.06, -0.02, 0.90, 0.95, 0.97, 1)$
OOD-robust	Reward in-distribution	1: OOD $\wedge$ Wrong; 2: InDist $\wedge$ Wrong 3: OOD $\wedge$ Correct; 4: InDist $\wedge$ Correct	$(0, 0.05, 0.10, 0.95, 1)$
Rare-event	OOD + abstention	1: OOD $\wedge$ Wrong; 2: OOD $\wedge$ Correct 3: InDist $\wedge$ Wrong; 4: InDist $\wedge$ Correct; 5: Abstain	$(0, 0.05, 1, 0.08, 0.95, 0.20)$
Verifier-calib.	Penalize false-positive	1: Wrong $\wedge$ $A \geq 0.6$; 2: Wrong $\wedge$ $A < 0.6$ 3–5: Correct $\times$ $\{A_{\text{low}}, A_{\text{mid}}, A_{\text{high}}\}$	$(0, -0.05, 0.05, 0.88, 0.94, 1)$
Verifier-only	No ground truth	0: Repeated; 1: Dominant $= C$ 2: Dominant $= B$; 3: Dominant $= A$	$(0, 0, 0.1, 1)$

F.2 Per-Dataset Results

Table 22 reports gold-standard agreement and self-consistency for each benchmark separately. The results show three main patterns.

Narrow spread on individual benchmarks. On each benchmark individually, all eight schemes achieve τ_{GS} between 0.73 and 0.83, with inter-scheme variation much smaller than on the combined benchmark. On AIME'24, the range across the 8 schemes is only 0.007 (0.813–0.820). This narrow spread reflects the limited information available from a single trial with $M = 30$ questions and $L = 11$ models; the combined benchmark ($M = 120$) offers finer discrimination among category structures.

Verifier-only degrades on hard benchmarks. The Verifier-only scheme exhibits the largest performance drop on the harder benchmarks: τ_{GS} falls from 0.813 (AIME'24) to 0.753 (HMMT'25) and 0.734 (BrUMO'25), a decline of 0.06–0.08. In contrast, correctness-driven schemes (Conservative, Efficiency-adjusted, Format-aware) remain above 0.80 on all benchmarks. This pattern suggests that CompassVerifier judgments are less reliable proxies for correctness on more challenging problems.

Self-consistency converges to gold-standard on individual benchmarks. On AIME'24, the self-consistency column is nearly identical to the gold-standard column for most schemes, indicating that the all-80 scheme ranking coincides with the binary $\text{Bayes}_{\mathcal{U}}$@80 ranking when the number of questions is small. On the combined benchmark (Table 5), self-consistency consistently exceeds gold-standard agreement, reflecting convergence of each scheme to its own distinct ordering when given enough questions.

G Arena Ranking

Our experiments consider a dense benchmark tensor $\mathbf{R} \in \{0, 1\}^{L \times M \times N}$: every model is evaluated on every question, and repeated stochastic trials provide multiple binary outcomes for each model–question pair. Arena evaluation, exemplified by preference leaderboards such as Chatbot Arena, uses a different observation model. Its primitive datum is a comparison among a small set of model responses to a prompt, so the data form a sparse, possibly time-varying comparison graph rather than a complete model–question–trial tensor. We analyze how the ranking families studied in this work transfer to this sparse-comparison regime, and where additional assumptions are required.

G.1 Observation Model

Let

$$\mathcal{D}_{\text{arena}} = \{(a_t, b_t, x_t, y_t, \omega_t)\}_{t=1}^{T}$$

denote a pairwise Arena log. At comparison t, models $a_t, b_t \in \{1, \ldots, L\}$ respond to prompt x_t; a human or model judge returns $y_t \in \{a_t \succ b_t,\ b_t \succ a_t,\ a_t \sim b_t\}$; and $\omega_t \geq 0$ is an optional weight for reliability, deduplication, or target-distribution reweighting. Writing $i \succ_t j$ for the event that model i is preferred to model j in comparison t, let $\mathcal{T}_{ij} = \{t : \{a_t, b_t\} = \{i, j\}\}$. We define

$$W_{ij} = \sum_{t \in \mathcal{T}_{ij}} \omega_t \mathbf{1}\{i \succ_t j\},$$

Table 22: Per-dataset categorical ranking at $N = 1$. Gold-standard agreement (τ_{GS}: vs. $\text{Bayes}_{\mathcal{U}}@80$) and self-consistency ($\tau_{\text{Self}}$: vs. Scheme@80) for the 8 representative schemes. Values are mean Kendall's τ_b over 80 single-trial draws.

	Gold-standard agreement (τ_{GS})				Self-consistency (τ_{Self})			
Scheme	AIME'24	AIME'25	HMMT'25	BrUMO'25	AIME'24	AIME'25	HMMT'25	BrUMO'25
Conservative	0.814	0.813	0.801	0.815	0.814	0.813	0.801	0.820
Efficiency-adj.	0.814	0.821	0.812	0.817	0.814	0.814	0.814	0.828
Format-aware	0.820	0.808	0.812	0.819	0.820	0.811	0.813	0.830
Balanced comp.	0.816	0.810	0.804	0.806	0.816	0.813	0.805	0.816
OOD-robust	0.819	0.806	0.788	0.810	0.819	0.803	0.802	0.824
Rare-event	0.816	0.804	0.793	0.816	0.816	0.801	0.807	0.828
Verifier-calib.	0.817	0.802	0.786	0.796	0.810	0.801	0.800	0.807
Verifier-only	0.813	0.805	0.753	0.734	0.806	0.809	0.795	0.810

$$T_{ij} = \sum_{t \in \mathcal{T}_{ij}} \omega_t \mathbf{1}\{a_t \sim b_t\},$$

with $W_{ii} = T_{ii} = 0$ and $C_{ij} = W_{ij} + W_{ji} + T_{ij}$. These counts induce the comparison graph

$$G_{\text{arena}} = (V, E), \qquad V = \{1, \dots, L\},$$
$$E = \{\{i, j\} : C_{ij} > 0\}.$$

In the dense benchmark setting, each question–trial pair induces pairwise outcomes for all model pairs, so G is complete and $C_{ij} = MN$ for every $i \neq j$. Arena ranking removes this completeness assumption: only co-observed models contribute to an edge, and missing comparisons should not be interpreted as losses.

G.2 Which Ranking Families Transfer

Methods whose sufficient statistics are pairwise comparisons transfer most directly. For example, Bradley–Terry estimation on Arena data uses the same win counts as in the dense reduction. Let

$$p_{ij} = \frac{\exp(\theta_i)}{\exp(\theta_i) + \exp(\theta_j)}, \qquad p_{ji} = 1 - p_{ij}.$$

The log-likelihood is

$$\ell_{\text{BT}}(\theta) = \sum_{i<j} \big(W_{ij} \log p_{ij} + W_{ji} \log p_{ji}\big).$$

Tie-aware variants such as Davidson or Rao–Kupper additionally use T_{ij}. Sequential rating systems (Elo, Glicko, TrueSkill) can be run directly on the timestamped comparison stream, whereas the dense benchmark setting must first expand each question–trial slice into induced pairwise matches.

Graph and spectral methods also transfer after estimating pairwise preference probabilities on observed edges, for example

$$\widehat{P}_{i \succ j} = \frac{W_{ij} + \frac{1}{2} T_{ij} + \alpha}{C_{ij} + 2\alpha}, \qquad \{i, j\} \in E,$$

where $\alpha \geq 0$ is a smoothing constant. These edge weights can be used by PageRank, Rank Centrality, HodgeRank, α-Rank, and related graph-based procedures. Hodge-style decompositions are diagnostically useful because they separate a global ranking potential from cyclic residuals, which may reveal non-transitive preferences caused by prompt specialization, heterogeneous judges, or context-dependent model strengths. If an Arena compares more than two responses to the same prompt, the event may instead be represented by winner and loser sets (U_t, V_t) and passed to listwise or setwise Luce-family models.

Pointwise metrics do not transfer without additional labels. Mean accuracy, Pass@k, Bayes@N, and standard IRT models require absolute model–item outcomes, whereas a preference log records only relative judgments. A response can win a comparison without being correct, or lose to a stronger response despite being acceptable. If the Arena protocol also records absolute signals—for example correctness, rubric scores, verifier labels, or categorical response features—then the problem becomes a masked benchmark rather than a pure preference arena. In that case, pointwise or IRT-style likelihoods can be written over the observed entries only,

$$\mathcal{L}_{\text{masked}} = \prod_{A_{lmn}=1} p(R_{lmn} \mid \theta_l, \beta_m),$$

where A is the observation mask. The mask is essential: zero-filling unobserved entries would conflate non-participation with failure and bias rankings toward frequently sampled models.

Sparsity also changes the role of regularization. In dense benchmarks, every model pair receives the same number of induced comparisons. In an Arena, low-degree models and disconnected components

may be weakly identified or incomparable from data alone. Priors, anchor models, or a small dense benchmark pilot can therefore be more important for Arena ranking than for the controlled dense benchmark setting.

G.3 Evaluation Under Comparison Budgets

The stability protocol in Section 3 extends to Arena logs by replacing the trial budget N with a comparison budget. For a ranking method r, let

$$\pi_r(t) = r(\mathcal{D}_{1:t}), \qquad \pi_r(T) = r(\mathcal{D}_{1:T}).$$

Agreement between $\pi_r(t)$ and $\pi_r(T)$, measured for example by Kendall's τ_b, gives the Arena analogue of low-budget stability and convergence. Prefix evaluation preserves time order and measures how quickly a live leaderboard stabilizes; bootstrap evaluation resamples comparisons, prompts, or user sessions to quantify uncertainty. When observations share prompts, users, or judges, resampling at those higher levels is preferable to treating all comparisons as independent.

Arena rankings should be interpreted conditionally on the prompt distribution, judge population, model-selection policy, and decoding policy used to collect the log. Reporting should therefore include both rank estimates and diagnostics: connectivity and degree statistics of G_{arena}, edge-count imbalance, posterior or bootstrap intervals for model strengths, and pairwise superiority probabilities such as $\Pr(\theta_i > \theta_j \mid \mathcal{D}_{\text{arena}})$. Randomizing presentation order, maintaining anchor models, and stratifying by prompt domain make the estimand more transparent and reduce artifacts from side bias or adaptive sampling.

Arena ranking is therefore the sparse-comparison counterpart of the dense repeated-trial setting. The ranking estimators need not be redesigned when they operate on pairwise or setwise sufficient statistics; the observation model and uncertainty structure instead change because comparisons are sparse, non-uniform, and evolving. In `Scorio`, Arena logs correspond to sparse win/tie matrices or setwise events that can be processed by the paired-comparison, graph, or listwise rankers analyzed in this work.

H Extended Related Work

Test-time scaling produces repeated stochastic outcomes per item, making LLM benchmarking closer to classical repeated-measurement settings than to single-run leaderboards. We summarize the main ranking families used in this work and their typical applications.

Paired-comparison and rating models. Paired-comparison models represent comparisons through win/tie counts and infer latent strengths, with Bradley–Terry as a canonical likelihood-based model (Bradley and Terry, 1952). Practical systems often use online rating updates such as Elo and its extensions (e.g., Glicko) or fully Bayesian skill ratings such as TrueSkill (Elo, 1978; Glickman, 1999; Herbrich et al., 2006). For data with ties, common generalizations include Rao–Kupper and Davidson models (Rao and Kupper, 1967; Davidson, 1970). These models are widely used for preference aggregation in LLM leaderboards (Chiang et al., 2024; Ameli et al., 2025), but are also natural in dense benchmarks once per-item outcomes are reduced to pairwise wins.

Listwise, setwise choice models. When each trial yields an ordering over many items, listwise choice models such as Plackett–Luce provide a likelihood over permutations (Plackett, 1975; Luce, 1959). Davidson–Luce extends setwise choice to allow ties within selected sets (Firth et al., 2019). In our binary benchmark setting, each trial induces a two-level partition (solved vs. unsolved), so these models reduce to structured forms of pairwise likelihoods while still providing a principled view of aggregation.

IRT and difficulty-aware benchmarking. Item response theory models couple model "ability" with item difficulty (and sometimes discrimination), with the Rasch and Birnbaum formulations as classic examples (Rasch, 1960; Birnbaum, 1968). IRT has recently been proposed as a way to disentangle model skill from benchmark composition in LLM evaluation (Zhou et al., 2025). When multiple trials per item are available, repeated-measures extensions and binomial-response formulations are natural (De Boeck and Wilson, 2004; Verhelst and Glas, 1993; Wang and Nydick, 2020), and difficulty reweighting has also been explored in NLP evaluation contexts (Gotou et al., 2020).

Graph, spectral, and social-choice methods. Beyond likelihood-based models, ranking from comparisons has a long tradition in social choice and graph-based aggregation. Voting rules such as Borda and Condorcet-style methods satisfy different axioms and can behave differently under

noise and ties (de Borda, 1781; Condorcet, 1785; Arrow, 1951; Brandt et al., 2016). Spectral and Markov-chain approaches derive scores from transition graphs, including PageRank and Rank Centrality (Page et al., 1999; Negahban et al., 2017); HodgeRank and related spectral methods interpret comparisons as edge flows and decompose them into global and cyclic components (Jiang et al., 2011; Fogel et al., 2016). AlphaRank was introduced for multi-agent evaluation with potentially non-transitive interactions (Omidshafiei et al., 2019), and related work studies open-ended evaluation dynamics (Balduzzi et al., 2019). We place these families in a common test-time-scaling benchmark setting and compare them under controlled increases in the number of repeated trials.

I Experiment Setup and Reproducibility

I.1 Models and Datasets

Datasets. We evaluate on four Olympiad-style math benchmarks: AIME'24 (Mathematical Association of America, 2024), AIME'25 (Mathematical Association of America, 2025), BrUMO'25 (Brown University Math Olympiad Organizers, 2025), and HMMT'25 (Harvard–MIT Mathematics Tournament, 2025). For AIME'24 and AIME'25, we combine AIME I and AIME II from the corresponding year, yielding 30 integer-answer problems per benchmark. For HMMT'25, we use the official February 2025 contest set, which spans algebra, geometry, number theory, and combinatorics. For BrUMO'25, we use the published 2025 problem sets from the tournament archive.

Models. To reduce prompt-format confounds, we use provider-recommended chat templates (defaulting to DeepSeek/Qwen-style templates when no model-specific template is given) and shared decoding settings across models unless noted otherwise. We evaluate the 20 models listed in Table 23: Sky-T1-32B-Flash (NovaSky Team, 2025) (Sky-T1 Flash release), Qwen3-30B-A3B-Thinking-2507 (Qwen Team, 2025) (Qwen3 thinking model), DeepSeek-R1-Distill-Qwen-1.5B (Guo et al., 2025) (1.5B distilled reasoning model), gpt-oss-20b (OpenAI, 2025) (OpenAI open-weight model, evaluated with low, medium, and high Harmony reasoning effort under the default MXFP4 quantization), LIMO-v2 (Ye et al., 2025) (reasoning model), EXAONE-4.0-1.2B (LG AI Research, 2025) (hybrid reasoning/non-reasoning model), OpenReasoning-Nemotron-1.5B (NVIDIA, 2025b) (NVIDIA reasoning model), OpenThinker2-32B (Guha et al., 2025) and OpenThinker3-1.5B (Guha et al., 2025) (models trained from the OpenThoughts data recipes), Phi-4-reasoning and Phi-4-reasoning-plus (Abdin et al., 2025), OpenR1-Distill-7B (Hugging Face, 2025), FuseO1-DeepSeekR1-QwQ-SkyT1-Flash-32B-Preview (FuseAI, 2025), Light-R1-14B-DS (Wen et al., 2025), AceReason-Nemotron-1.1-7B (Liu et al., 2026), NVIDIA-Nemotron-Nano-9B-v2 (NVIDIA, 2025a), Qwen3-4B-Thinking-2507 (Qwen Team, 2025), and Bespoke-Stratos-7B (Bespoke Labs, 2025).

ID	Model	Short name
1	DeepSeek-R1-Distill-Qwen-1.5B	DS-R1-Qwen
2	LIMO-v2	LIMO-v2
3	OpenThinker2-32B	OpenThinker2
4	OpenThinker3-1.5B	OpenThinker3
5	Qwen3-30B-A3B-Thinking-2507	Qwen3-Thinking
6	Sky-T1-32B-Flash	Sky-T1-Flash
7	gpt-oss-20b_high	gpt-oss-high
8	gpt-oss-20b_low	gpt-oss-low
9	gpt-oss-20b_medium	gpt-oss-medium
10	EXAONE-4.0-1.2B	EXAONE-4.0
11	OpenReasoning-Nemotron-1.5B	OR-Nemotron
12	Phi-4-reasoning	Phi-4
13	Phi-4-reasoning-plus	Phi-4-plus
14	OpenR1-Distill-7B	OR1-Distill
15	FuseO1-DeepSeekR1-QwQ-SkyT1-Flash-32B-Preview	FuseO1-DS-QwQ-SkyT1
16	Light-R1-14B-DS	Light-R1-DS
17	AceReason-Nemotron-1.1-7B	AR-Nemotron
18	NVIDIA-Nemotron-Nano-9B-v2	NVIDIA-Nemotron
19	Qwen3-4B-Thinking-2507	Qwen3-4B
20	Bespoke-Stratos-7B	Bespoke

Table 23: Mapping between model IDs, full model names, and the shortened names used in figures and legends.

Prompting. We use provider-recommended prompt templates for each model. For most models, we adopt the standard DeepSeek/Qwen-style prompt, "Please reason step by step, and put your final answer within `\boxed{}`." For `gpt-oss-20b`, we use the OpenAI Harmony prompt template, which specifies three discrete levels of reasoning effort. For `OpenReasoning-Nemotron-1.5B`, we use the task-specific prompt, "Solve the following math problem. Make sure to put the answer (and only the answer) inside `\boxed{}`."

I.2 Reproducibility

For stochastic runs, we use top-p sampling with temperature 0.6, $p = 0.95$, batch size 1, and ran-

Task	Inference Time (hours)	Completion Tokens (M)
AIME'24	1,699.4	680.0
AIME'25	1,878.4	728.3
HMMT'25	2,216.5	851.2
BrUMO'25	1,650.9	666.9
TOTAL	**7,445.2**	**2,926.4**

Table 24: Task-level computational cost aggregated over 20 models, 80 trials, four tasks, and 30 questions per task. Token counts correspond to completion tokens only.

```python
import numpy as np
from scorio import rank

# Binary response tensor: L=3 models, M=4 questions, N=5 trials
R = np.random.randint(0, 2, size=(3, 4, 5))

# Rank by mean accuracy
rankings = rank.avg(R)

# Return both rankings and scores
rankings, scores = rank.avg(R, return_scores=True)
```

Listing 1: Constructing the response tensor and computing rankings with `Scorio`.

dom seeds 1234 through 1313, yielding $N = 80$ trials per dataset–model pair. All models are served with `vLLM` (PagedAttention) (Kwon et al., 2023) in `bf16` precision, except releases that require MXFP4 quantization (e.g., `gpt-oss`). We record log-probabilities for both input prompts and generated tokens, with `max_tokens` set to 32,768. All experiments run on clusters equipped with 8× NVIDIA H200 GPUs (141 GB per GPU).

I.3 Computational Cost and Token Statistics

We evaluate 20 models across four benchmarks, with 80 trials per model and 30 questions per benchmark, for a total of 192,000 independent inference runs. The full evaluation requires 7,445 GPU-hours (approximately 310 GPU-days) and generates 2.96B tokens (2,963,318,176 total); Table 24 reports the task-level totals. Of these tokens, 37M (1.2%) are prompt tokens and 2.93B (98.8%) are completion tokens, for an average of 15,434 tokens per query. Among the four benchmarks, HMMT'25 is the most computationally expensive at 2,217 GPU-hours, whereas BrUMO'25 is the least expensive at 1,651 GPU-hours. Across model configurations, `gpt-oss-20b-low` is the most efficient (48.4 GPU-hours for 9,600 queries) and `LIMO-v2` the least efficient (894.3 GPU-hours for the same workload), with a corpus-wide average of 139.6 seconds per query.

I.4 Rank Correlation Metrics

Kendall's tau Kendall's tau (τ) (Kendall, 1938) measures ordinal agreement between two rankings through pairwise concordance and discordance. For rankings of n items, let n_c and n_d denote the numbers of concordant and discordant pairs, let $n_0 = n(n-1)/2$ be the total number of pairs, and let n_1 and n_2 be the numbers of tied pairs in the two rankings. The two common variants are

$$\text{Tau-a:} \quad \tau_a = \frac{n_c - n_d}{n_0}, \tag{18}$$

$$\text{Tau-b:} \quad \tau_b = \frac{n_c - n_d}{\sqrt{(n_0 - n_1)(n_0 - n_2)}}. \tag{19}$$

Tau-a ignores ties, whereas Tau-b corrects for them. Because ties are common in our setting, we use τ_b throughout.

J `Scorio`, Open-Source Library for LLM Ranking

`Scorio` is a Python library for ranking LLMs from repeated-trial benchmark evaluations under test-time scaling. It provides a unified interface for mapping the response tensor $\mathbf{R} \in \{0,1\}^{L\times M\times N}$ (and, where relevant, optional prior outcomes) to model scores and rankings across evaluation metrics, probabilistic paired-comparison and rating systems, voting rules, listwise choice models, item response theory, and graph- or spectral-based methods. The library is distributed through PyPI as `scorio`.

All ranking methods in `Scorio` operate on the response tensor $\mathbf{R}$, where L is the number of models, M the number of questions, and N the number of trials per question. The implementation represents this tensor as a NumPy array of shape `(L, M, N)`. Listing 1 gives a minimal example of constructing $\mathbf{R}$ and calling a basic ranking method.

The `rank` module uses a common interface: each function takes the tensor $\mathbf{R}$ as the first argument, returns a ranking array of shape `(L,)`, and accepts an optional `return_scores=True` flag to additionally return the underlying scores. Rankings are 1-indexed, with lower values indicating better models.

```python
# Pass@k: probability at least 1 of k draws succeeds
rankings, scores = rank.pass_at_k(R, k=3, return_scores=True)

# G-Pass@k with threshold tau
rankings = rank.g_pass_at_k_tau(R, k=5, tau=0.6)

# Bayesian posterior ranking with optional prior outcomes
R0 = np.random.randint(0, 2, size=(3, 4, 2))  # prior data
rankings = rank.bayes(R, R0=R0)
```

Listing 2: Evaluation-based ranking methods.

```python
# Categorical outcomes: 0=wrong, 1=partial, 2=correct
# L=3 models, M=4 questions, N=5 trials
R_cat = np.random.randint(0, 3, size=(3, 4, 5))

# Weight vector mapping categories to scores
w = np.array([0.0, 0.5, 1.0])

rankings, scores = rank.bayes(R_cat, w=w,
                              return_scores=True)

# Using greedy decoding results as Bayesian prior
# R0 shape (M, D): shared prior across all models
R0_greedy = np.random.randint(0, 3, size=(4, 2))
rankings = rank.bayes(R_cat, w=w, R0=R0_greedy)

# Conservative ranking via posterior quantile
rankings = rank.bayes(R_cat, w=w, R0=R0_greedy,
                      quantile=0.05)
```

Listing 3: Bayes@N with categorical outcomes and greedy prior.

`Scorio` implements ranking methods from several families. Listing 2 illustrates evaluation-based methods, including the Pass@k family that quantifies how reliably models solve questions within k sampled trials.

The `bayes` method generalizes beyond binary correctness to categorical outcomes $R_{lmn} \in \{0, \dots, C\}$ via a weight vector $\mathbf{w} \in \mathbb{R}^{C+1}$ that maps each category to a score. It also accepts an optional prior tensor $\mathbf{R}_0$ that incorporates outcomes from a different evaluation setting (e.g., greedy decoding) as a Bayesian prior. Listing 3 gives examples of both cases.

For probabilistic paired-comparison models, `Scorio` implements the Bradley–Terry model and its extensions, as well as Elo and TrueSkill rating systems (Listing 4). These methods construct pairwise comparisons from $\mathbf{R}$ and estimate latent

```python
# Bradley-Terry maximum likelihood
rankings, scores = rank.bradley_terry(R, return_scores=True)

# Bradley-Terry with MAP regularization
rankings = rank.bradley_terry_map(R, prior=1.0)

# Elo rating system
rankings, scores = rank.elo(R, K=32.0, return_scores=True)

# TrueSkill Bayesian rating
rankings = rank.trueskill(R)
```

Listing 4: Paired-comparison and rating system methods.

```python
# PageRank on the pairwise win-probability graph
rankings, scores = rank.pagerank(R, damping=0.85,
                                 return_scores=True)

# Spectral ranking (principal eigenvector)
rankings = rank.spectral(R)

# Rank centrality via Markov chain stationary distribution
rankings = rank.rank_centrality(R)
```

Listing 5: Graph-based ranking methods.

strength parameters.

Graph-based and spectral methods rank models by analyzing the structure of a pairwise comparison graph derived from $\mathbf{R}$, as shown in Listing 5.

A family-wise list of ranking methods is given in Section J.1, and the exact method configurations used in our experiments are reported in Section J.2.

J.1 Ranking Methods

J.1.1 Pointwise Methods

Mean accuracy. The simplest pointwise score is the mean accuracy

$$s_l^{\text{mean}} := \frac{1}{M} \sum_{m=1}^{M} \widehat{p}_{lm}, \tag{20}$$

which corresponds to `avg` in `Scorio`.

Inverse-difficulty weighting. To emphasize hard questions, `inverse_difficulty` weights each question by the inverse of its global solve rate

Algorithm 1 Pointwise scoring (mean and inverse-difficulty)

Require: $R \in \{0,1\}^{L\times M\times N}$, $\epsilon > 0$
Ensure: Scores $s \in \mathbb{R}^L$
1: Compute $\widehat{p}_{lm} \leftarrow \frac{1}{N}\sum_{n=1}^{N} R_{lmn}$
2: **Mean:** $s_l \leftarrow \frac{1}{M}\sum_{m=1}^{M} \widehat{p}_{lm}$
3: **Inv-diff:** compute $p_m \leftarrow \frac{1}{LN}\sum_{l,n} R_{lmn}$
4: Set $w_m \propto 1/\operatorname{clip}(p_m, \epsilon, 1-\epsilon)$ and normalize to $\sum_m w_m = 1$
5: $s_l \leftarrow \sum_m w_m \widehat{p}_{lm}$

$p_m := \frac{1}{LN}\sum_{l,n} R_{lmn}$:

$$w_m \propto \frac{1}{\operatorname{clip}(p_m, \epsilon, 1-\epsilon)}, \qquad s_l^{\text{inv-diff}} := \sum_{m=1}^{M} w_m \widehat{p}_{lm}, \tag{21}$$

with weights normalized to $\sum_m w_m = 1$.

J.1.2 Evaluation-metric Methods

These methods rank models by *evaluation metrics* computed from per-question trial outcomes. The simplest baseline is mean accuracy (`avg`; Section J.1.1); we next define Pass@k-family metrics and Bayes@N. For a fixed model l, define the per-question success counts $\nu_{lm} := \sum_{n=1}^{N} R_{lmn}$. Each metric defines a per-question score $f(\nu_{lm}; N)$ (or $f(\nu_{lm}; N, k, \tau)$) and then averages across questions.

Pass@k (`pass_at_k`). Pass@k (Chen et al., 2021) is the probability that at least one of k samples is correct. For each question m,

$$\text{Pass@}k_{lm} := 1 - \frac{\binom{N-\nu_{lm}}{k}}{\binom{N}{k}}, \tag{22}$$

and the model-level score is $s_l^{\text{Pass@}k} := \frac{1}{M}\sum_{m=1}^{M} \text{Pass@}k_{lm}$.

Pass-hat@k / G-Pass@k (`pass_hat_k`). This metric (also called G-Pass@k in parts of the recent LLM evaluation literature (Yao et al., 2025)) is the probability that *all* k selected samples are correct:

$$\widehat{\text{Pass@}k}_{lm} := \frac{\binom{\nu_{lm}}{k}}{\binom{N}{k}}, \tag{23}$$

with $s_l^{\widehat{\text{Pass@}k}} := \frac{1}{M}\sum_{m=1}^{M} \widehat{\text{Pass@}k}_{lm}$.

G-Pass@k$_\tau$ (`g_pass_at_k_tau`). G-Pass@k$_\tau$ (Liu et al., 2025) generalizes these metrics by requiring at least $j_0 := \lceil \tau k \rceil$ successes among the k selected samples. Let $X_{lm} \sim \text{Hypergeom}(N, \nu_{lm}, k)$ be the number of successes in a draw of size k without replacement; then

$$\text{G-Pass@}k_{\tau,lm} := \Pr(X_{lm} \ge j_0) = \sum_{j=j_0}^{k} \frac{\binom{\nu_{lm}}{j}\binom{N-\nu_{lm}}{k-j}}{\binom{N}{k}}, \tag{24}$$

and $s_l^{\text{G-Pass@}k_\tau} := \frac{1}{M}\sum_{m=1}^{M} \text{G-Pass@}k_{\tau,lm}$. `Scorio` defines the endpoint $\tau = 0$ to recover Pass@k (and for any $\tau \in (0, 1/k]$ the threshold $j_0 = \lceil \tau k \rceil$ equals 1, so the expression matches Pass@k), while $\tau = 1$ recovers Pass-hat@k.

mG-Pass@k (`mg_pass_at_k`). mG-Pass@k (Liu et al., 2025) aggregates G-Pass@k$_\tau$ over $\tau \in [0.5, 1]$. In `Scorio`, we use the equivalent expectation form

$$\text{mG-Pass@}k_{lm} := \frac{2}{k}\,\mathbb{E}[(X_{lm} - m_0)_+], \qquad m_0 := \left\lceil \tfrac{k}{2} \right\rceil, \tag{25}$$

where $(x)_+ := \max(x, 0)$ and $X_{lm} \sim \text{Hypergeom}(N, \nu_{lm}, k)$. The model-level score is $s_l^{\text{mG-Pass@}k} := \frac{1}{M}\sum_{m=1}^{M} \text{mG-Pass@}k_{lm}$.

Bayes@N (`bayes`). Bayes@N (Hariri et al., 2026) applies to multi-category outcomes $R_{lmn} \in \{0, \dots, C\}$ with a weight vector $w \in \mathbb{R}^{C+1}$. For a fixed model l and question m, let $n_{mk} := \sum_{n=1}^{N} \mathbf{1}\{R_{lmn} = k\}$ be category counts. Optionally, a prior outcome matrix $R_0 \in \{0, \dots, C\}^{M\times D}$ contributes pseudo-counts $n_{mk}^0 := 1 + \sum_{d=1}^{D} \mathbf{1}\{(R_0)_{md} = k\}$ (a Dirichlet$(1, \dots, 1)$ prior), giving $\nu_{mk} := n_{mk} + n_{mk}^0$ and $T := 1 + C + D + N$. Bayes@N returns a posterior mean μ_l and uncertainty σ_l of the weighted score:

$$\mu_l = w_0 + \frac{1}{MT}\sum_{m=1}^{M}\sum_{k=0}^{C} \nu_{mk}(w_k - w_0), \tag{26}$$

$$\sigma_l = \Big(\frac{1}{M^2(T+1)}\sum_{m=1}^{M}\Big[\sum_k \frac{\nu_{mk}}{T}(w_k - w_0)^2 - \Big(\sum_k \frac{\nu_{mk}}{T}(w_k - w_0)\Big)^2\Big]\Big)^{1/2}. \tag{27}$$

`Scorio` ranks by μ_l (default) or by a conservative normal-quantile score $\mu_l + \Phi^{-1}(q)\sigma_l$ for a chosen $q \in [0, 1]$.

J.1.3 Bayesian Methods

Thompson sampling ranking (`thompson`). Thompson sampling (Thompson, 1933; Russo et al., 2018) ranks by Monte Carlo samples from a conjugate Beta–Binomial posterior over each model's aggregate success probability. We model $p_l \sim \mathrm{Beta}(\alpha, \beta)$ and treat all MN trials as i.i.d. Bernoulli outcomes (Gelman et al., 2013). Let $S_l := \sum_{m=1}^{M} \sum_{n=1}^{N} R_{lmn}$ be the total number of successes for model l; then

$$p_l \mid \mathbf{R} \sim \mathrm{Beta}(\alpha + S_l,\ \beta + MN - S_l)\,. \quad (28)$$

For $t = 1, \ldots, T$ we draw $p_l^{(t)} \sim p_l \mid \mathbf{R}$ independently for each model, compute the induced rank $r_l^{(t)} \in \{1, \ldots, L\}$ (smaller is better), and score by the negative average rank

$$s_l^{\mathrm{TS}} := -\frac{1}{T} \sum_{t=1}^{T} r_l^{(t)}. \quad (29)$$

Bayesian Bradley–Terry via MCMC (`bayesian_mcmc`). To obtain a full Bayesian posterior over paired-comparison strengths, we combine the Bradley–Terry likelihood (Bradley and Terry, 1952) with a Gaussian prior and approximate the posterior with Metropolis–Hastings sampling (Metropolis et al., 1953; Hastings, 1970). We first form decisive win counts

$$W_{ij} := \sum_{m=1}^{M} \sum_{n=1}^{N} \mathbf{1}\{R_{imn} = 1,\ R_{jmn} = 0\}, \quad (30)$$

ignoring ties (both correct or both incorrect). Parameterizing $\pi_i = \exp(\theta_i)$, the BT likelihood is

$$\begin{aligned} \Pr(i \succ j \mid \theta) &= \frac{\exp(\theta_i)}{\exp(\theta_i) + \exp(\theta_j)}, \\ \log p(\mathbf{W} \mid \theta) &= \sum_{i \neq j} W_{ij} \log \Pr(i \succ j \mid \theta), \end{aligned} \quad (31)$$

with an independent prior $\theta_i \sim \mathcal{N}(0, \sigma^2)$ (Caron and Doucet, 2012). We sample from $p(\theta \mid \mathbf{W})$ and rank models by the posterior mean score $s_i^{\mathrm{MCMC}} := \mathbb{E}[\theta_i \mid \mathbf{W}]$.

J.1.4 Voting-based Methods

Voting rules aggregate per-question preferences into a global ranking. To adapt them to our test-time-scaling setting, we treat each question m as a "voter" that ranks models by their per-question solve frequency across trials:

$$k_{lm} := \sum_{n=1}^{N} R_{lmn} \in \{0, 1, \ldots, N\}. \quad (32)$$

When $N = 1$, each question induces only a two-level ranking (correct vs. incorrect), so Borda/-Copeland reduce to (ties of) accuracy-based ordering; when $N > 1$ these rules exploit the additional resolution from k_{lm}.

Borda count. For each question m, let $r_{lm} \in \{1, \ldots, L\}$ be the (tie-averaged) rank of model l when sorting $k_{\cdot m}$ in descending order (smaller rank is better). The Borda score is

$$s_l^{\mathrm{Borda}} := \sum_{m=1}^{M} (L - r_{lm}), \quad (33)$$

which assigns $(L-1)$ points for a unique first place and 0 for a unique last place, with ties receiving the average of the tied positions (de Borda, 1781; Brandt et al., 2016).

Copeland. For each pair (i, j), define the number of questions that prefer i to j as $W_{ij}^{(q)} := \sum_m \mathbb{I}[k_{im} > k_{jm}]$. Copeland declares i to beat j if $W_{ij}^{(q)} > W_{ji}^{(q)}$ and scores each model by net pairwise dominance:

$$s_i^{\mathrm{Copeland}} := \sum_{j \neq i} \mathrm{sign}\left(W_{ij}^{(q)} - W_{ji}^{(q)}\right), \quad (34)$$

where $\mathrm{sign}(0) = 0$ (Copeland, 1951; Brandt et al., 2016).

Win rate. Using the same question-level win counts $W^{(q)}$, define a model's win rate as the fraction of decisive pairwise outcomes it wins:

$$s_i^{\mathrm{winrate}} := \frac{\sum_{j \neq i} W_{ij}^{(q)}}{\sum_{j \neq i} \left(W_{ij}^{(q)} + W_{ji}^{(q)}\right)}, \quad (35)$$

with the convention $s_i^{\mathrm{winrate}} = 0.5$ if the denominator is zero.

Condorcet-style pairwise-majority rules. Many voting rules are defined from an aggregated pairwise preference matrix. To incorporate per-question ties when $k_{im} = k_{jm}$, we define

$$P_{ij}^{(q)} := \sum_{m=1}^{M} \left(\mathbb{I}[k_{im} > k_{jm}] + \tfrac{1}{2}\mathbb{I}[k_{im} = k_{jm}]\right), \quad (36)$$

so that $P_{ij}^{(q)} + P_{ji}^{(q)} = M$. Let margins be $\Delta_{ij} := P_{ij}^{(q)} - P_{ji}^{(q)}$.

Minimax (Simpson–Kramer). The minimax score is based on a model's worst pairwise defeat:

$$s_i^{\text{minimax}} := -\max_{j \neq i} \max(0, \Delta_{ji}), \tag{37}$$

and ranks models by the size of their worst defeat (closer to 0 is better) (Brandt et al., 2016).

Schulze (beatpath). Schulze computes strongest-path strengths p_{ij} in the directed graph of pairwise victories and ranks i above j if $p_{ij} > p_{ji}$ (Schulze, 2011; Brandt et al., 2016).

Ranked Pairs (Tideman). Ranked Pairs sorts pairwise victories by strength (e.g., margin Δ_{ij}), then locks them in that order whenever doing so does not introduce a cycle; the resulting acyclic dominance graph induces a ranking (Tideman, 1987; Brandt et al., 2016).

Kemeny–Young. Kemeny–Young returns an ordering π that maximizes agreement with the pairwise preferences:

$$\pi \in \arg \max_{\text{total orders } \pi} \sum_{i \prec_\pi j} P_{ij}^{(q)}, \tag{38}$$

which is equivalent to a maximum-likelihood ranking under certain noise models and is a classic Condorcet extension (Kemeny, 1959; Young, 1977; Brandt et al., 2016). (Exact optimization is NP-hard in general; we solve the induced linear ordering problem via MILP for the problem sizes in this paper.)

Borda elimination rules (Nanson and Baldwin). Nanson's method iteratively recomputes Borda scores over remaining candidates and removes those below the mean, while Baldwin's method removes the lowest Borda scorer(s) each round (Nanson, 1883; Baldwin, 1926; Brandt et al., 2016).

Majority Judgment. Majority Judgment treats $k_{lm} \in \{0, \ldots, N\}$ as discrete grades and ranks models by their median grade, breaking ties using the majority-gauge rule (Balinski and Laraki, 2011).

J.1.5 Paired-comparison Probabilistic Models

These methods first reduce $\mathbf{R}$ to pairwise win/tie counts between models, then fit a parametric paired-comparison model. For each ordered pair (i, j), define wins W_{ij} and ties T_{ij} as in Section 2.2 (pairwise representation).

Algorithm 2 Voting rules on per-question trial counts

Require: $R \in \{0,1\}^{L \times M \times N}$
Ensure: Borda scores s^{Borda}, Copeland scores s^{Copeland}, win-rate scores s^{winrate}
Compute $k_{lm} \leftarrow \sum_{n=1}^{N} R_{lmn}$
$s^{\text{Borda}} \leftarrow 0$; $s^{\text{Copeland}} \leftarrow 0$; initialize $W^{(q)} \leftarrow 0$
for $m = 1$ **to** M **do**
  Rank models by $k_{\cdot m}$ (descending) with average-tie ranks $r_{\cdot m}$
  $s_l^{\text{Borda}} \mathrel{+}= L - r_{lm}$ for all l
end for
for $1 \leq i < j \leq L$ **do**
  $W_{ij}^{(q)} \leftarrow \sum_m \mathbb{I}[k_{im} > k_{jm}]$
  $W_{ji}^{(q)} \leftarrow \sum_m \mathbb{I}[k_{jm} > k_{im}]$
  if $W_{ij}^{(q)} > W_{ji}^{(q)}$ **then** $s_i^{\text{Copeland}} \mathrel{+}= 1$; $s_j^{\text{Copeland}} \mathrel{-}= 1$
  else if $W_{ji}^{(q)} > W_{ij}^{(q)}$ **then** $s_i^{\text{Copeland}} \mathrel{-}= 1$; $s_j^{\text{Copeland}} \mathrel{+}= 1$
  end if
end for
$s_i^{\text{winrate}} \leftarrow \frac{\sum_{j \neq i} W_{ij}^{(q)}}{\sum_{j \neq i} (W_{ij}^{(q)} + W_{ji}^{(q)})}$ (or 0.5 if denominator is 0)

Bradley–Terry (BT). The BT model (Bradley and Terry, 1952) assigns each model a positive strength $\pi_i > 0$ and assumes

$$\Pr(i \succ j) = \frac{\pi_i}{\pi_i + \pi_j}. \tag{39}$$

Given win counts W_{ij}, the log-likelihood is

$$\log p(\mathbf{W} \mid \pi) = \sum_{i \neq j} W_{ij} \Big[\log \pi_i - \log(\pi_i + \pi_j) \Big], \tag{40}$$

with identifiability enforced by centering log-strengths. `Scorio` provides ML (`bradley_terry`) and MAP (`bradley_terry_map`) estimation; MAP adds a prior penalty on log-strengths (e.g., Gaussian) (Caron and Doucet, 2012).

Tie extensions. In our binary setting, a pairwise tie occurs when both models are correct or both are incorrect on the same question–trial. `Scorio` implements two classic tie models:

- **Davidson** (Davidson, 1970): adds a tie parameter and models $(i \succ j)$, $(j \succ i)$, and $(i \sim j)$ explicitly (`bradley_terry_davidson`, `bradley_terry_davidson_map`).
- **Rao–Kupper** (Rao and Kupper, 1967): alternative tie parameterization via $\kappa \geq 1$ (`rao_kupper`, `rao_kupper_map`).

Algorithm 3 Paired-comparison models (BT, Davidson, Rao–Kupper) via ML/MAP

Require: $R \in \{0,1\}^{L\times M\times N}$; model family; optional prior penalty on log-strengths; max iterations T
Ensure: Scores (strengths) $\hat{\pi} \in \mathbb{R}_+^L$
1: Compute pairwise win/tie counts (W_{ij}, T_{ij}) from R
2: Parameterize strengths by log-strengths $\theta_i = \log \pi_i$ and enforce identifiability by centering: $\theta \leftarrow \theta - \frac{1}{L}\sum_i \theta_i$
3: Define the family-specific log-likelihood $\log p(W, T \mid \theta, \text{tie-params})$
4: Define objective $\mathcal{L} = -\log p(\cdot) + \text{prior}(\theta)$ (prior term is 0 for ML)
5: Optimize $\mathcal{L}$ with L-BFGS for up to T iterations
6: Return $\hat{\pi}_i = \exp(\hat{\theta}_i)$ as scores (larger is better)

For Davidson, with tie parameter $\nu > 0$,

$$\begin{aligned}\Pr(i \succ j) &= \frac{\pi_i}{\pi_i + \pi_j + \nu\sqrt{\pi_i\pi_j}},\\ \Pr(j \succ i) &= \frac{\pi_j}{\pi_i + \pi_j + \nu\sqrt{\pi_i\pi_j}},\\ \Pr(i \sim j) &= \frac{\nu\sqrt{\pi_i\pi_j}}{\pi_i + \pi_j + \nu\sqrt{\pi_i\pi_j}}.\end{aligned} \tag{41}$$

For Rao–Kupper, with $\kappa \geq 1$,

$$\begin{aligned}\Pr(i \succ j) &= \frac{\pi_i}{\pi_i + \kappa\pi_j},\\ \Pr(j \succ i) &= \frac{\pi_j}{\kappa\pi_i + \pi_j},\\ \Pr(i \sim j) &= \frac{(\kappa^2 - 1)\pi_i\pi_j}{(\pi_i + \kappa\pi_j)(\kappa\pi_i + \pi_j)}.\end{aligned} \tag{42}$$

J.1.6 Sequential Rating Systems

Sequential rating systems process a stream of head-to-head "matches" rather than aggregating all pairwise outcomes into a single count matrix. In our benchmark setting, the natural match stream is induced by each question–trial (m, n): for every pair of models (i, j), we observe a binary outcome pair $(R_{imn}, R_{jmn}) \in \{0,1\}^2$ and declare i to beat j if $(1, 0)$ and j to beat i if $(0, 1)$. When $(R_{imn}, R_{jmn}) \in \{(1, 1), (0, 0)\}$, the comparison is a tie; `Scorio` exposes tie-handling policies (e.g., treat ties as draws or ignore certain ties) for these methods.

Elo. Elo (Elo, 1978) maintains a scalar rating r_i for each model. For a match between i and j, define the expected score

$$E_{ij} := \frac{1}{1 + 10^{(r_j - r_i)/400}}, \tag{43}$$

and let $S_{ij} \in \{0, \frac{1}{2}, 1\}$ be the realized match score for i against j (win/draw/loss, depending on the tie-handling rule). The sequential Elo update is

$$\begin{aligned}r_i &\leftarrow r_i + K(S_{ij} - E_{ij}),\\ r_j &\leftarrow r_j + K((1 - S_{ij}) - (1 - E_{ij})),\end{aligned} \tag{44}$$

with learning rate $K > 0$ (`elo` in `Scorio`). Because the updates are sequential, the final ratings can depend on the order in which the match stream is processed.

Glicko. Glicko (Glickman, 1999) augments Elo with an uncertainty parameter (rating deviation) RD_i and updates ratings using batches of matches within rating periods. In our implementation, each question–trial (m, n) constitutes one rating period containing all pairwise matches on that (m, n). Define $q := \ln(10)/400$ and

$$g(\mathrm{RD}) := \frac{1}{\sqrt{1 + \frac{3q^2\mathrm{RD}^2}{\pi^2}}}. \tag{45}$$

For a player i in a rating period with opponents $j \in \mathcal{O}_i$ and outcomes S_{ij}, define expected scores

$$E_{ij} := \frac{1}{1 + 10^{-g(\mathrm{RD}_j)(r_i - r_j)/400}}, \tag{46}$$

and

$$d_i^2 := \left(q^2 \sum_{j\in\mathcal{O}_i} g(\mathrm{RD}_j)^2 E_{ij}(1 - E_{ij})\right)^{-1}. \tag{47}$$

The Glicko updates are

$$\begin{aligned}\mathrm{RD}_i' &:= \left(\frac{1}{\mathrm{RD}_i^2} + \frac{1}{d_i^2}\right)^{-1/2},\\ r_i' &:= r_i + \frac{q}{\frac{1}{\mathrm{RD}_i^2} + \frac{1}{d_i^2}} \sum_{j\in\mathcal{O}_i} g(\mathrm{RD}_j)(S_{ij} - E_{ij}),\end{aligned} \tag{48}$$

with optional RD inflation between rating periods and a maximum RD cap (as in the original Glicko specification). This corresponds to `glicko` in `Scorio`; we rank by r_i' (larger is better), and RD_i' can be used as an uncertainty summary.

TrueSkill. TrueSkill (Herbrich et al., 2006) is a Bayesian rating system that models each model's latent skill as a Gaussian $\mathcal{N}(\mu_i, \sigma_i^2)$ and updates (μ_i, σ_i) after each match using approximate inference. In `Scorio`, we apply a two-player TrueSkill update to each decisive $(1, 0)$ or $(0, 1)$ pairwise match in the induced stream (ties are ignored) and return the final μ_i as the score (`trueskill`); a per-round dynamics parameter τ inflates σ between rounds to model drift.

J.1.7 Listwise / Setwise Choice Models (Luce Family)

Unlike pairwise models, these methods operate on *setwise* events induced by each question–trial (m, n). Define the winner and loser sets

$$U_{mn} := \{l : R_{lmn} = 1\}, \quad V_{mn} := \{l : R_{lmn} = 0\}. \tag{49}$$

If $U_{mn} = \emptyset$ or $U_{mn} = \mathcal{L}$, the event contains no ranking information and is discarded.

Plackett–Luce (PL). The PL model (Plackett, 1975; Luce, 1959) is a listwise generalization of BT for full rankings. In our binary setting we apply PL to the pairwise win matrix (equivalently BT) and estimate strengths using the MM update from Hunter (2004) (`plackett_luce`, `plackett_luce_map`).

Davidson–Luce (setwise ties). Davidson–Luce (Firth et al., 2019) models the probability of a tied winner set U_{mn} emerging from the full set $U_{mn} \cup V_{mn}$, explicitly accounting for ties within U_{mn} and V_{mn} (`davidson_luce`, `davidson_luce_map`). Let $\pi_i > 0$ be strengths and $\delta_t > 0$ be tie-prevalence parameters with $\delta_1 \equiv 1$. For a comparison set S and tie order t, define $g_t(T) := \left(\prod_{i \in T} \pi_i\right)^{1/t}$ and

$$Z(S) := \sum_{t'=1}^{\min(D,|S|)} \delta_{t'} \cdot \sum_{\substack{T \subseteq S \\ |T|=t'}} g_{t'}(T), \tag{50}$$

where D is the maximum tie order considered. Then, for an event (U, V) with $S = U \cup V$ and $t = |U|$,

$$\Pr(U \succ V \mid S) = \frac{\delta_t \, g_t(U)}{Z(S)}. \tag{51}$$

Bradley–Terry–Luce (BTL) setwise-choice construction. BTL converts each winner $i \in U_{mn}$ into a Luce choice event from $\{i\} \cup V_{mn}$, with choice probability $\Pr(i \mid \{i\} \cup V) = \pi_i / (\pi_i + \sum_{j \in V} \pi_j)$ (`bradley_terry_luce`, `bradley_terry_luce_map`). Equivalently, for an event (U, V) the BTL likelihood factorizes as

$$\Pr(U \succ V) = \prod_{i \in U} \frac{\pi_i}{\pi_i + \sum_{j \in V} \pi_j}. \tag{52}$$

Algorithm 4 MM algorithm for PL/BT on the pairwise win matrix

Require: Pairwise win matrix $W \in \mathbb{R}_+^{L \times L}$, iterations T
Ensure: Strengths $\hat{\pi} \in \mathbb{R}_+^L$ (normalized)
1: $w_i \leftarrow \sum_j W_{ij}$ (total wins); $n_{ij} \leftarrow W_{ij} + W_{ji}$ (total comparisons)
2: Initialize $\pi_i \propto w_i$ and normalize $\sum_i \pi_i = 1$
3: **for** $t = 1$ **to** T **do**
4: **for** $i = 1$ **to** L **do**
5: $d_i \leftarrow \sum_{j \neq i:\, n_{ij} > 0} \frac{n_{ij}}{\pi_i + \pi_j}$
6: $\pi_i \leftarrow w_i / d_i$
7: **end for**
8: Normalize π to sum to 1
9: **end for**
10: Return π

Algorithm 5 Setwise event extraction and Luce-family estimation (Davidson–Luce / BTL)

Require: $R \in \{0,1\}^{L \times M \times N}$; model type $\in \{\text{Davidson–Luce}, \text{BTL}\}$; optional prior on log-strengths; max iterations T
Ensure: Strength scores $\hat{\pi} \in \mathbb{R}_+^L$
1: Build events $\mathcal{E} \leftarrow \{(U_{mn}, V_{mn}) : 0 < |U_{mn}| < L\}$
2: Parameterize $\pi_i = \exp(\theta_i)$ with centered θ for identifiability
3: Define the event log-likelihood $\sum_{(U,V) \in \mathcal{E}} \log p(U \succ V \mid \theta)$ for the chosen model
4: Add prior penalty on θ for MAP (or 0 for ML)
5: Optimize with L-BFGS for up to T iterations and return $\hat{\pi}$

J.1.8 Item Response Theory (IRT) Methods

`Scorio` includes several IRT-inspired ranking methods that treat each model as an "examinee" with a latent ability and each question as an "item" with latent parameters (e.g., difficulty). We use IRT primarily as a *ranking model*: we estimate abilities $\{\theta_l\}_{l=1}^L$ and rank models by θ_l (larger is better), using `rank_scores` for tie-aware rank variants.

Data and binomial reduction. Our raw observations are binary trial outcomes $R_{lmn} \in \{0, 1\}$ for model $l \in \{1, \ldots, L\}$, question $m \in \{1, \ldots, M\}$, and trial $n \in \{1, \ldots, N\}$. When trials are i.i.d. conditional on parameters, the sufficient statistic

for an item-model pair is the correct-count

$$k_{lm} := \sum_{n=1}^{N} R_{lmn} \in \{0, 1, \ldots, N\}, \tag{53}$$

so that likelihood-based IRT estimation can be written as a binomial-response model (McCullagh and Nelder, 1989; De Boeck and Wilson, 2004).

Rasch (1PL). The Rasch model (Rasch, 1960) assumes a single item parameter (difficulty b_m):

$$k_{lm} \sim \text{Binomial}(N,\ \sigma(\theta_l - b_m)), \tag{54}$$

where $\sigma(x) = 1/(1+e^{-x})$. The model is invariant to global shifts $(\theta, b) \mapsto (\theta + c, b + c)$, so we impose an identifiability constraint by centering item difficulties (e.g., $\sum_m b_m = 0$).

2PL and 3PL. The 2PL model (Birnbaum, 1968) adds an item discrimination parameter $a_m > 0$:

$$k_{lm} \sim \text{Binomial}\big(N,\ \sigma\big(a_m(\theta_l - b_m)\big)\big). \tag{55}$$

The 3PL model further adds a pseudo-guessing parameter c_m:

$$\begin{aligned} k_{lm} &\sim \text{Binomial}(N,\ p_{lm}), \\ p_{lm} &:= c_m + (1 - c_m)\sigma\big(a_m(\theta_l - b_m)\big). \end{aligned} \tag{56}$$

In our implementation, we constrain a_m via a log-parameterization and keep c_m in a bounded range (or optionally fix c_m to a known chance level).

Estimation variants used in `Scorio`.

- **JMLE / MLE** (`rasch`, `rasch_2pl`, `rasch_3pl`): optimize the joint log-likelihood over θ and item parameters.
- **MAP** (`rasch_map`, `rasch_2pl_map`, `rasch_3pl_map`): add a prior penalty on abilities, typically Gaussian, as in Bayes modal estimation (Mislevy, 1986).
- **MML + EAP** (`rasch_mml`): integrate out abilities under a population model (we use a standard normal prior), fit item parameters by EM, then compute EAP ability estimates (Bock and Aitkin, 1981; Chen et al., 1998).
- **Credible/LB scoring** (`rasch_mml_credible`): rank by a posterior quantile of θ_l (e.g., a lower bound), which yields a conservative, uncertainty-aware ranking.
- **Dynamic IRT** (`dynamic_irt`): a longitudinal extension that allows per-model trends across trials (Verhelst and Glas, 1993; Wang and Nydick, 2020).

Algorithm 6 Binomial xPL IRT (JMLE/MAP) for ranking

Require: Response tensor $R \in \{0,1\}^{L \times M \times N}$; model type $\in \{1\text{PL}, 2\text{PL}, 3\text{PL}\}$; optional ability prior $p(\theta)$; max iterations T

Ensure: Ability scores $\hat{\theta} \in \mathbb{R}^L$ and optional item parameters

1: Compute counts $k_{lm} \leftarrow \sum_{n=1}^{N} R_{lmn}$ and set $n \leftarrow N$

2: Initialize θ from per-model accuracy; initialize b from per-item solve rate; set $a_m \leftarrow 1$ (2PL/3PL); set $c_m \leftarrow 0.25$ (3PL)

3: Define $p_{lm}(\theta, b, a, c)$ according to the chosen xPL link

4: Define the binomial log-likelihood $\ell(k; n, p) \leftarrow k \log p + (n - k) \log(1 - p)$

5: Define objective (negative log posterior)

$$\mathcal{L}(\theta, b, a, c) = -\sum_{l,m} \ell(k_{lm}; n, p_{lm}) - \log p(\theta).$$

Set $\log p(\theta) = 0$ for pure MLE.

6: Impose identifiability at each iteration by centering item difficulties: $b \leftarrow b - \frac{1}{M}\sum_m b_m$

7: Optimize $\mathcal{L}$ with a quasi-Newton method (e.g., L-BFGS) for up to T iterations

8: Return $\hat{\theta}$ as scores (larger is better) and optionally $\hat{b}$, $\hat{a}$, $\hat{c}$

J.1.9 Graph and Spectral Methods

These methods operate on the pairwise comparison graph derived from the win/tie counts (W_{ij}, T_{ij}) defined in Section 2.2. A common derived quantity is the empirical tied-split win probability

$$\widehat{P}_{i \succ j} := \frac{W_{ij} + \frac{1}{2}T_{ij}}{W_{ij} + W_{ji} + T_{ij}}, \qquad \widehat{P}_{i \succ i} := \tfrac{1}{2}. \tag{57}$$

In our fully observed benchmark setting, $W_{ij} + W_{ji} + T_{ij} = MN$ for all $i \neq j$ (Section 2.2), so $\widehat{P}_{i \succ j}$ is a simple rescaling of aggregated counts.

PageRank. We build a directed weighted graph where an edge from j to i has weight $\widehat{P}_{i \succ j}$ (interpreting "losers link to winners"), then form a column-stochastic transition matrix P by normalizing each column:

$$P_{ij} := \frac{\widehat{P}_{i \succ j}}{\sum_{k \neq j} \widehat{P}_{k \succ j}} \quad (i \neq j), \tag{58}$$

Algorithm 7 Rasch MML (EM + quadrature) with EAP and posterior-quantile scoring

Require: Counts $k \in \{0, \dots, N\}^{L \times M}$; trials N; quadrature points $\{\theta_q, w_q\}_{q=1}^{Q}$; EM iterations S

Ensure: EAP scores $\hat{\theta}^{\mathrm{EAP}}$ (or quantile scores) and item difficulties $\hat{b}$

Initialize item difficulties b from per-item solve rates and center b
for $s = 1$ **to** S **do**
  E-step: compute $\log p(k_l \mid \theta_q, b)$ for each model l and quadrature point q
  Compute posterior weights $w_{lq} \propto \exp(\log p(k_l \mid \theta_q, b))\, w_q$ and normalize over q
  Define $\ell(k; n, p) \leftarrow k \log p + (n - k) \log(1 - p)$
  M-step: for each item m, update b_m by minimizing

$$-\sum_{l,q} w_{lq}\, \ell(k_{lm}; N, \sigma(\theta_q - b_m))\,.$$

  Center b
end for
Recompute posterior weights w_{lq} under final b
Compute EAP scores: $\hat{\theta}_l^{\mathrm{EAP}} \leftarrow \sum_q w_{lq} \theta_q$
(Optional) Compute quantile score $Q_\alpha(\theta_l \mid k)$ from the discrete posterior CDF (used by `rasch_mml_credible`)
Return scores and $\hat{b}$

with the standard dangling-node convention of a uniform column if the denominator is zero. PageRank scores $r \in \Delta^{L-1}$ solve

$$r = d\, P\, r + (1 - d)\, \tfrac{1}{L} \mathbf{1}, \tag{59}$$

where $d \in (0, 1)$ is the damping factor and $\mathbf{1}$ is the all-ones vector (Page et al., 1999). This corresponds to `pagerank` in `Scorio`.

Spectral (eigenvector centrality). We form the nonnegative matrix W with off-diagonal entries $W_{ij} := \widehat{P}_{i \succ j}$ and set the diagonal to the row sum $W_{ii} := \sum_{j \neq i} W_{ij}$ (a self-loop that makes the matrix diagonally dominant). The spectral score vector is the principal right eigenvector $v \geq 0$ of W, normalized to $\sum_i v_i = 1$. This corresponds to `spectral` in `Scorio`.

Rank Centrality. Rank Centrality (Negahban et al., 2017) constructs a random walk on the comparison graph whose transition probabilities prefer moving from a model to those that beat it. Let $d_{\max}$ be the maximum (undirected) degree of the comparison graph (in our benchmark setting $d_{\max} = L - 1$). Define a row-stochastic matrix

$$\begin{aligned} P_{ij} &:= \frac{1}{d_{\max}} \widehat{P}_{j \succ i} \quad (i \neq j), \\ P_{ii} &:= 1 - \sum_{j \neq i} P_{ij}. \end{aligned} \tag{60}$$

The stationary distribution π of P is used as the score vector (larger π_i is better). This corresponds to `rank_centrality` in `Scorio`.

Algorithm 8 Dynamic IRT growth model (logistic longitudinal Rasch)

Require: Response tensor $R \in \{0, 1\}^{L \times M \times N}$; normalized time grid $t_n \in [0, 1]$; max iterations T

Ensure: Baseline abilities $\hat{\theta}_0 \in \mathbb{R}^L$, slopes $\hat{\theta}_1 \in \mathbb{R}^L$, and item difficulties $\hat{b} \in \mathbb{R}^M$

Fit the longitudinal model $P(R_{lmn} = 1) = \sigma(\theta_{0,l} + \theta_{1,l} t_n - b_m)$ by maximizing the Bernoulli likelihood over all (l, m, n)
Add weak regularization on slopes (e.g., $\|\theta_1\|_2^2$) to avoid overfitting i.i.d. sampling noise
Center b for identifiability
Optimize with a quasi-Newton method (e.g., L-BFGS) for up to T iterations
Return $\hat{\theta}_0$ as ranking scores and optionally $\hat{\theta}_1, \hat{b}$

α-Rank. α-Rank (Omidshafiei et al., 2019) ranks strategies via evolutionary dynamics by constructing a Markov chain over models using fixation probabilities in a finite population. In our constant-sum binary evaluation setting, we treat $\widehat{P}_{i \succ j}$ as the payoff to strategy i against j (so the per-match payoff sum is 1 when ties are split as $\frac{1}{2}$). For population size $m \geq 2$ and selection intensity $\alpha \geq 0$, the (constant-sum) fixation probability of a mutant r in a resident population s is

$$\rho_{r,s} := \begin{cases} \frac{1 - \exp(-u)}{1 - \exp(-mu)} & u \neq 0, \\ \frac{1}{m} & u = 0, \end{cases} \tag{61}$$

$$u := \alpha \frac{m}{m - 1} \left(\widehat{P}_{r \succ s} - \tfrac{1}{2} \right). \tag{62}$$

The induced Markov chain on models has off-diagonal transitions $C_{sr} := \frac{1}{L-1} \rho_{r,s}$ and diagonal $C_{ss} := 1 - \sum_{r \neq s} C_{sr}$; the stationary distribution of C is the α-Rank score vector. This corresponds to `alpharank` in `Scorio`.

Nash equilibrium mixture. Following the use of Nash equilibria as evaluation summaries in symmetric zero-sum games (Balduzzi et al., 2019), we define a zero-sum payoff matrix

$$A_{ij} := 2\widehat{P}_{i\succ j} - 1, \qquad A_{ii} := 0, \tag{63}$$

which is antisymmetric when $\widehat{P}$ is derived from tied-split win rates. We compute a maximin mixed strategy $x \in \Delta^{L-1}$ (a Nash equilibrium strategy for the row player)

$$x \in \arg\max_{x\in\Delta^{L-1}} \min_{y\in\Delta^{L-1}} x^\top A y, \tag{64}$$

via a standard linear program. To obtain a per-model evaluation score ("Nash averaging"), we then score each model by its expected performance against the equilibrium mixture opponent:

$$s_i := \sum_{j=1}^{L} \widehat{P}_{i\succ j}\, x_j \in [0,1], \tag{65}$$

and rank models by s (higher is better). We additionally report the equilibrium mixture x as a strategic summary of the meta-game when needed. This corresponds to `nash` in `Scorio`.

J.1.10 Seriation-based Methods

SerialRank. SerialRank (Fogel et al., 2016) is a spectral seriation method that constructs a similarity graph from a skew-symmetric comparison matrix. From pairwise counts (W, T), define

$$C_{ij} := \frac{W_{ij} - W_{ji}}{W_{ij} + W_{ji} + T_{ij}} \in [-1,1], \qquad C_{ii} := 0, \tag{66}$$

so that $C_{ij} > 0$ indicates i tends to beat j (and C is skew-symmetric). SerialRank forms the similarity matrix

$$S := \tfrac{1}{2}\left(L\,\mathbf{1}\mathbf{1}^\top + CC^\top\right), \tag{67}$$

then computes the graph Laplacian $L_S := \mathrm{diag}(S\mathbf{1}) - S$. The ordering is given by sorting a Fiedler vector (the eigenvector associated with the second-smallest eigenvalue of L_S), with the sign chosen to best agree with the observed comparisons. This corresponds to `serial_rank` in `Scorio`.

J.1.11 Hodge-theoretic Methods

HodgeRank. HodgeRank (Jiang et al., 2011) interprets pairwise comparisons as a skew-symmetric edge flow on a graph and recovers global scores by least squares. Using the tied-split probabilities from Section J.1.9, define the observed edge flow

$$\overline{Y}_{ij} := \widehat{P}_{j\succ i} - \widehat{P}_{i\succ j} = \frac{W_{ji} - W_{ij}}{W_{ij} + W_{ji} + T_{ij}}, \qquad \overline{Y}_{ii} := 0, \tag{68}$$

and choose symmetric edge weights w_{ij} (e.g., the total number of comparisons on edge (i,j)). HodgeRank solves the weighted least-squares problem

$$s^\star \in \arg\min_{s\in\mathbb{R}^L} \sum_{i<j} w_{ij}\left((s_j - s_i) - \overline{Y}_{ij}\right)^2 = \arg\min_s \|\mathrm{grad}(s) - \overline{Y}\|_{2,w}^2, \tag{69}$$

which reduces to a weighted graph Laplacian system; we compute the minimum-norm solution via the Moore–Penrose pseudoinverse and rank by $s^\star$ (higher is better). This corresponds to `hodge_rank` in `Scorio`.

J.2 Ranking Method APIs and Hyperparameters

We evaluate the ranking methods described in Section J.1. Each method maps the trial outcome tensor $R \in \{0,1\}^{L\times M\times N}$ (and, where applicable, an optional prior tensor R_0) to a ranking over the L models. For reproducibility, we list the exact API identifiers and argument values used in our experiments; `None` denotes an unset optional argument.

Metrics.

- `avg`
- `pass_at_k_2 (k=2)`
- `pass_hat_k_2 (k=2)`
- `mg_pass_at_k_2 (k=2)`
- `bayes (R0=None, quantile=None)`
- `bayes_greedy (R0=R0, quantile=None)`
- `bayes_ci (R0=None, quantile=0.05)`
- `inverse_difficulty (return_scores=false, clip_range=[0.01, 0.99])`

Pairwise rating.

- `elo_tie_skip (K=0.05, initial_rating=1500.0, tie_handling=skip)`
- `elo_tie_draw (K=0.05, initial_rating=1500.0, tie_handling=draw)`
- `elo_tie_correct_draw_only (K=0.05, initial_rating=1500.0, tie_handling=correct_draw_only)`

- glicko_tie_skip (initial_rating=1500.0, initial_rd=350.0, c=0.0, rd_max=350.0, tie_handling=skip, return_deviation=false)
- glicko_tie_draw (initial_rating=1500.0, initial_rd=350.0, c=0.0, rd_max=350.0, tie_handling=draw, return_deviation=false)
- glicko_tie_correct_draw_only (initial_rating=1500.0, initial_rd=350.0, c=0.0, rd_max=350.0, tie_handling=correct_draw_only, return_deviation=false)
- trueskill (mu_initial=25.0, sigma_initial=8.333333333333334, beta=4.166666666666667, tau=0.00333333333)

Probabilistic comparisons.

- bradley_terry (return_scores=false, max_iter=500)
- bradley_terry_map (prior=1.0, max_iter=500)
- bradley_terry_davidson (return_scores=false, max_iter=500)
- bradley_terry_davidson_map (prior=1.0, max_iter=500)
- rao_kupper (tie_strength=1.1, max_iter=500)
- rao_kupper_map (tie_strength=1.1, prior=1.0, max_iter=500)
- thompson (n_samples=10000, prior_alpha=1.0, prior_beta=1.0, seed=42)
- bayesian_mcmc (n_samples=5000, burnin=1000, prior_var=1.0, seed=42)
- plackett_luce (return_scores=false, max_iter=500, tol=1e-08)
- plackett_luce_map (prior=1.0, max_iter=500)
- bradley_terry_luce (return_scores=false, max_iter=500)
- bradley_terry_luce_map (prior=1.0, max_iter=500)

Voting rules.

- borda (return_scores=false)
- copeland (return_scores=false)
- win_rate (return_scores=false)
- minimax_variant_margin_tie_ignore (variant=margin, tie_policy=ignore)
- minimax_variant_margin_tie_half (variant=margin, tie_policy=half)
- minimax_variant_winning_votes_tie_ignore (variant=winning_votes, tie_policy=ignore)
- minimax_variant_winning_votes_tie_half (variant=winning_votes, tie_policy=half)
- schulze_tie_ignore (tie_policy=ignore)
- schulze_tie_half (tie_policy=half)
- ranked_pairs_strength_margin_tie_ignore (strength=margin, tie_policy=ignore)
- ranked_pairs_strength_margin_tie_half (strength=margin, tie_policy=half)
- ranked_pairs_strength_winning_votes_tie_ignore (strength=winning_votes, tie_policy=ignore)
- ranked_pairs_strength_winning_votes_tie_half (strength=winning_votes, tie_policy=half)
- kemeny_young_tie_ignore (tie_policy=ignore, time_limit=None)
- kemeny_young_tie_half (tie_policy=half, time_limit=None)
- nanson_rank_ties_average (rank_ties=average)
- nanson_rank_ties_max (rank_ties=max)
- baldwin_rank_ties_average (rank_ties=average)
- baldwin_rank_ties_max (rank_ties=max)
- majority_judgment (return_scores=false)

IRT.

- rasch (return_scores=false, max_iter=500, return_item_params=false)
- rasch_map (prior=1.0, max_iter=500, return_item_params=false)
- rasch_2pl (return_scores=false, max_iter=500, return_item_params=false)
- rasch_2pl_map (prior=1.0, max_iter=500, return_item_params=false)
- rasch_3pl (return_scores=false, max_iter=500, fix_guessing=None, return_item_params=false)
- rasch_3pl_map (prior=1.0, max_iter=500, fix_guessing=None, return_item_params=false)
- rasch_mml (return_scores=false, max_iter=100, em_iter=20, n_quadrature=21,

return_item_params=false)
- rasch_mml_credible (quantile=0.05, max_iter=100, em_iter=20, n_quadrature=21)
- dynamic_irt_linear (variant=linear, max_iter=500, return_item_params=false)
- dynamic_irt_growth (variant=growth, max_iter=500, return_item_params=false)

Graph/game.

- pagerank (damping=0.85, max_iter=100, tol=1e-12)
- spectral (max_iter=10000, tol=1e-12)
- alpharank (alpha=1.0, population_size=50, max_iter=100000, tol=1e-12)
- nash_vs_equilibrium (n_iter=100, temperature=0.1, solver=lp, score_type=vs_equilibrium, return_equilibrium=false)
- nash_advantage_vs_equilibrium (n_iter=100, temperature=0.1, solver=lp, score_type=advantage_vs_equilibrium, return_equilibrium=false)
- rank_centrality_tie_ignore (tie_handling=ignore, smoothing=0.0, teleport=0.0, max_iter=10000, tol=1e-12)
- rank_centrality_tie_half (tie_handling=half, smoothing=0.0, teleport=0.0, max_iter=10000, tol=1e-12)
- serial_rank_prob_diff (comparison=prob_diff)
- serial_rank_sign (comparison=sign)
- hodge_rank_binary_total (pairwise_stat=binary, weight_method=total, return_diagnostics=false)
- hodge_rank_binary_decisive (pairwise_stat=binary, weight_method=decisive, return_diagnostics=false)
- hodge_rank_binary_uniform (pairwise_stat=binary, weight_method=uniform, return_diagnostics=false)
- hodge_rank_log_odds_total (pairwise_stat=log_odds, weight_method=total, epsilon=0.5, return_diagnostics=false)
- hodge_rank_log_odds_decisive (pairwise_stat=log_odds, weight_method=decisive, epsilon=0.5, return_diagnostics=false)
- hodge_rank_log_odds_uniform (pairwise_stat=log_odds, weight_method=uniform, epsilon=0.5, return_diagnostics=false)