How Fast Should a Model Commit to Supervision? Training Reasoning Models on the J_Q Loss Continuum

Anonymous authors
Paper under double-blind review

Abstract

The standard recipe for training reasoning models — SFT on rationale annotations followed by RLVR — relies on rationale annotations being available and clean. When no annotated rationales exist, SFT cannot start at all, and when the model is too unaligned, RLVR alone stalls at cold start; when annotations are weak, SFT memorizes the errors. Using the Tsallis q-logarithm, we define a loss family J_Q that interpolates between RLVR (at $q{=}0$, the *exploitation pole*) and the log-marginal-likelihood over latent trajectories (at $q{=}1$, the *density-estimation pole*). The standard pipeline corresponds to a stepwise $q{=}1 \to q{=}0$ schedule, and intermediate q enables training in regimes where the two-step pipeline cannot apply. All members share the same gradient direction, differing only by a per-instance amplification $P_{\boldsymbol{\theta}}^{-q}$ that reweights each instance independently of the learning rate. Under gradient flow analysis, we show that the exploitation pole requires $\Omega(1/p_0)$ time to escape cold start but is robust to label noise, while the density-estimation pole escapes in $\Theta\big(\log(1/p_0)\big)$ but memorizes label noise. This separation explains why SFT ($q{=}1$) precedes RLVR ($q{=}0$) in the standard pipeline. We further derive two Monte Carlo estimators that directly optimize fixed-q on the J_Q continuum, without annotated rationales or external verifiers: Gradient-Amplified RL (GARL) and Posterior-Attenuated Fine-Tuning (PAFT), with shared bias $O\big(q/MP_{\boldsymbol{\theta}}^{q}\big)$ but different variance and stability properties. On FinQA, HotPotQA, and MuSiQue, GARL at sufficiently high q escapes cold start where GRPO fails entirely under Qwen 3 and Gemma 4 architectures. Without task prompts, cold-start GARL on Qwen 3 0.6B also matches or exceeds prompted GRPO with reshaped reward on majority vote and best-of-k on all three benchmarks.

1 Introduction

The standard recipe for adapting reasoning models is supervised fine-tuning (SFT) on annotated rationales followed by reinforcement learning from verifiable rewards (RLVR) (Ouyang et al., 2022; DeepSeek-AI, 2025; Shao et al., 2024; Chu et al., 2025). Yet two questions about it lack a unifying theoretical account: why this specific ordering and why RLVR alone stalls at cold start (when initial $P_{\boldsymbol{\theta}}$ is near zero). For the latter, Rao–Blackwellized variants of RL algorithms (Zhou et al., 2026) ensure non-zero gradients but, as we show, reduce variance without accelerating escape.

We provide such an account under exact-match supervision. Using the Tsallis q-logarithm (Tsallis, 1988), we define a loss continuum J_Q with a scalar *commitment* parameter $q \in [0, 1]$ that interpolates between REINFORCE-style exploitation and $\log$-marginal-likelihood maximization. RLVR with exact-match reward optimizes the exploitation endpoint exactly (Proposition B.1); SFT on annotated rationales maximizes an evidence lower bound (ELBO; Neal & Hinton, 1998) on the log-marginal likelihood with the annotations as a fixed variational distribution (Proposition B.2). All members of J_Q share one per-instance gradient direction, differing only by a factor $P_{\boldsymbol{\theta}}^{-q}$ (Figure 1; formal definitions in Section 2). This per-instance reweighting amplifies the gradient on unfamiliar (low-$P_{\boldsymbol{\theta}}$) instances when q is large — an effect no global learning rate can replicate.[1]

[1] Adaptive and momentum optimizers such as Adam (Kingma & Ba, 2014) rescale updates per *parameter* or average them over steps, not per *example*; they cannot substitute for $P_{\boldsymbol{\theta}}^{-q}$.

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

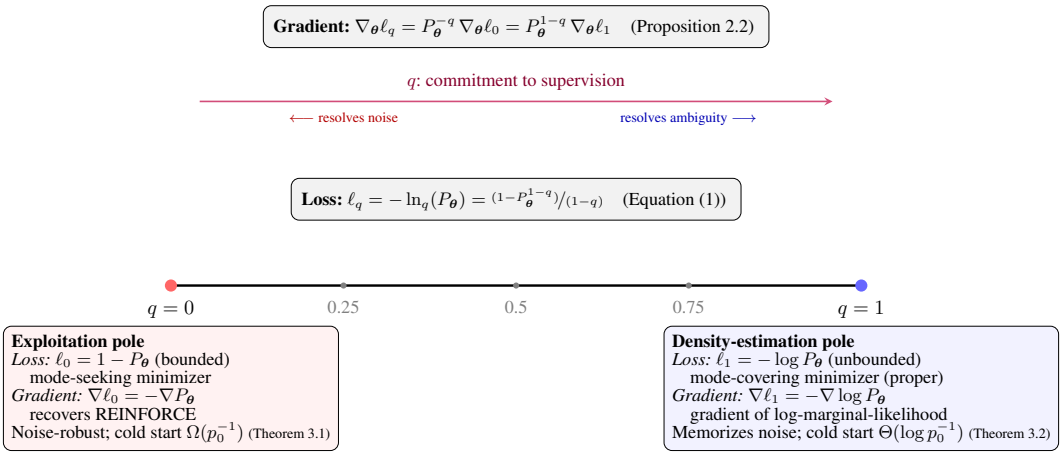


Figure 1: The J_Q loss family is a continuum between exploitation ($q = 0$) and density estimation ($q = 1$) losses (poles at either end of the axis below); correspondingly, commitment is the induced gradient amplification ($P_{\boldsymbol{\theta}}^{-q}$; top arrow). High q resolves ambiguity (fast cold-start escape) but also memorizes noise; low q resolves noise (robust filtering) but cannot escape cold start. p_0 denotes initial success probability; convergence results assume bounded score (Section 3).

The **commitment** q thus acts as a training-time counterpart of inference temperature: q decides which examples the gradient pulls on (Equation (2)), and J_Q's minimizer is the q-tempered data distribution (Theorem 2.1). High q enables fast cold-start escape in $\Theta(\log({}^1/_{p_0}))$ time (Theorem 3.2) but memorizes label errors (Proposition D.3); low q is noise-robust but escape slows to $\Omega({}^1/_{p_0})$ (Theorem 3.1); both halves are confirmed empirically (Section 5). This explains why SFT-then-RLVR succeeds: SFT on annotated rationales optimizes an ELBO of J_1, which carries the same inverse-probability amplification and so escapes cold start; switching to RLVR (q=0) afterward filters noisy supervision. It also suggests that an intermediate q can cold-start a reasoning model under J_Q directly, without SFT. Since $P_{\boldsymbol{\theta}}$ is intractable, we estimate $\nabla_{\boldsymbol{\theta}} J_Q$ by two Monte Carlo factorizations with different stability (Section 4).

Contributions. **(1) The J_Q loss family** (Sections 2 to 3). J_Q interpolates between a bounded, noise-robust loss at q=0 and an unbounded, mode-covering loss at q=1. Its categorical minimizer is the escort $\theta_j^* \propto \alpha_j^{1/q}$ (Theorem 2.1); J_Q also enforces a dispersion penalty across examples (Proposition C.1). The shared $P_{\boldsymbol{\theta}}^{-q}$ amplification separates escape speed: $\Omega({}^1/_{p_0})$ at q=0 vs. $\Theta(\log {}^1/_{p_0})$ at q=1 (Theorems 3.1 and 3.2). **(2) Two gradient estimators: GARL and PAFT** (Section 4). The dual factorization yields **Gradient-Amplified RL** (prior sampling, amplified by $P_{\boldsymbol{\theta}}^{-q}$; generalizes RB-REINFORCE (q=0; Zhou et al., 2026) and IWAE (q=1; Burda et al., 2015)) and **Posterior-Attenuated Fine-Tuning** (posterior resampling, attenuated by $P_{\boldsymbol{\theta}}^{1-q}$; generalizes the EM gradient update (q=1; Dempster et al., 1977; Phan et al., 2023)). Both have bias $O({}^q/_{MP_{\boldsymbol{\theta}}^q})$; GARL has lower variance, but PAFT only downweights gradients, and can have better training stability (Section G). **(3) Empirical validation** (Section 5). On FinQA, HotPotQA, and MuSiQue with exact-match training rewards: cold-start GARL at sufficiently high q escapes (Table 1 and Figure 3) where GRPO fails entirely, for Qwen 3 and Gemma 4 models, and under 20% label corruption on FinQA q=0.75 degrades 3–5× less than q=1, the predicted dual of the escape asymmetry (Section 5.3). Without task prompts, cold-start GARL (q=0.75, Qwen 3 0.6B) also matches or exceeds prompted GRPO on `m@16` and `p@16` on all three benchmarks, whether GRPO uses the matched (exact-match) reward or a reshaped one, which requires task knowledge (Section 5.4). With task prompts, PAFT (q=0.75) outperforms matched GRPO, most on HotPotQA (47.9 vs. 34.0 `m@16`; Section 5.5).

2 Setup and the J_Q Loss Family

We consider supervised conditional generation with latent reasoning trajectories: an autoregressive language model $p_{\boldsymbol{\theta}}$ with parameters $\boldsymbol{\theta} \in \mathbb{R}^d$, trained on a dataset $\mathcal{D}$ of input-output pairs $(\mathbf{x}^*, \mathbf{y}^*)$.

Given input $\mathbf{x}$, the model samples an unannotated latent rationale $\mathbf{z}$ from $p_{\boldsymbol{\theta}}(\cdot \mid \mathbf{x})$ then an output $\hat{\mathbf{y}} \sim p_{\boldsymbol{\theta}}(\cdot \mid \mathbf{x}, \mathbf{z})$, inducing the marginal $p_{\boldsymbol{\theta}}(\mathbf{y} \mid \mathbf{x}) = \sum_{\mathbf{z}} p_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{y} \mid \mathbf{x})$. The latent $\mathbf{z}$ may be a chain of thought (Wei et al., 2022), proof trace, program, etc.; we treat it as an *operational* latent mediating the output distribution.

Success probability and endpoint losses. For each supervised example, the **success probability** is $P_{\boldsymbol{\theta}} \triangleq p_{\boldsymbol{\theta}}(\mathbf{y}^* \mid \mathbf{x}^*)$. We define the **exploitation loss** $J_0(\boldsymbol{\theta}) \triangleq \mathbb{E}_{\mathcal{D}}[1 - P_{\boldsymbol{\theta}}]$ and **density-estimation loss** $J_1(\boldsymbol{\theta}) \triangleq \mathbb{E}_{\mathcal{D}}[-\log P_{\boldsymbol{\theta}}]$, both minimized at $P_{\boldsymbol{\theta}} = 1$. Under exact-match supervision $R(\hat{\mathbf{y}}, \mathbf{y}^*) = \mathbb{I}(\hat{\mathbf{y}} = \mathbf{y}^*)$, $J_0 = 1 - \mathbb{E}_{\mathcal{D}}[\text{reward}]$ (Proposition B.1), so minimizing J_0 maximizes expected reward: the objective that REINFORCE and its modern variants (Williams, 1992; Kool et al., 2019; Shao et al., 2024) ascend with different gradient estimators. Conversely, J_1 is the negative log-marginal likelihood over latent rationales, defined from $(\mathbf{x}^*, \mathbf{y}^*)$ pairs alone; for any annotation distribution over rationales, supervised fine-tuning (SFT) on annotated rationales optimizes an evidence lower bound (ELBO) on that same log-marginal likelihood (Proposition B.2).

The J_Q family. The Tsallis q-logarithm (Tsallis, 1988), $\ln_q(u) = {}^{(u^{1-q}-1)}/_{(1-q)}$ for $u \in (0, 1]$ with $\ln_1(u) \triangleq \ln u$, defines the per-example loss and dataset objective

$$\ell_q(\boldsymbol{\theta}; \mathbf{x}^*, \mathbf{y}^*) \triangleq -\ln_q P_{\boldsymbol{\theta}} = \frac{1 - P_{\boldsymbol{\theta}}^{1-q}}{1-q}, \qquad J_Q(\boldsymbol{\theta}, q) = \mathop{\mathbb{E}}_{(\mathbf{x}^*, \mathbf{y}^*) \sim \mathcal{D}}[\ell_q(\boldsymbol{\theta}; \mathbf{x}^*, \mathbf{y}^*)], \tag{1}$$

recovering $J_Q(\boldsymbol{\theta}, 0) = J_0$ and $J_Q(\boldsymbol{\theta}, 1) = J_1$.[2] At $q < 1$ the per-example loss is bounded and noise-robust; at $q = 1$ it is unbounded and the model fits the training distribution exactly, including label errors. Strict convexity of $-\ln_q$ for $q > 0$ gives $J_Q \geq -\ln_q(\mathbb{E}_{\mathcal{D}}[P_{\boldsymbol{\theta}}])$: J_Q penalizes non-uniform success across examples (*dispersion penalty*, Proposition C.1).

q reflects training-time commitment. Just as inference temperature controls output spread at decoding time, q controls it at training time: ℓ_q penalizes non-uniform $\boldsymbol{\theta}$ more when q increases. To illustrate this point, we consider K-category models with empirical frequencies $\alpha_k > 0$. J_Q's minimizer for such models is the **escort distribution** (Beck & Schlögl, 1993) of order ${}^1/_q$:

Theorem 2.1. *[Minimizers of J_Q in the categorical model] For $q \in (0, 1]$, the unique minimizer of $J_Q(\theta, q) = \sum_j \alpha_k(-\ln_q \theta_k)$ over $\theta \in \Delta_K$ is $\theta_k^*(q) = \frac{\alpha_k^{1/q}}{\sum_{k'} \alpha_{k'}^{1/q}}$. For $q = 0$, any vertex e_j with $j \in \operatorname{argmax}_k \alpha_k$ is optimal.*

The escort interpolates continuously from full coverage (q=1: $\theta^* = \alpha$) to pure mode-seeking ($q \to 0$), with q=1 the unique strictly proper scoring rule in J_Q (Corollary C.3; proofs in Section C).

Gradient geometry. All members of J_Q share one per-example gradient direction, factoring through either the exploitation loss endpoint $\nabla_{\boldsymbol{\theta}} \ell_0$ or the density-estimation loss endpoint $\nabla_{\boldsymbol{\theta}} \ell_1$:

Proposition 2.2 (Gradient geometry and dual factorization)**.** *For any fixed supervised example $(\mathbf{x}^*, \mathbf{y}^*)$ with $P_{\boldsymbol{\theta}} > 0$ and any $q \in [0, 1]$,*

$$\nabla_{\boldsymbol{\theta}} \ell_q(\boldsymbol{\theta}; \mathbf{x}^*, \mathbf{y}^*) \;=\; \underbrace{P_{\boldsymbol{\theta}}^{-q}}_{\textit{amplify}} \nabla_{\boldsymbol{\theta}} \ell_0(\boldsymbol{\theta}; \mathbf{x}^*, \mathbf{y}^*) \;=\; \underbrace{P_{\boldsymbol{\theta}}^{1-q}}_{\textit{attenuate}} \nabla_{\boldsymbol{\theta}} \ell_1(\boldsymbol{\theta}; \mathbf{x}^*, \mathbf{y}^*). \tag{2}$$

Proof. By the chain rule and $\frac{d}{du} \ln_q(u) = u^{-q}$: $\nabla_{\boldsymbol{\theta}} \ell_q = -P_{\boldsymbol{\theta}}^{-q} \nabla_{\boldsymbol{\theta}} P_{\boldsymbol{\theta}} = P_{\boldsymbol{\theta}}^{-q} \nabla_{\boldsymbol{\theta}} \ell_0$. Since $\nabla_{\boldsymbol{\theta}} \ell_0 = -\nabla_{\boldsymbol{\theta}} P_{\boldsymbol{\theta}} = P_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\theta}} \ell_1$, the second equality follows. □

The amplification $P_{\boldsymbol{\theta}}^{-q} \in [1, \infty)$ controls both cold-start escape speed (Section 3) and ratio-estimator bias (Section 4); the RL factorization motivates GARL (Section 4.1), the FT factorization motivates PAFT (Section 4.2).

[2] For observed labels, ℓ_q reduces to known losses for label-noise robustness; see Section 6.

3 COMMITMENT DYNAMICS UNDER GRADIENT FLOW

Under gradient flow, escape from a cold start ($p_0 = P_{\boldsymbol{\theta}(0)} \ll 1$) takes $\Omega(1/p_0)$ time at the exploitation pole (q=0) but only $\Theta(\log(1/p_0))$ at the density-estimation pole (q=1). This exponential separation in $1/p_0$ is governed by the amplification factor $P_{\boldsymbol{\theta}}^{-q}$ and the dynamics $\dot{p} = p^{2-q}\|s(\boldsymbol{\theta})\|^2$.[3]

Dynamics of the success probability. We study gradient flow $\dot{\boldsymbol{\theta}} = -\nabla_{\boldsymbol{\theta}}\ell(\boldsymbol{\theta})$ (Su et al., 2016), which isolates closed-form rates from step-size effects without requiring convexity ($\dot{p} \geq 0$ always). For a single example with score $s(\boldsymbol{\theta}) \triangleq \nabla_{\boldsymbol{\theta}} \log P_{\boldsymbol{\theta}}$, Proposition 2.2 gives

$$\dot{p} = \nabla_{\boldsymbol{\theta}} P_{\boldsymbol{\theta}} \cdot \dot{\boldsymbol{\theta}} = P_{\boldsymbol{\theta}}^{-q} \, \|\nabla_{\boldsymbol{\theta}} P_{\boldsymbol{\theta}}\|^2 = p^{2-q} \, \|s(\boldsymbol{\theta})\|^2, \tag{3}$$

where q's entire effect on convergence is captured by the exponent $2 - q$ ($\|s\|^2$ is q-independent).

Why q matters at cold start. For $p_0 \triangleq p(0) \ll 1$ and approximately constant $\|s\|$, the time to reach target δ is $T \sim \int_{p_0}^{\delta} u^{-(2-q)}\, du$. The exponent $2 - q$ sets the divergence rate as $p_0 \to 0$: at $q = 0$, $\int u^{-2}\, du \sim p_0^{-1}$; at $q = 1$, $\int u^{-1}\, du \sim \log(1/p_0)$.

Cold-start escape rates. We present the separation in two results: an $\Omega(\cdot)$ bound assuming that score is upper-bounded (training with low-q is *provably slow*), then a matching $\Theta(\cdot)$ rate assuming that the score is also lower-bounded.

Theorem 3.1. *[Exploitation is provably slow] Let $\boldsymbol{\theta} \in \mathbb{R}^d$ parameterize any differentiable model. Consider gradient flow on $\ell_q(\boldsymbol{\theta}) = -\ln_q(P_{\boldsymbol{\theta}})$, starting from $p_0 = P_{\boldsymbol{\theta}(0)} \in (0, 1/2)$ with fixed target $\delta \in (p_0, 1/2]$. Suppose $\|s(\boldsymbol{\theta}(t))\| \leq C \in \mathbb{R}$. Then as $p_0 \to 0$:*

$$T_q(p_0, \delta) = \Omega\left(\frac{p_0^{-(1-q)}}{1-q}\right) \text{ for } q \in [0, 1),$$

$$T_1(p_0, \delta) = \Omega\left(\log \frac{1}{p_0}\right).$$

The proof integrates $\dot{p} \leq C^2 p^{2-q}$ (Section D). $\|s\| \leq C$ is a common regularity assumption (verified in closed form for the scalar sigmoid in Section D.1), so the exploitation pole has escape time $\Omega(1/p_0)$.

Theorem 3.2. *[Tight cold-start escape rates] Under the same setup as Theorem 3.1, suppose additionally that $\|s(\boldsymbol{\theta}(t))\| \geq c > 0$ throughout the trajectory. Then as $p_0 \to 0$,*

$$T_q(p_0, \delta) = \Theta\left(\tfrac{p_0^{-(1-q)}}{1-q}\right) \text{ for } q \in [0, 1), \qquad T_1(p_0, \delta) = \Theta\left(\log \tfrac{1}{p_0}\right),$$

and consequently $T_q(p_0,\delta)/T_{q'}(p_0,\delta) \to \infty$ for any $q < q' \leq 1$.

The lower bound $\dot{p} \geq c^2 p^{2-q}$ gives the matching upper bound via the same integration (Section D). For a single example, all ℓ_q share one gradient direction (Proposition 2.2), so their flows trace the same path at different speeds and see the same scores; along it $T_q = \int_{p_0}^{\delta} du/(u^{2-q}\|s\|^2)$, so $T_q \geq T_{q'}$ for $q < q'$ even where $\|s\| \geq c$ fails; only the Θ rates need it. Restricting the target to $\delta \leq 1/2$ avoids $p \to 1$, where softmax scores vanish.

Noise fitting is symmetric. The same machinery gives an exact dual: under the canonical sigmoid model, growing noise contamination from $\tilde{p}_0$ to a fixed target takes $T_q^{\text{noise}}(\tilde{p}_0) = \Theta(\tilde{p}_0^{-(1-q)}/((1-q)\epsilon))$ for $q \in (0, 1)$ and $\Theta(\log(1/\tilde{p}_0)/\epsilon)$ at q=1 (Proposition D.3 in Section D.5, diverging at q=0) — matching cold-start escape's exponent in the small starting probability, with ϵ the only additional rate factor. So $P_{\boldsymbol{\theta}}^{-q}$ accelerates clean and corrupted commitment by the same factor, and SFT-then-RL (Ouyang et al., 2022; DeepSeek-AI, 2025; Chu et al., 2025) becomes a hard q=1 $\to$ q=0 switch: SFT escapes cold start via $P_{\boldsymbol{\theta}}^{-1}$ amplification; RL afterwards halts noise commitment ($T_q^{\text{noise}} \to \infty$ at q=0). The reverse order gets neither; J_Q replaces the hard switch with a smooth interpolation.

[3] We track the single-example success probability under continuous-time gradient flow. The main cold-start theoretical result is *conservatively* stylized: interference between examples and discrete minibatch steps cannot beat the single-example speed limit (Section D.4), and Section D.1 gives exact rates for a scalar sigmoid model.

4 GRADIENT ESTIMATORS FOR J_Q

The marginal $P_{\boldsymbol{\theta}} = \sum_{\mathbf{z}\in\mathcal{Z}} p_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{y}^* \mid \mathbf{x}^*)$ in $\nabla_{\boldsymbol{\theta}} \ell_q$ is intractable, so we estimate the gradient by Monte Carlo. The dual factorization (Proposition 2.2) yields two natural estimators:

- **GARL** (Section 4.1): sample from the prior $p_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathbf{x}^*)$, estimate $\nabla_{\boldsymbol{\theta}} \ell_0$ and $P_{\boldsymbol{\theta}}$ from the same samples, amplify by $(\bar{w}_M)^{-q}$ (a plug-in estimator of the amplification factor $P_{\boldsymbol{\theta}}^{-q}$).
- **PAFT** (Section 4.2): approximately sample from the posterior $p_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathbf{x}^*, \mathbf{y}^*)$, estimate $\nabla_{\boldsymbol{\theta}} \ell_1$ via teacher forcing, attenuate by $(\bar{w}_M)^{1-q}$ (estimating $P_{\boldsymbol{\theta}}^{1-q}$).

Both are drop-in replacements for RB-RLOO (Zhou et al., 2026) at the same rollout budget and need no extra forward passes: GARL adds an $O(M)$ scalar reweighting, and PAFT one categorical resample before teacher forcing on already-generated tokens.

4.1 GARL: GRADIENT-AMPLIFIED RL

A plug-in Monte Carlo estimator. Fix a supervised example $(\mathbf{x}^*, \mathbf{y}^*)$ and draw M i.i.d. latent trajectories $\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(M)} \sim p_{\boldsymbol{\theta}}(\cdot \mid \mathbf{x}^*)$. Define the per-sample likelihood weight and gradient contribution:

$$w_m \triangleq p_{\boldsymbol{\theta}}(\mathbf{y}^* \mid \mathbf{x}^*, \mathbf{z}^{(m)}), \qquad g_m \triangleq -w_m \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{z}^{(m)}, \mathbf{y}^* \mid \mathbf{x}^*), \tag{4}$$

with empirical means $\bar{w}_M \triangleq \frac{1}{M}\sum_m w_m$ and $\bar{g}_M \triangleq \frac{1}{M}\sum_m g_m$. By the log-trick,

$$\mathbb{E}[\bar{w}_M] = P_{\boldsymbol{\theta}}, \qquad \mathbb{E}[\bar{g}_M] = -\sum_{\mathbf{z}} \nabla_{\boldsymbol{\theta}} p_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{y}^* \mid \mathbf{x}^*) = -\nabla_{\boldsymbol{\theta}} P_{\boldsymbol{\theta}} = \nabla_{\boldsymbol{\theta}} \ell_0. \tag{5}$$

Plugging these into the RL factorization of Proposition 2.2 yields the plug-in estimator

$$\hat{\nabla}_{\boldsymbol{\theta}} \ell_q(q, \boldsymbol{\theta}; \mathbf{x}^*, \mathbf{y}^*, M) \triangleq \frac{\bar{g}_M}{(\bar{w}_M)^q}. \tag{6}$$

The dataset-level estimator of $\nabla_{\boldsymbol{\theta}} J_Q$ averages Equation (6) over a minibatch: GARL amplifies the RL gradient $\bar{g}_M$ by the plug-in estimate $(\bar{w}_M)^{-q}$ of $P_{\boldsymbol{\theta}}^{-q}$. At the endpoints, GARL recovers RB-REINFORCE (q=0; Zhou et al., 2026) and the IWAE gradient estimator (q=1; Burda et al., 2015); see Section E.2.

Consistency and finite-sample bias. Equation (6) is a ratio estimator: it reuses the same samples in numerator and denominator, so it is biased at finite M even though $\bar{w}_M$ and $\bar{g}_M$ are individually unbiased.[4]

Theorem 4.1. *[Consistency and bias expansion] Fix a supervised example $(\mathbf{x}^*, \mathbf{y}^*)$ and assume:*

1. *$P_{\boldsymbol{\theta}} > 0$;*
2. *$\mathbb{E}[\|g_m\|^2] < \infty$;*
3. *$w_m \geq \epsilon$ a.s. for some $\epsilon > 0$.*

Then for any fixed $q \in [0, 1]$, the estimator is consistent: $\hat{\nabla}_{\boldsymbol{\theta}} \ell_q \xrightarrow{a.s.} \nabla_{\boldsymbol{\theta}} \ell_q$ as $M \to \infty$. Moreover, the leading-order bias is

$$\mathbb{E}\Big[\hat{\nabla}_{\boldsymbol{\theta}} \ell_q\Big] - \nabla_{\boldsymbol{\theta}} \ell_q \;=\; \frac{q}{M P_{\boldsymbol{\theta}}^{q+1}} \big[\tfrac{q+1}{2} \nabla_{\boldsymbol{\theta}} \ell_1 \,\mathbf{Var}(w_m) - \mathbf{Cov}(g_m, w_m)\big] + O(M^{-2}) \quad \text{as } M \to \infty. \tag{7}$$

Under additionally bounded marginal and per-trajectory scores ($\|\nabla_{\boldsymbol{\theta}} \log P_{\boldsymbol{\theta}}\| \leq C$, $\|\nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{y}^* \mid \mathbf{x}^*)\| \leq C'$), the bracketed term is $O(P_{\boldsymbol{\theta}})$, so the bias simplifies to $O(q/M P_{\boldsymbol{\theta}}^q)$.

At q=0 the bias vanishes *exactly* for all M: the estimator reduces to the unbiased sample mean $\bar{g}_M$ (Equation (5)). The proof is a delta-method expansion of $\bar{g}_M / \bar{w}_M^q$ around $(P_{\boldsymbol{\theta}}, \nabla \ell_0)$ (Section E). The J_Q-specific feature is the joint dependence on q and $P_{\boldsymbol{\theta}}$: the same $P_{\boldsymbol{\theta}}^{-q}$ that enables fast escape (Theorems 3.1 and 3.2) degrades estimator quality at the same rate, predicting that intermediate q outperforms both endpoints — confirmed in Section 5.

[4] Assumptions 1–2 are standard regularity. Assumption 3 controls the ratio-estimator denominator at fixed $\boldsymbol{\theta}$: for autoregressive softmax models, $w_m = \prod_{t=1}^{T} p_{\boldsymbol{\theta}}(y_t^* \mid \cdot) \geq \epsilon_0^T$ for some $\epsilon_0 > 0$. The bound is not uniform over training, and may also shrink as $P_{\boldsymbol{\theta}} \to 0$.

Control variate. We apply the standard leave-one-out control variate (Kool et al., 2019) to GARL's score-function term, centering the per-sample coefficient $w_m/(\bar{w}_M)^q$ against $(\bar{w}_{\neg m})^{1-q}$ where $\bar{w}_{\neg m} \triangleq \frac{1}{M-1}\sum_{j\neq m} w_j$ (full RLOO estimator and derivation in Section E.1). The control variate preserves the bias of Theorem 4.1 (Proposition E.1). At $q = 0$ this recovers the Rao–Blackwellized RLOO of Zhou et al. (2026); at $q = 1$ the centered weight becomes $w_m/\bar{w}_M - 1$, a self-normalizing baseline. Pseudocode is in Algorithm 1.

4.2 PAFT: Posterior-Attenuated Fine-Tuning

GARL samples from the prior and amplifies by $P_{\boldsymbol{\theta}}^{-q}$ — sometimes massively. The FT factorization (Equation (2)) offers an alternative: sample from the posterior $p_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathbf{x}^*, \mathbf{y}^*)$ — where rationales already agree with $\mathbf{y}^*$ — and attenuate by $P_{\boldsymbol{\theta}}^{1-q} \in [0, 1]$.

Posterior form of the gradient. Expanding $\nabla_{\boldsymbol{\theta}}\ell_1 = -\nabla_{\boldsymbol{\theta}} \log P_{\boldsymbol{\theta}}$ as a posterior expectation:

$$\nabla_{\boldsymbol{\theta}}\ell_q = -P_{\boldsymbol{\theta}}^{1-q} \cdot \mathop{\mathbb{E}}_{\mathbf{z}\sim p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x}^*,\mathbf{y}^*)} [\nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{y}^* \mid \mathbf{x}^*)]. \tag{8}$$

Each sample gradient is standard SFT (teacher forcing) on a *semantically coherent* (input, rationale, answer) triple: the rationale is posterior-weighted toward agreement with $\mathbf{y}^*$.

Approximate posterior sampling. The posterior is intractable for autoregressive models. We use importance resampling (IR; Rubin, 1988), which reuses GARL's pool and weights: resample K indices $r_1, \ldots, r_K \in \{1, \ldots, M\}$ with replacement, with r_k drawn proportional to w_{r_k}. The PAFT estimator is

$$\hat{\nabla}_{\text{PAFT}} = -(\bar{w}_M)^{1-q} \cdot \frac{1}{K}\sum_{k=1}^{K} \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{z}^{(r_k)}, \mathbf{y}^* \mid \mathbf{x}^*). \tag{9}$$

At $q = 1$, the attenuation vanishes ($(\bar{w}_M)^{1-q} = 1$) and PAFT recovers the EM gradient update — the M-step gradient evaluated over E-step posterior samples (Dempster et al., 1977; Phan et al., 2023), i.e., SFT with self-generated rationales, the zero-gap case of Proposition B.2; Section E.2 lists all endpoint reductions.

Bias and variance. Importance resampling preserves the gradient mean: PAFT inherits GARL's leading bias expansion (Proposition E.3), which under the bounded-score conditions of Theorem 4.1 simplifies to $O(q/MP_{\boldsymbol{\theta}}^q)$, and has strictly higher variance by the law of total variance (Proposition E.4; full derivations in Section E.3).

Yet PAFT can produce better training dynamics: GARL's lower variance comes from mixing bad rationales with small weights, while PAFT excludes them before the gradient is formed, preserving the FT endpoint's semantic coherence. With prompted inputs, PAFT degrades more gradually than GARL despite its higher variance (Section G); see Algorithm 2.

5 Empirical Validation

We validate the theoretical predictions and the empirical effectiveness of GARL and PAFT on three reasoning benchmarks — FinQA (Chen et al., 2021), HotPotQA (Yang et al., 2018), and MuSiQue (Trivedi et al., 2022) — using Qwen 3 and Gemma 4 models (Yang et al., 2025; Team et al., 2026).

5.1 Experimental setup

Our experiments target the no-rationale-annotation regime, where SFT cannot apply, so we do not compare against SFT-then-RLVR; we evaluate fixed-q GARL and PAFT and leave annealed q to future work. We organize the empirical findings around four research questions: **RQ1** — does increasing q help escape cold start? **RQ2** — does decreasing q help resist label noise, as the theory predicts? **RQ3** — without task prompts, how does cold-start GARL compare with GRPO trained on prompted inputs? **RQ4** — do J_Q estimators help with prompted inputs, the standard RLVR setting?

Scenarios and cold-start remedies. *Cold start* uses linearized $(\mathbf{x}^*, \mathbf{y}^*)$ pairs with no task description and no formatting instructions; the model must discover both how to solve the problem and how to format the answer, and initial $P_{\boldsymbol{\theta}}$ is very low. We consider two standard remedies:

- **Prompting** raises the initial success probability p_0 (Wei et al., 2022; Kojima et al., 2022; DeepSeek-AI, 2025): inputs carry task descriptions and answer-formatting instructions (Figure 2), so the un-adapted model occasionally answers correctly and reward is not sparse.
- **Reward shaping** densifies the reward (Ng et al., 1999): our *reshaped* reward fires whenever the last `<answer>` block equals $\mathbf{y}^*$,[5] whatever else the response contains.

Both presuppose *a priori* task knowledge: prompting needs task descriptions and formatting instructions, and shaping needs a task-specific notion of partial success. Our methods need neither remedy: GARL escapes cold start by amplifying the same exact-match objective on bare $(\mathbf{x}^*, \mathbf{y}^*)$ pairs, with no prompt and no shaped reward.

Datasets, methods, and evaluation. GRPO, GARL, and PAFT all use $M = 32$ rollouts per prompt during training for Qwen 3 0.6B and Gemma 4 E4B, and $M = 16$ for Qwen 3 8B. All methods use 16 samples per prompt at evaluation. GARL (Algorithm 1) uses the RLOO variance reduction (Equation (19)), with advantages normalized by M^q (Section E.4); PAFT (Algorithm 2) resamples $K = M$ trajectories from the same pool; it is not run at cold start, where $P_{\boldsymbol{\theta}}^{1-q} \approx 0$ suppresses its gradient. We evaluate $q \in \{0, 0.25, 0.5, 0.75, 1\}$ for Qwen 3 0.6B, $q \in \{0.25, 0.5, 0.75, 1\}$ for Gemma 4 E4B (q=0.25 on FinQA only), and $q \in \{0, 0.75, 0.85, 1\}$ for Qwen 3 8B. All methods train toward the exact-match success probability $P_{\boldsymbol{\theta}}$ (Section 2). GARL and PAFT teacher-force the canonical completion $\texttt{t}_{\text{end}}$`\n<answer>`$\mathbf{y}^*$`</answer>` after each sampled rationale, where $\texttt{t}_{\text{end}}$ indicates the thinking-end token; GRPO's *matched* reward is a binary reward that requires this completion, so its expected reward is $1 - J_0$. We also report prompted GRPO trained with both matched and reshaped reward functions; evaluation uses the reshaped reward for all methods. We report `p@1` (single-sample accuracy), `p@`k (best-of-k, rewards coverage), and `m@`k (majority vote over k samples (Wang et al., 2023)). Reported test numbers are taken from the checkpoint with highest validation `m@16`; unless otherwise marked with $\pm$, numbers are single-seed. Additional experiment setup details are in Section F.

5.2 RQ1: Does increasing q help escape cold start?

Cold start tests whether $P_{\boldsymbol{\theta}}^{-q}$ determines escape from a sparse-reward regime (Theorem 3.2).

Yes, across model families, but only above a critical q that depends on the model. GRPO, Rao–Blackwellized RLOO (q=0), and all $q \leq 0.5$ fail entirely on Qwen 3 0.6B; only $q \geq 0.75$ escapes. Rao–Blackwellization (Zhou et al., 2026) reduces variance but cannot accelerate escape: at q=0 the dynamics $\dot{p} = p^2\|s\|^2$ have no amplification. The sharp transition at $q = 0.75$ matches Theorem 3.1: the lower bound $\Omega(p_0^{-(1-q)})$ grows rapidly as q decreases, so the training budget sets a critical q below which escape fails. The transition is not specific to one model: it is equally sharp on Qwen 3 8B, where the threshold shifts to $q \geq 0.85$ (q=0.75 now fails), and on Gemma 4 E4B, where GRPO, and GARL at q=0.25 on FinQA, score exactly zero and every $q \geq 0.5$ escapes. The critical q moves with the model, consistent with the fixed-budget reading of Theorem 3.1, under which the initial success probability sets the threshold. For every model the best run sits below q=1: q=0.75 on Qwen 3 0.6B and Gemma 4 E4B, and q=0.85 on Qwen 3 8B, achieve higher `p@1` than q=1 on every benchmark. The reason may be attributed to both the escape-vs-bias tradeoff (Theorem 4.1) and the noise-fitting speed (Proposition D.3): q=1's stronger amplification escapes faster but yields higher-bias gradient estimates, and it also commits faster to any errors in the training labels (Section 5.3 isolates this effect by corrupting labels).

5.3 RQ2: Does decreasing q help resist label noise?

Proposition D.3 is the mirror image of the escape-time separation: the same $P_{\boldsymbol{\theta}}^{-q}$ amplification that lets high q commit quickly to correct supervision lets it commit quickly to corrupted supervision, so

[5] Our implementation for Qwen 3 experiments additionally check that the block appears after the thinking-end token $\texttt{t}_{\text{end}}$.

Table 1: Cold-start results for Qwen 3 0.6B and 8B (Yang et al., 2025) and Gemma 4 E4B (Team et al., 2026) ("—": not run). GRPO and GARL at low q score zero; the escape threshold is $q \geq 0.75$ at 0.6B, $q \geq 0.85$ at 8B, and $q \geq 0.5$ on Gemma 4 E4B FinQA (q=0.25 was not run on HotPotQA). Prompted GRPO with the matched or reshaped reward, and prompted PAFT, are cross-regime references (Sections 5.4 and 5.5). Best per model × benchmark × metric in bold; ±: standard deviation over 3 seeds (2 for reshaped-reward GRPO on HotPotQA), otherwise single-seed.

	FinQA			HotPotQA			MuSiQue		
Method	p@1	p@16	m@16	p@1	p@16	m@16	p@1	p@16	m@16
Qwen 3 0.6B (cold-start)									
GRPO	0	0	0	0	0	0	0	0	0
GRPO (prompted, matched)	20.6 ±2.0	48.5 ±0.7	27.8 ±1.1	29.7 ±0.6	56.8 ±1.6	34.0 ±0.7	12.9 ±1.2	35.7 ±1.9	15.4 ±0.4
GRPO (prompted, reshaped)	30.1 ±0.1	57.9 ±0.1	37.1 ±0.2	53.2 ±0.2	67.7 ±1.4	55.4 ±0.2	**28.9**	51.7	34.0
PAFT q=0.75 (prompted)	20.9	49.9	28.6	43.5	64.3	47.9	13.5	38.4	17.5
GARL q=0 (RB-RLOO)	0	0	0	0	0	0	0	0	0
GARL q=0.25	0	0	0	0	0	0	0	0	0
GARL q=0.5	0	0	0	0	0	0	0	0	0
GARL q=0.75	**30.5** ±0.3	**61.5** ±0.4	**38.6** ±0.6	**53.4** ±0.6	74.6 ±1.1	**57.4** ±0.9	27.5 ±1.0	**58.3** ±0.6	**35.6** ±1.4
GARL q=1	21.9	58.7	33.5	48.7	**75.5**	56.6	21.6	58.1	32.5
Qwen 3 8B (cold-start)									
GRPO	0	0	0	0	0	0	0	0	0
GRPO (prompted, matched)	18.7	26.2	19.6	34.9	50.5	39.6	26.7	51.9	31.1
GARL q=0	0	0	0	0	0	0	0	0	0
GARL q=0.75	0	0	0	0	0	0	0	0	0
GARL q=0.85	**45.0**	75.2	**52.9**	**64.8**	**81.5**	**68.6**	**58.7**	78.8	62.9
GARL q=1	38.4	**75.6**	50.1	61.6	81.4	67.9	57.1	**79.6**	**64.5**
Gemma 4 E4B (cold-start)									
GRPO	0	0	0	0	0	0	—	—	—
GRPO (prompted, matched)	18.8	25.3	19.8	61.6	69.7	62.6	—	—	—
GRPO (prompted, reshaped)	19.5	26.2	20.5	—	—	—	—	—	—
GARL q=0	0	0	0	0	0	0	—	—	—
GARL q=0.25	0	0	0	—	—	—	—	—	—
GARL q=0.5	43.7	63.0	46.7	63.3	75.7	64.3	—	—	—
GARL q=0.75	**45.2**	69.8	**50.2**	**65.5**	80.1	**67.4**	—	—	—
GARL q=1	33.8	**71.2**	46.1	60.0	**80.2**	64.6	—	—	—

Table 2: Robustness to 20% label noise on FinQA (Qwen 3 0.6B, cold-start GARL). Each setting is compared with its own clean run; Δ is the absolute drop with the relative drop in parentheses. All runs are single-seed except for GARL q=0.75 clean, where the numbers are taken from the 3-seed mean reported in Table 1.

	GARL q=1			GARL q=0.75		
Metric	clean	20% noise	Δ	clean	20% noise	Δ
p@1	21.9	15.6	−6.3 (−29%)	30.5	28.6	−1.9 (−6%)
p@16	58.7	51.4	−7.3 (−12%)	61.5	59.4	−2.1 (−3%)
m@16	33.5	27.3	−6.2 (−19%)	38.6	37.1	−1.5 (−4%)

at a fixed training budget lower q should fit label noise less. We test this on FinQA at cold start by corrupting the supervision directly (Table 2): for 20% of training examples we replace the answer with a random answer drawn from another training example (plausible, wrong) and retrain from the same cold start with GARL at q=1 and q=0.75, the two settings that escape at 0.6B.

Yes, the predicted asymmetry appears on every metric. At q=1 the corruption costs 6–7 points (12–29% relative); at q=0.75 it costs 1–2 points (3–6% relative), a 3–5× smaller degradation from identical labels, and the noised q=0.75 run still exceeds the clean q=1 run on all three metrics. Together with Table 1, this brackets the commitment axis from both ends: the ordering predicted by the theory holds for escape and for memorization, and an intermediate q keeps enough of each to train in a single stage from imperfect supervision.

5.4 RQ3: How does cold-start GARL compare with prompted GRPO?

GARL matches or exceeds prompted GRPO on m@16 and p@16 under both matched and reshaped rewards. Unprompted GRPO scores zero at cold start, so we compare against prompted GRPO (Table 1). On Qwen 3 0.6B, GARL without prompts, at q=0.75 (the q with the highest

validation m@16 on every benchmark), exceeds matched GRPO on every metric across all three benchmarks ($+10.8$ to $+23.4$ m@16). With the reshaped reward, prompted GRPO is much stronger (37.1, 55.4, 34.0 m@16 on FinQA, HotPotQA, MuSiQue), yet cold-start GARL still leads on m@16 ($+1.5$, $+2.0$, $+1.6$) and p@16 ($+3.6$, $+6.9$, $+6.6$) and is level on p@1 ($+0.4$, $+0.2$, -1.4); the reshaped-reward HotPotQA baseline has two seeds and the MuSiQue baseline one, and the MuSiQue m@16 margin is about one GARL seed standard deviation. Against matched-reward baselines, the only ones for the other models, GARL also leads on every metric on Qwen 3 8B (at both escaped q; $+28.3$ to $+33.4$ m@16) and on Gemma 4 E4B (at q=0.75, again the validation-selected q; $+30.4$ m@16 on FinQA, $+4.8$ on HotPotQA). Reward shaping does not account for GRPO's cold-start failure: the un-prompted 0.6B model starts at exactly zero validation accuracy even under the reshaped reward, and GARL at q=0, which uses the same teacher-forced signal as GARL at q=0.75 without amplification, also fails there.

5.5 RQ4: Do J_Q estimators help with prompted inputs?

We conduct prompted experiments with PAFT; while PAFT is not suitable for cold-start setups, the gradient attenuation nature makes it stable when $|P_{\boldsymbol{\theta}}^{-q}|$ can vary greatly over training examples.[6]

PAFT improves on the matched baseline, most clearly on HotPotQA. With prompted inputs, $P_{\boldsymbol{\theta}}$ is no longer negligible and GRPO already makes progress. PAFT at q=0.75 exceeds matched-reward GRPO, which trains toward the same success event, by 13.9 m@16 on HotPotQA (Table 1); on FinQA and MuSiQue the m@16 margins (0.8 and 2.1) are small, and the FinQA m@16 and p@1 gaps are within GRPO's seed standard deviation (PAFT is single-seed). Reshaped-reward GRPO scores higher on all three benchmarks, but its reward encodes task knowledge that PAFT does not use.

6 Related Work

For observed labels, the same loss appears, up to reparameterization, as the L_q-likelihood (Ferrari & Yang, 2010), generalized cross-entropy (Zhang & Sabuncu, 2018), and α-loss (Sypherd et al., 2022), which target robustness to label noise. We apply the family to a marginal over latent rationales, where it connects RL and maximum marginal likelihood. Prior connections between RL and maximum likelihood (Levine, 2018; Norouzi et al., 2016; Guu et al., 2017) treat them as distinct frameworks, and latent-rationale methods sit at one endpoint: RLVR (Shao et al., 2024; DeepSeek-AI, 2025) and RB-REINFORCE (Zhou et al., 2026) at q=0, STaR (Zelikman et al., 2022) and TRICE (Phan et al., 2023) at q=1. Recently, Tajwar et al. (2026) interpolate between RL and maximum likelihood by truncating $\log p$, but their estimator is zero when no sample succeeds, as at cold start, and Wang et al. (2026) apply ℓ_q per token in SFT. On the theory side, exact policy gradient for tabular softmax policies converges at an $O(1/t)$ rate whose constants depend on the initialization, and can slow down near suboptimal deterministic policies (Mei et al., 2020a;b); Theorem 3.1 lower-bounds the gradient-flow escape time at q=0 by $\Omega(1/p_0)$ for any differentiable model with a bounded score. A full survey is in Section A.

7 Conclusion

J_Q smooths SFT-then-RLVR into a single commitment parameter q, recovering the pipeline as a stepwise q=1 $\rightarrow$ q=0 schedule. Its dual factorization yields GARL, whose prior-sampling amplification breaches GRPO's $\Omega(1/p_0)$ cold-start bottleneck, and PAFT, whose posterior-sampling attenuation connects J_Q to EM and to SFT. We find both methods compare favorably against GRPO and RB-RLOO, especially in cold-start scenarios. We discuss limitations and future directions in Section I.

AI use statement

In this work, we used generative AI tools for [formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, implement methods,

[6] In preliminary experiments, GARL is performant with prompted inputs but can be unstable (Section G). We leave application of RL training stabilization techniques on GARL as future work.

clean and reformat dataset]. We have not used generative AI tools for [help develop theoretical models or conceptual frameworks, propose or refine hypotheses, design or provide feedback on research methodology or experiments, support qualitative and thematic data analysis, interpret results], and [generate synthetic data sets, assist with translation] are not applicable to this work. Additionally, we used generative AI tools for [create or modify scientific figures or images, create or edit software code, summarize or analyse existing literature, discover research topics or identify gaps, edit a research paper to improve readability]. We have reviewed all AI-assisted work, including generated code and manuscript edits. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Christian Beck and Friedrich Schlögl. *Thermodynamics of Chaotic Systems: An Introduction*. Cambridge Nonlinear Science Series. Cambridge University Press, 1993.

Yuri Burda, Roger Baker Grosse, and Ruslan Salakhutdinov. Importance weighted autoencoders. volume abs/1509.00519, 2015. URL https://api.semanticscholar.org/CorpusID:11383178.

Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Reema Moussa, Matt Beane, Ting-Hao Huang, Bryan Routledge, and William Yang Wang. FinQA: A dataset of numerical reasoning over financial data. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 3697–3711, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.300. URL https://aclanthology.org/2021.emnlp-main.300/.

Tianzhe Chu, Yuexiang Zhai, Jihan Yang, Shengbang Tong, Saining Xie, Dale Schuurmans, Quoc V Le, Sergey Levine, and Yi Ma. SFT memorizes, RL generalizes: A comparative study of foundation model post-training. 2025. URL https://openreview.net/forum?id=dYur3yabMj.

DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. 2025. URL https://arxiv.org/abs/2501.12948.

AP Dempster, NM Laird, and DB Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 1–38, 1977.

Nan Ding and Radu Soricut. Cold-start reinforcement learning with softmax policy gradient. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 2814–2823, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Davide Ferrari and Yuhong Yang. Maximum L_q-likelihood estimation. *The Annals of Statistics*, 38 (2):753–783, 2010.

Kelvin Guu, Panupong Pasupat, Evan Liu, and Percy Liang. From language to programs: Bridging reinforcement learning and maximum marginal likelihood. In Regina Barzilay and Min-Yen Kan (eds.), *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1051–1062, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1097. URL https://aclanthology.org/P17-1097/.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. volume abs/1412.6980, 2014. URL https://api.semanticscholar.org/CorpusID:6628106.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Wouter Kool, Herke van Hoof, and Max Welling. Buy 4 REINFORCE samples, get a baseline for free! 2019. URL https://openreview.net/forum?id=r1lgTGL5DE.

Kyungjae Lee, Sungjoon Choi, and Songhwai Oh. Sparse markov decision processes with causal sparse tsallis entropy regularization for reinforcement learning. *IEEE Robotics and Automation Letters*, 3(3):1466–1473, 2018. doi: 10.1109/LRA.2018.2800085.

Sergey Levine. Reinforcement learning and control as probabilistic inference: Tutorial and review. *ArXiv*, abs/1805.00909, 2018. URL https://api.semanticscholar.org/CorpusID:19077536.

Yingzhen Li and Richard E Turner. Rényi divergence variational inference. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper_files/paper/2016/file/7750ca3559e5b8e1f44210283368fc16-Paper.pdf.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=5PAF7PAY2Y.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Jincheng Mei, Chenjun Xiao, Bo Dai, Lihong Li, Csaba Szepesvári, and Dale Schuurmans. Escaping the gravitational pull of softmax. In *Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*, 2020a. URL https://papers.nips.cc/paper/2020/hash/f1cf2a082126bf02de0b307778ce73a7-Abstract.html.

Jincheng Mei, Chenjun Xiao, Csaba Szepesvári, and Dale Schuurmans. On the global convergence rates of softmax policy gradient methods. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org, 2020b.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. s1: Simple test-time scaling. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 20275–20321, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.1025. URL https://aclanthology.org/2025.emnlp-main.1025/.

Ofir Nachum, Yinlam Chow, and Mohammad Ghavamzadeh. Path consistency learning in tsallis entropy regularized mdps. *ArXiv*, abs/1802.03501, 2018. URL https://api.semanticscholar.org/CorpusID:3653343.

Radford M. Neal and Geoffrey E. Hinton. *A View of the Em Algorithm that Justifies Incremental, Sparse, and other Variants*, pp. 355–368. Springer Netherlands, Dordrecht, 1998. ISBN 978-94-011-5014-9. doi: 10.1007/978-94-011-5014-9_12. URL https://doi.org/10.1007/978-94-011-5014-9_12.

Andrew Y. Ng, Daishi Harada, and Stuart J. Russell. Policy invariance under reward transformations: Theory and application to reward shaping. In *Proceedings of the Sixteenth International Conference on Machine Learning*, ICML '99, pp. 278–287, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers Inc. ISBN 1558606122.

Mohammad Norouzi, Samy Bengio, Zhifeng Chen, Navdeep Jaitly, Mike Schuster, Yonghui Wu, and Dale Schuurmans. Reward augmented maximum likelihood for neural structured prediction. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, pp. 1731–1739, Red Hook, NY, USA, 2016. Curran Associates Inc. ISBN 9781510838819.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke E. Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Francis Christiano, Jan

Leike, and Ryan J. Lowe. Training language models to follow instructions with human feedback. *ArXiv*, abs/2203.02155, 2022. URL https://api.semanticscholar.org/CorpusID:246426909.

Du Phan, Matthew D. Hoffman, David Dohan, Sholto Douglas, Tuan Anh Le, Aaron Parisi, Pavel Sountsov, Charles Sutton, Sharad Vikram, and Rif A. Saurous. Training chain-of-thought via latent-variable inference. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

Tom Rainforth, Adam R. Kosiorek, Tuan Anh Le, Chris J. Maddison, Maximilian Igl, Frank Wood, and Yee Whye Teh. Tighter variational bounds are not necessarily better. In *International Conference on Machine Learning (ICML)*, pp. 4277–4285, 2018.

Geoffrey Roeder, Yuhuai Wu, and David K. Duvenaud. Sticking the landing: Simple, lower-variance gradient estimators for variational inference. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Donald B. Rubin. Using the sir algorithm to simulate posterior distributions. 1988. URL https://api.semanticscholar.org/CorpusID:115305396.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y.K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Weijie Su, Stephen Boyd, and Emmanuel J. Candès. A differential equation for modeling nesterov's accelerated gradient method: theory and insights. *J. Mach. Learn. Res.*, 17(1):5312–5354, January 2016. ISSN 1532-4435.

Tyler Sypherd, Mario Diaz, John Kevin Cava, Gautam Dasarathy, Peter Kairouz, and Lalitha Sankar. A tunable loss function for robust classification: Calibration, landscape, and generalization. *IEEE Trans. Inf. Theor.*, 68(9):6021–6051, September 2022. ISSN 0018-9448. doi: 10.1109/TIT.2022.3169440. URL https://doi.org/10.1109/TIT.2022.3169440.

Fahim Tajwar, Guanning Zeng, Yueer Zhou, Yuda Song, Daman Arora, Yiding Jiang, Jeff Schneider, Ruslan Salakhutdinov, Haiwen Feng, and Andrea Zanette. Maximum likelihood reinforcement learning. 2026. URL https://arxiv.org/abs/2602.02710.

Gemma Team, Sherif El Abd, Vaibhav Aggarwal, Robin Algayres, Alek Andreev, Olivier Bachem, Ian Ballantyne, Cormac Brick, Victor Cărbune, Michelle Casbon, Mayank Chaturvedi, Aditya Chawla, Victor Cotruta, Alice Coucke, Phil Culliton, Robert Dadashi, Lucas Dixon, Mohamed Elhawaty, Utku Evci, Clément Farabet, Johan Ferret, Filippo Galgani, Sertan Girgin, Jean-Bastien Grill, Maarten Grootendorst, Jiaxian Guo, Cassidy Hardin, Yanzhang He, Steven M. Hernandez, Omri Homburger, Léonard Hussenot, Juyeong Ji, Armand Joulin, Aishwarya Kamath, Parnian Kassraie, Olivier Lacombe, Preethi Lahoti, Gaël Liu, Gus Martins, Luciano Martins, Tatiana Matejovicova, Ramona Merhej, Nikola Momchev, Sneha Mondal, Ryan Mullins, Sindhu Raghuram Panyam, Shreya Pathak, Sarah Perrin, André Susano Pinto, Etienne Pot, Angéline Pouget, Alexandre Ramé, Sabela Ramos, Douglas Reid, David Rim, Morgane Rivière, Karsten Roth, Louis Rouillard, Omar Sanseviero, Pier Giuseppe Sessa, Shane Settle, Danila Sinopalnikov, Sara Smoot, Piotr Stanczyk, Andreas Steiner, Lawrence Stewart, Ilya Tolstikhin, Michael Tschannen, Anton Tsitsulin, Nino Vieillard, Renjie Wu, Pingmei Xu, Haichuan Yang, Edouard Yvinec, Biao Zhang, Li Zhang, Joe Zou, Nicolas Aagnes, Abdelrahman Abdelhamed, Jakub Adamek, Shivani Agrawal, Shubham Agrawal, Ibrahim Alabdulmohsin, Jean Baptiste Alayrac, Uri Alon, Chandramouli Amarnath, Ankesh Anand, Chrysovalantis Anastasiou, Setareh Ariafar, François-Xavier Aubet, Kyriakos Axiotis, Federico Barbero, Joelle Barral, Alexei Bendebury, Urs Bergmann, Stanley Bileschi, Kat Black, Mathieu Blondel, Sebastian Borgeaud, Arthur Bražinskas, Ryan Burnell, Robert Busa-Fekete, Mu Cai, Daniele Calandriello, Glenn Cameron, Charlotte Caucheteux, Rahma Chaabouni, Garima Chadha, Jetha Chan, Blake Jianhang Chen, Jesse Chen, Lin Chen, Xu Chen, Derek Cheng, Tzu hsiang Chien, Nikolai Chinaev, Yi Chou, Zhaohui Chu, Benjamin Coleman, Pooja Consul, Sam Conway-Rahman, Scott Crowell, Dylan Cutler, Vivek Dani, Samira Daruki, Anil Das, Daniel Deutsch, Nishanth Dikkala, Li Ding, Qiuhan Ding, Shenil Dodhia, Konstantin Donhauser, Tulsee Doshi, Anca Dragan, Alex Druinsky, Sahil Dua, Zoltan Egyed, Danielle Eisenbud, Daniel Eppens, Cindy Fan, Bahare Fatemi, Yassir Fathullah, Vlad Feinberg, Milen Ferev, Sebastian

Flennerhag, Takumi Fujimoto, João Gabriel Oliveira, Isaac Galatzer-Levy, João Gante, Simon Geisler, Soham Ghosal, Antonious M. Girgis, Tamara von Glehn, Alec Go, Alhaad Gokhale, Alex Grills, Yiming Gu, Mayank Gupta, Pramod Gupta, Guru Guruganesh, Raia Hadsell, Hamza Harkous, Jitendra Harlalka, Demis Hassabis, Anja Hauth, Joe Heyward, Arian Hosseini, Chih-Yang Hsia, I-Hung Hsu, Xiaopeng Huang, Yangsibo Huang, Kevin Hui, Adrian Hutter, Te I, Fotis Iliopoulos, Advait Jain, Ganesh Jawahar, Ziwei Ji, Qilin Jin, Melvin Johnson, Kandarp Joshi, Arun Kandoor, Wang-Cheng Kang, Koray Kavukcuoglu, Mehran Kazemi, Kathleen Kenealy, Amr Khalifa, Phoebe Kirk, Ivan Korotkov, Suraj Kothawade, Vitaly Kovalev, Neel Kovelamudi, Adam Kraft, Ravin Kumar, Vivek Kumar, Harish Kuppam, Justin Lannin, Chen-Yu Lee, Seungji Lee, Dmitry Lepikhin, Alon Levkovitch, Dongdong Li, Qiujia Li, Valentin Liévin, Ethan Lin, Ziqian Lin, Casper Liu, Tianlin Liu, Tianqi Liu, Xin Liu, Ivan Lobov, Mayank Lunayach, Min Ma, Gagan Madan, Andrii Maksai, Eric Malmi, Michal Matuszak, Daniel McDuff, Gaurav Menghani, Maciej Mikuła, Daniil Mirylenka, Karolis Misiunas, Vedant Misra, Andreea Mitran, Kareem Mohamed, Maksim Mukha, Eric Noland, James O'Donnell, Brendan O'Donoghue, Kate Olszewska, Bernett Orlando, Wanqiong Pan, Rina Panigrahy, Unnati Parekh, Nicolas Perez-Nieves, Chunjong Park, Eric Paskie, Liqian Peng, Bryce Petrini, Slav Petrov, Jonas Pfeiffer, Bilal Piot, Martyna Plomecka, Siim Poder, Octavio Ponce, Arijit Pramanik, David Racz, Anish Rajan, Michelle Ramanovich, Anand Rao, Marvin Ritter, Vitor Rodrigues, Evan Rosen, Mikołaj Rybiński, Noveen Sachdeva, Michaël E. Sander, Rohit Sathyanarayana, Sagar Savla, Samuel Schmidgall, Tal Schuster, George Scrivener, Benoit Seguin, Andrew Sellergren, Aliaksei Severyn, Izhak Shafran, Dhruv Shah, Bobak Shahriari, Yuan Shangguan, Ashish Shenoy, Pradeep Shenoy, Rakesh Shivanna, Pauline Sho, Lucas Spangher, Wojciech Stokowiec, Tim Strother, Yao Su, Yinghao Sun, Mukund Sundararajan, Andrea Tacchetti, Mor Hazan Taege, Pouya Tafti, Jean Tarbouriech, Chetan Tekur, Shantanu Thakoor, Rahul Thapa, Madeleine Traverse, Lenart Treven, Tao Tu, Chien Te Tung, Çağlar Ünlü, Petar Veličković, Malini Pooni Venkat, Sagar Gubbi Venkatesh, Vidya Venkiteswaran, Francesco Visin, Alex Vitvitskyi, Kiran Vodrahalli, Weiyi Wang, Xin Wang, Tris Warkentin, Jan Wassenberg, John Wieting, Cindy Wu, Lechao Xiao, Hao Xu, Yuhui Xu, Fuzhao Xue, Arun Yadav, Jun Yan, Antoine Yang, Lin Yang, Ming-Hsuan Yang, Ziyu Ying, Jae Hyeon Yoo, Morteza Zadimoghaddam, Sajjad Zafar, Fred Zhang, Jiageng Zhang, Jianyi Zhang, Xiaofan Zhang, Chao Zhao, David Zhou, and Chen Zou. Gemma 4 technical report, 2026. URL https://arxiv.org/abs/2607.02770.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. MuSiQue: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 2022.

Constantino Tsallis. Possible generalization of boltzmann-gibbs statistics. *Journal of Statistical Physics*, 52:479–487, 1988. URL https://api.semanticscholar.org/CorpusID:16385640.

George Tucker, Dieterich Lawson, Shixiang Gu, and Chris J. Maddison. Doubly reparameterized gradient estimators for Monte Carlo objectives. In *International Conference on Learning Representations (ICLR)*, 2019.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=1PL1NIMMrw.

Zecheng Wang, Deyuan Liu, Chunshan Li, Yupeng Zhang, Zhengyun Zhao, Dianhui Chu, Bingning Wang, and Dianbo Sui. Gradients must earn their influence: Unifying sft with generalized entropic objectives, 2026. URL https://arxiv.org/abs/2602.11424.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Xueru Wen, Jie Lou, Yanjiang Liu, Hongyu Lin, Ben He, Xianpei Han, Le Sun, Yaojie Lu, and Debing Zhang. Coupled variational reinforcement learning for language model general reasoning, 2026. URL https://arxiv.org/abs/2512.12576.

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.*, 8(3–4):229–256, May 1992. ISSN 0885-6125. doi: 10.1007/BF00992696. URL https://doi.org/10.1007/BF00992696.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2018.

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Yang Yue, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in LLMs beyond the base model? In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=4OsgYD7em5.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. Star: self-taught reasoner bootstrapping reasoning with reasoning. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Zhilu Zhang and Mert R. Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, pp. 8792–8802, Red Hook, NY, USA, 2018. Curran Associates Inc.

Xiangxin Zhou, Zichen Liu, Anya Sims, Haonan Wang, Tianyu Pang, Chongxuan Li, Liang Wang, Min Lin, and Chao Du. Reinforcing general reasoning without verifiers. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=nnwvwge40d.

A Extended Related Work

q-log losses and continua. The Tsallis q-logarithm originates in non-extensive statistical mechanics (Tsallis, 1988); escort distributions were studied by Beck & Schlögl (1993). Ferrari & Yang (2010) introduced maximum L_q-likelihood (MLqE), which reweights the score by $f(X;\theta)^{1-q}$, trading a small loss of asymptotic efficiency for outlier robustness, and showed that its population minimizer is the escort distribution of order $1/q$ (Theorem 2.1 is the categorical case); the PAFT gradient Equation (8) is the marginal-likelihood analog of this weighted score. Generalized cross-entropy (Zhang & Sabuncu, 2018) and α-loss (Sypherd et al., 2022) are reparameterizations of ℓ_q for classification with noisy labels; Zhang & Sabuncu (2018) motivate generalized cross-entropy by the slow training of mean absolute error (q=0) relative to cross-entropy (q=1), a trade-off our cold-start and noise-fitting rates make precise. Concurrently, Wang et al. (2026) apply the deformed-log family at the *token* level for SFT; their token-level gate p^{α} is the single-token specialization of our example-level $P_{\boldsymbol{\theta}}^{-q}$, but their p is an exact softmax probability whereas $P_{\boldsymbol{\theta}}$ is an intractable marginal. Tsallis entropy has also been used as a policy regularizer in RL (Lee et al., 2018; Nachum et al., 2018); we use it in the loss function rather than as a policy regularizer. Tajwar et al. (2026) concurrently propose MaxRL, an RL-to-ML continuum via Maclaurin truncation of $\log p$; their estimator is unbiased for the truncated objective but exactly zero when no sample succeeds, while GARL targets the true q-loss and always has nonzero gradient since $w_m > 0$.

RL–MLE bridges and latent-variable training for reasoning. The RL-as-inference connection (Levine, 2018; Norouzi et al., 2016; Guu et al., 2017) treats MLE and RL as distinct frameworks; we embed them as endpoints of a single continuously parameterized family. Rényi variational inference (Li & Turner, 2016) provides a complementary continuum that tightens the ELBO toward $-\log P_{\boldsymbol{\theta}}$, the target J_Q shares at q=1. On the latent-variable side, RLVR and GRPO (DeepSeek-AI, 2025; Shao et al., 2024) optimize expected reward; STaR (Zelikman et al., 2022) bootstraps reasoning by generating and filtering rationales; TRICE (Phan et al., 2023) and CoVRL (Wen et al., 2026) are ELBO-based variational methods at the q=1 pole (TRICE via MCMC-EM; CoVRL via composite prior-posterior with hybrid sampling); SPG (Ding & Soricut, 2017) samples from a reward-tilted proposal $q_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathbf{x}, \mathbf{y}) \propto p_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathbf{x}) \exp(R(\mathbf{z} \mid \mathbf{y}))$ for cold-start sequence-level RL, coinciding with the posterior under log-likelihood reward. At q=1, PAFT recovers SPG's gradient and TRICE's EM gradient update over posterior samples; CoVRL further hybridizes PAFT (posterior) with GARL (prior, IWAE) via composite sampling. STaR's rejection-sampling strategy is a hard-acceptance variant of PAFT's importance resampling (Section E.2). The J_Q continuum extends these with the $\Omega(1/p_0) \to \Theta(\log(1/p_0))$ separation across q and the dual factorization through GARL.

Convergence of policy gradient. Mei et al. (2020b) prove that exact policy gradient for tabular softmax policies converges at an $O(1/t)$ rate, with constants that depend on the problem and on the smallest probability of an optimal action during training, which they bound in terms of its initial value. Mei et al. (2020a) show that such trajectories are drawn toward suboptimal deterministic policies, a "softmax gravity well" that makes progress depend strongly on the initialization, and that softmax cross-entropy with one-hot labels converges more slowly than linearly ("softmax damping"), because near the one-hot label the squared gradient norm shrinks faster than the loss; this is the $p \to 1$ regime that our escape analysis excludes ($\delta \leq 1/2$). They mitigate both by replacing the softmax with an escort transform, which normalizes powers of the parameters' magnitudes instead of their exponentials. Theorem 3.1 complements these results with a lower bound on the time to escape a small initial success probability, $\Omega(1/p_0)$ at q=0, for any differentiable model with a bounded score, and Theorem 3.2 shows that, when the score is also bounded away from zero, the per-example weight $P_{\boldsymbol{\theta}}^{-q}$ shortens it to $\Theta(\log(1/p_0))$ at q=1. J_Q thus changes the objective rather than the parameterization, keeping the softmax of a pretrained language model; the escort distribution of Theorem 2.1 is the minimizer of J_Q, not a parameterization.

Gradient estimators and verifier-free training. GARL recovers RB-REINFORCE (q=0; Zhou et al., 2026) and the IWAE gradient (q=1; Burda et al., 2015). Rainforth et al. (2018) showed IWAE's inference-network gradient SNR shrinks as M grows, motivating doubly reparameterized variants (Roeder et al., 2017; Tucker et al., 2019); our bias expansion $O(q/MP_{\boldsymbol{\theta}}^{q})$ exposes a related phenomenon along the J_Q continuum, with intermediate q balancing escape against estimator quality. Zhou et al. (2026) introduce VeriFree, the RB-REINFORCE estimator GARL extends; while Rao–

Blackwellization reduces variance, Section 5 shows it does not address the cold-start escape bottleneck. Both GARL and PAFT are verifier-free across the J_Q continuum. Finally, Yue et al. (2025) observed that RLVR narrows the reasoning capability boundary during training; our framework attributes this to mode-seeking at q=0 (Corollary C.2), with PAFT (Section 4.2) an empirically more stable alternative to GARL with prompted inputs (Section 5).

B Proofs for Section 2: Setup and Background

Proposition B.1 (RLVR connection)**.** *Under the conditional model of Section 2 and exact-match reward $R(\hat{\mathbf{y}}, \mathbf{y}^*) = \mathbb{I}(\hat{\mathbf{y}} = \mathbf{y}^*)$, the expected reward equals $\mathbb{E}_{\mathcal{D}}[P_{\boldsymbol{\theta}}]$; consequently $J_0(\boldsymbol{\theta}) = 1 - \mathbb{E}_{\mathcal{D}}[P_{\boldsymbol{\theta}}]$, and minimizing J_0 is equivalent to maximizing expected reward.*

Proof. For a fixed example $(\mathbf{x}^*, \mathbf{y}^*)$,

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\substack{\mathbf{z}\sim p_{\boldsymbol{\theta}}(\cdot|\mathbf{x}^*),\\ \hat{\mathbf{y}}\sim p_{\boldsymbol{\theta}}(\cdot|\mathbf{x}^*,\mathbf{z})}} [R(\hat{\mathbf{y}}, \mathbf{y}^*)] \\
&\quad = \sum_{\substack{\mathbf{z}\in\mathcal{Z},\\ \mathbf{y}\in\mathcal{Y}}} \Big[p_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathbf{x}^*) \\
&\qquad \cdot p_{\boldsymbol{\theta}}(\mathbf{y} \mid \mathbf{x}^*, \mathbf{z}) \mathbb{I}(\mathbf{y} = \mathbf{y}^*) \Big].
\end{aligned}$$

The indicator picks out the correct output, giving

$$\begin{aligned}
\mathop{\mathbb{E}}_{\substack{\mathbf{z}\sim p_{\boldsymbol{\theta}}(\cdot|\mathbf{x}^*),\\ \hat{\mathbf{y}}\sim p_{\boldsymbol{\theta}}(\cdot|\mathbf{x}^*,\mathbf{z})}} [R(\hat{\mathbf{y}}, \mathbf{y}^*)] &= \sum_{\mathbf{z}\in\mathcal{Z}} p_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathbf{x}^*) p_{\boldsymbol{\theta}}(\mathbf{y}^* \mid \mathbf{x}^*, \mathbf{z}) \\
&= P_{\boldsymbol{\theta}}.
\end{aligned}$$

Taking an expectation over training examples from $\mathcal{D}$, we have

$$\mathop{\mathbb{E}}_{\substack{(\mathbf{x}^*,\mathbf{y}^*)\sim\mathcal{D}\\ \mathbf{z}\sim p_{\boldsymbol{\theta}}(\cdot|\mathbf{x}^*),\\ \hat{\mathbf{y}}\sim p_{\boldsymbol{\theta}}(\cdot|\mathbf{x}^*,\mathbf{z})}} [R(\hat{\mathbf{y}}, \mathbf{y}^*)] = \mathop{\mathbb{E}}_{(\mathbf{x}^*,\mathbf{y}^*)\sim\mathcal{D}} [P_{\boldsymbol{\theta}}].$$

□

Proposition B.2 (SFT on annotated rationales is an ELBO for the q=1 objective)**.** *Fix an example $(\mathbf{x}^*, \mathbf{y}^*)$ with $P_{\boldsymbol{\theta}} > 0$, and let A be a distribution over rationales $\mathbf{z} \in \mathcal{Z}$ (an **annotation distribution**) supported where $p_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathbf{x}^*, \mathbf{y}^*) > 0$. Then*

$$\underbrace{\mathop{\mathbb{E}}_{\mathbf{z}\sim A}[-\log p_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{y}^* \mid \mathbf{x}^*)]}_{\textit{SFT loss on } (\mathbf{x}^*,\mathbf{z},\mathbf{y}^*)} = \underbrace{-\log P_{\boldsymbol{\theta}}}_{q=1 \textit{ loss}} + \mathrm{KL}\big(A \,\big\|\, p_{\boldsymbol{\theta}}(\cdot \mid \mathbf{x}^*, \mathbf{y}^*)\big) + \mathrm{H}(A), \tag{10}$$

where the entropy $\mathrm{H}(A)$ does not depend on $\boldsymbol{\theta}$. Consequently, (i) the SFT loss is an upper bound on the q=1 loss $-\log P_{\boldsymbol{\theta}}$; equivalently, $\mathbb{E}_{\mathbf{z}\sim A}[\log p_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{y}^* \mid \mathbf{x}^*)] + \mathrm{H}(A)$ is an evidence lower bound (ELBO) on $\log P_{\boldsymbol{\theta}}$ with variational distribution A. (ii) The SFT gradient is $-\nabla_{\boldsymbol{\theta}} \log P_{\boldsymbol{\theta}} + \nabla_{\boldsymbol{\theta}} \mathrm{KL}(A \,\|\, p_{\boldsymbol{\theta}}(\cdot \mid \mathbf{x}^*, \mathbf{y}^*))$; the KL gap vanishes iff $A = p_{\boldsymbol{\theta}}(\cdot \mid \mathbf{x}^*, \mathbf{y}^*)$, in which case the SFT gradient equals $-\nabla_{\boldsymbol{\theta}} \log P_{\boldsymbol{\theta}}$ exactly.

Proof. Fix $(\mathbf{x}^*, \mathbf{y}^*)$ with $P_{\boldsymbol{\theta}} > 0$ and write $\pi_{\boldsymbol{\theta}}(\mathbf{z}) \triangleq p_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathbf{x}^*, \mathbf{y}^*) = p_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{y}^* \mid \mathbf{x}^*)/P_{\boldsymbol{\theta}}$ for the model's posterior over rationales. The support condition on A makes every term below finite. For each $\mathbf{z}$ in the support of A, the chain rule gives

$$-\log p_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{y}^* \mid \mathbf{x}^*) = -\log P_{\boldsymbol{\theta}} - \log \pi_{\boldsymbol{\theta}}(\mathbf{z}) = -\log P_{\boldsymbol{\theta}} + \log \frac{A(\mathbf{z})}{\pi_{\boldsymbol{\theta}}(\mathbf{z})} - \log A(\mathbf{z}).$$

Taking expectations under $\mathbf{z} \sim A$,

$$\mathop{\mathbb{E}}_{\mathbf{z}\sim A}[-\log p_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{y}^* \mid \mathbf{x}^*)] = -\log P_{\boldsymbol{\theta}} + \mathrm{KL}(A \,\|\, \pi_{\boldsymbol{\theta}}) + \mathrm{H}(A),$$

which is Equation (10); $\mathrm{H}(A) = -\mathbb{E}_{\mathbf{z}\sim A}[\log A(\mathbf{z})]$ does not involve $\boldsymbol{\theta}$.

(i) Since $\mathrm{KL}(A \,\|\, \pi_{\boldsymbol{\theta}}) \geq 0$ and $\mathrm{H}(A) \geq 0$, the SFT loss is at least $-\log P_{\boldsymbol{\theta}}$. Rearranging Equation (10),

$$\log P_{\boldsymbol{\theta}} = \mathop{\mathbb{E}}_{\mathbf{z}\sim A}[\log p_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{y}^* \mid \mathbf{x}^*)] + \mathrm{H}(A) + \mathrm{KL}(A \,\|\, \pi_{\boldsymbol{\theta}}) \geq \mathop{\mathbb{E}}_{\mathbf{z}\sim A}[\log p_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{y}^* \mid \mathbf{x}^*)] + \mathrm{H}(A),$$

the evidence lower bound with variational distribution A; the same inequality follows from Jensen's inequality applied to $\log P_{\boldsymbol{\theta}} = \log \mathbb{E}_{\mathbf{z}\sim A}[p_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{y}^* \mid \mathbf{x}^*)/A(\mathbf{z})]$.

(ii) Differentiating Equation (10) in $\boldsymbol{\theta}$ with A fixed,

$$\nabla_{\boldsymbol{\theta}} \mathop{\mathbb{E}}_{\mathbf{z}\sim A}[-\log p_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{y}^* \mid \mathbf{x}^*)] = -\nabla_{\boldsymbol{\theta}} \log P_{\boldsymbol{\theta}} + \nabla_{\boldsymbol{\theta}} \mathrm{KL}(A \,\|\, \pi_{\boldsymbol{\theta}}),$$
$$\nabla_{\boldsymbol{\theta}} \mathrm{KL}(A \,\|\, \pi_{\boldsymbol{\theta}}) = -\mathop{\mathbb{E}}_{\mathbf{z}\sim A}[\nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(\mathbf{z})].$$

By Gibbs' inequality the KL term vanishes iff $A = \pi_{\boldsymbol{\theta}}$. When $A = \pi_{\boldsymbol{\theta}}$ at the current iterate, $-\mathbb{E}_{\mathbf{z}\sim\pi_{\boldsymbol{\theta}}}[\nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(\mathbf{z})] = -\nabla_{\boldsymbol{\theta}} \sum_{\mathbf{z}\in\mathcal{Z}} \pi_{\boldsymbol{\theta}}(\mathbf{z}) = 0$ (for countably infinite $\mathcal{Z}$, by dominated convergence), so the SFT gradient equals $-\nabla_{\boldsymbol{\theta}} \log P_{\boldsymbol{\theta}}$ exactly. Equivalently, $\nabla_{\boldsymbol{\theta}} \log P_{\boldsymbol{\theta}} = \mathbb{E}_{\mathbf{z}\sim\pi_{\boldsymbol{\theta}}}[\nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{y}^* \mid \mathbf{x}^*)]$ (Fisher's identity), the posterior form of the gradient used in Equation (8). □

Remark B.3 (The single-annotation case and cold-start escape). With one annotated rationale per example, $A = \delta_{\mathbf{z}^*}$, we have $\mathrm{H}(A) = 0$ and $\mathrm{KL}(\delta_{\mathbf{z}^*} \,\|\, \pi_{\boldsymbol{\theta}}) = -\log \pi_{\boldsymbol{\theta}}(\mathbf{z}^*)$, so Equation (10) reads $-\log p_{\boldsymbol{\theta}}(\mathbf{z}^*, \mathbf{y}^* \mid \mathbf{x}^*) = -\log P_{\boldsymbol{\theta}} - \log p_{\boldsymbol{\theta}}(\mathbf{z}^* \mid \mathbf{x}^*, \mathbf{y}^*)$: the standard SFT loss is the $q{=}1$ loss plus a penalty on the posterior probability the model assigns to the annotated rationale. Under gradient flow, the SFT loss is a log-loss on the joint probability $p_{\boldsymbol{\theta}}(\mathbf{z}^*, \mathbf{y}^* \mid \mathbf{x}^*)$. Theorems 3.1 and 3.2 are stated for gradient flow on $-\ln_q$ of any differentiable probability with bounded score and therefore apply to it verbatim with $P_{\boldsymbol{\theta}}$ replaced by the joint; since the joint is at most $P_{\boldsymbol{\theta}}$, driving it to δ drives the success probability to at least δ. This is the sense in which the SFT stage of the SFT-then-RLVR pipeline escapes cold start at the $q{=}1$ rate (Section 3). The same penalty commits the model to the annotation whether or not it is correct, so annotation errors are fit rather than filtered.

Remark B.4 (The no-annotation case: EM and PAFT at $q{=}1$). The KL gap in Equation (10) measures how far the annotations sit from the model's own posterior over rationales; it is zero, and the ELBO is tight, exactly when $A = \pi_{\boldsymbol{\theta}}$. Taking A to be the model posterior at the current iterate is therefore the natural choice when no annotations exist: by Proposition B.2(ii), an SFT step on posterior-sampled rationales follows the exact $q{=}1$ gradient $-\nabla_{\boldsymbol{\theta}} \log P_{\boldsymbol{\theta}}$ (Fisher's identity; Equation (8)). With A frozen at an annotator, SFT is a generalized-EM M-step whose E-step never updates; with A re-tightened to the posterior at every iterate, it is the EM gradient update (Dempster et al., 1977), which PAFT recovers at $q{=}1$ (Section 4.2, Section E.2): SFT with self-generated rationales. This is how the $q{=}1$ pole is realized in our no-annotation experiments (Section 5); PAFT at $q < 1$ keeps the same posterior-sampled SFT step and attenuates it by $P_{\boldsymbol{\theta}}^{1-q}$ (Proposition 2.2).

C Proofs for Section 2: Loss Landscape

Proposition C.1 (Dispersion penalty)**.** *For $q > 0$, $J_Q(\boldsymbol{\theta}, q) \geq -\ln_q(\bar{P})$, where $\bar{P} \triangleq \mathbb{E}_{(\mathbf{x}^*,\mathbf{y}^*)\sim\mathcal{D}}[P_{\boldsymbol{\theta}}]$ is the mean success probability across examples, with equality if and only if $P_{\boldsymbol{\theta}}$ is constant across all examples in $\mathcal{D}$.*

Proof. For $q > 0$, the function $h_q(u) = -\ln_q(u) = \frac{1-u^{1-q}}{1-q}$ is strictly convex on $(0, 1]$, since $h_q''(u) = q\, u^{-q-1} > 0$. Applying Jensen's inequality:

$$J_Q(\boldsymbol{\theta}, q) = \mathop{\mathbb{E}}_{(\mathbf{x}^*,\mathbf{y}^*)\sim\mathcal{D}}[h_q(P_{\boldsymbol{\theta}})]$$
$$\geq h_q\Big(\mathop{\mathbb{E}}_{(\mathbf{x}^*,\mathbf{y}^*)\sim\mathcal{D}}[P_{\boldsymbol{\theta}}]\Big) = -\ln_q(\bar{P}),$$

with equality iff $P_{\boldsymbol{\theta}}$ is constant across all examples. □

Theorem 2.1. *[Minimizers of J_Q in the categorical model] For $q \in (0, 1]$, the unique minimizer of $J_Q(\theta, q) = \sum_j \alpha_k(-\ln_q \theta_k)$ over $\theta \in \Delta_K$ is $\theta_k^*(q) = \frac{\alpha_k^{1/q}}{\sum_{k'} \alpha_{k'}^{1/q}}$. For $q = 0$, any vertex e_j with $j \in \operatorname{argmax}_k \alpha_k$ is optimal.*

Proof. Case $q \in (0, 1]$. Since h_q is strictly convex for $q > 0$, the objective is strictly convex on the interior of Δ_K, and the minimizer is unique. Since all $\alpha_j > 0$, the minimizer lies in the interior (any boundary point has infinite loss for $q = 1$ and suboptimal loss for $q < 1$), so we can use Lagrange multipliers for the equality constraint $\sum_j \theta_j = 1$:

$$-\alpha_j \theta_j^{-q} - \lambda = 0 \quad \Longrightarrow \quad \alpha_j \theta_j^{-q} = \mu \quad \text{for all } j,$$

where $\mu \triangleq -\lambda > 0$. Solving: $\theta_j = (\alpha_j/\mu)^{1/q}$. The constraint $\sum_j \theta_j = 1$ yields $\mu^{1/q} = \sum_k \alpha_k^{1/q}$, giving $\theta_j^*(q) = \alpha_j^{1/q} / \sum_k \alpha_k^{1/q}$ as in Theorem 2.1.

Case $q = 0$. The objective $J_Q(\boldsymbol{\theta}, 0) = 1 - \sum_j \alpha_j \theta_j$ is linear, minimized at any vertex e_j with $j \in \operatorname{argmax}_k \alpha_k$. □

Corollary C.2 (Endpoint behavior and monotone sharpening)**.** *Under the categorical model:*

1. ***Density-estimation pole*** *($q = 1$): $\theta_j^*(1) = \alpha_j$. The model exactly recovers the data distribution.*
2. ***Exploitation pole*** *($q \to 0^+$): assuming a unique mode $j^* = \operatorname{argmax}_k \alpha_k$, $\theta_j^*(q) \to \mathbb{I}(j = j^*)$. The model concentrates all mass on the most frequent output.*
3. ***Monotone sharpening****: for $0 < q' < q \leq 1$ and $\alpha_j > \alpha_k$, $\theta_j^*(q')/\theta_k^*(q') > \theta_j^*(q)/\theta_k^*(q)$.*

Proof. Part (1): $1/q = 1$. Part (2): $(\alpha_j/\alpha_{j^*})^{1/q} \to 0$ for $j \neq j^*$. Part (3): $\theta_j^*/\theta_k^* = (\alpha_j/\alpha_k)^{1/q}$, increasing in $1/q$. □

Corollary C.3 (Propriety)**.** *The Tsallis q-logarithmic scoring rule is strictly proper if and only if $q = 1$.*

Proof. By Theorem 2.1, the maximizer of $\mathbb{E}_{y\sim\alpha}[\ln_q(\theta_y)]$ is $\theta_j^* \propto \alpha_j^{1/q}$, which equals α iff $q = 1$. For $q \in (0, 1)$ the true distribution α is not even a maximizer (the rule is not proper at all), let alone the unique one. □

The robustness counterpart under label noise — both static (where the escort minimizer concentrates) and dynamic (how fast the model gets there) — appears in Section D.5.

D Proofs for Section 3: Commitment Dynamics under Gradient Flow

D.1 Warm-up: exact analysis on the sigmoid model

Before proving the general results, we work through the scalar sigmoid model $P(\theta) = \sigma(\theta) = (1 + e^{-\theta})^{-1}$ as a warm-up. This model admits exact closed-form escape times that validate the $\Theta(\cdot)$ bounds in Theorem 3.2.

Under gradient flow on $\ell_q(\theta) = -\ln_q(\sigma(\theta))$, the parameter evolves as $\dot{\theta} = P(\theta)^{-q} P'(\theta)$. Since $P'(\theta) = P(\theta)(1 - P(\theta))$, the chain rule gives:

$$\dot{p} = [P'(\theta)]^2 P(\theta)^{-q} = p^{2-q}(1 - p)^2.$$

This is a special case of the general dynamics (Equation (3)) with score norm $\|s(\theta)\|^2 = (1 - p)^2$, which satisfies $\|s\|^2 \in [(1 - \delta)^2, 1]$ on $p \in [p_0, \delta]$ — confirming the bounded score assumption.

The separable ODE gives the exact escape time:

$$T_q(p_0, \delta) = \int_{p_0}^{\delta} \frac{du}{u^{2-q}(1-u)^2}. \tag{11}$$

We evaluate this integral using a dominant/remainder decomposition. Write $(1-u)^{-2} = 1 + r(u)$ where $r(u) = \frac{2u-u^2}{(1-u)^2}$. On $u \in [0, \delta]$ with $\delta \leq 1/2$, we have $0 \leq r(u) \leq 8u$. Substituting and distributing:

$$T_q(p_0, \delta) = \underbrace{\int_{p_0}^{\delta} \frac{du}{u^{2-q}}}_{\text{dominant}} + \underbrace{\int_{p_0}^{\delta} \frac{r(u)}{u^{2-q}}\, du}_{\text{remainder}}.$$

Case $q \in (0, 1)$. The dominant integral evaluates to $\frac{p_0^{-(1-q)} - \delta^{-(1-q)}}{1-q} = \frac{p_0^{-(1-q)}}{1-q}(1 + o(1))$. The remainder satisfies $0 \leq \int r(u)\, u^{-(2-q)}\, du \leq 8 \int u^{q-1}\, du = \frac{8\delta^q}{q}$, a constant. So the remainder is negligible and $T_q = \frac{p_0^{-(1-q)}}{1-q}(1 + o(1))$.

Case $q = 0$. The dominant integral gives $\frac{1}{p_0}(1+o(1))$. The remainder is $O(\log(1/p_0))$, still negligible compared to $1/p_0$. So $T_0 = \frac{1}{p_0}(1 + o(1))$.

Case $q = 1$. The dominant integral is $\log(1/p_0) + \log \delta$. The remainder satisfies $\int r(u)\, u^{-1}\, du \leq 8(\delta - p_0) = O(1)$. So $T_1 = \log(1/p_0)(1 + o(1))$.

Note that the sigmoid model yields exact $1 + o(1)$ asymptotics (not just $\Theta(\cdot)$) because $\|s\|^2 = (1-p)^2 \to 1$ as $p \to 0$, so the score norm converges to a known constant. This is stronger than the general theorem, which only assumes bounded score norms.

D.2 Proof of Theorem 3.1: Exploitation is provably slow

Theorem 3.1. *[Exploitation is provably slow] Let $\boldsymbol{\theta} \in \mathbb{R}^d$ parameterize any differentiable model. Consider gradient flow on $\ell_q(\boldsymbol{\theta}) = -\ln_q(P_{\boldsymbol{\theta}})$, starting from $p_0 = P_{\boldsymbol{\theta}(0)} \in (0, 1/2)$ with fixed target $\delta \in (p_0, 1/2]$. Suppose $\|s(\boldsymbol{\theta}(t))\| \leq C \in \mathbb{R}$. Then as $p_0 \to 0$:*

$$T_q(p_0, \delta) = \Omega\left(\frac{p_0^{-(1-q)}}{1-q}\right) \text{ for } q \in [0, 1),$$

$$T_1(p_0, \delta) = \Omega\left(\log \frac{1}{p_0}\right).$$

Proof. From Equation (3), $\dot{p} = p^{2-q}\|s(\boldsymbol{\theta})\|^2 \leq C^2\, p^{2-q}$. By the ODE comparison principle (since $u \mapsto u^{2-q}$ is nondecreasing on $(0, 1]$), $p(t) \leq p^*(t)$ where p^* solves $\dot{p}^* = C^2 (p^*)^{2-q}$ with $p^*(0) = p_0$. So p reaches δ no sooner than p^*:

$$T_q \geq \frac{1}{C^2} \int_{p_0}^{\delta} \frac{du}{u^{2-q}}.$$

For $q \in [0, 1)$, the integral evaluates to $\frac{p_0^{-(1-q)} - \delta^{-(1-q)}}{1-q} = \frac{p_0^{-(1-q)}}{1-q}(1 + o(1))$, giving $T_q = \Omega(p_0^{-(1-q)}/(1-q))$.

For $q = 1$, the integral is $\log(\delta/p_0) = \log(1/p_0)(1 + o(1))$, giving $T_1 = \Omega(\log(1/p_0))$. □

D.3 Proof of Theorem 3.2: Tight cold-start escape rates

Theorem 3.2. *[Tight cold-start escape rates] Under the same setup as Theorem 3.1, suppose additionally that $\|s(\boldsymbol{\theta}(t))\| \geq c > 0$ throughout the trajectory. Then as $p_0 \to 0$,*

$$T_q(p_0, \delta) = \Theta\left(\frac{p_0^{-(1-q)}}{1-q}\right) \text{ for } q \in [0, 1), \qquad T_1(p_0, \delta) = \Theta\left(\log \tfrac{1}{p_0}\right),$$

and consequently $T_q(p_0, \delta)/T_{q'}(p_0, \delta) \to \infty$ for any $q < q' \leq 1$.

Proof. The lower bound on time (Ω) follows from Theorem 3.1. For the upper bound, the additional assumption $\|s\| \geq c > 0$ gives $\dot{p} \geq c^2\, p^{2-q}$; by the ODE comparison principle, $p(t) \geq p_*(t)$ where p_* solves $\dot{p}_* = c^2 (p_*)^{2-q}$, so p reaches δ no later than p_*:

$$T_q \leq \frac{1}{c^2} \int_{p_0}^{\delta} \frac{du}{u^{2-q}}.$$

This integral evaluates to $\frac{p_0^{-(1-q)}}{1-q}(1 + o(1))$ for $q \in [0, 1)$ and $\log(1/p_0)(1 + o(1))$ for $q = 1$. Combined with the lower bound, $T_q = \Theta(p_0^{-(1-q)}/(1-q))$ for $q < 1$ and $T_1 = \Theta(\log(1/p_0))$.

Speedup ratio. For $q < q' < 1$: $T_q/T_{q'} = \Theta(p_0^{-(q'-q)}) \to \infty$. For $q < 1$ and $q' = 1$: $T_q/T_1 = \Theta(p_0^{-(1-q)}/\log(1/p_0)) \to \infty$. □

D.4 The lower bound under interference and discrete steps

Theorem 3.1 tracks one example under gradient flow. Both simplifications are conservative for this lower bound: interference between examples and discrete, choppy minibatch updates cannot beat the single-example speed limit $\dot{p} \leq C^2 p^{2-q}$ of Equation (3).

Consider N training examples with success probabilities $P_i \triangleq P_{\boldsymbol{\theta}}(\mathbf{x}_i^*, \mathbf{y}_i^*)$, scores $s_i \triangleq \nabla_{\boldsymbol{\theta}} \log P_i$ satisfying $\|s_i\| \leq C$ along the trajectory, and $p_{\max} \triangleq \max_i P_i$. By Proposition 2.2, a gradient step on J_Q restricted to a minibatch $B \subseteq \{1, \ldots, N\}$ moves $\boldsymbol{\theta}$ along $\mathbf{d}_B \triangleq \frac{1}{|B|} \sum_{j \in B} P_j^{-q} \nabla_{\boldsymbol{\theta}} P_j$. Since $\|\nabla_{\boldsymbol{\theta}} P_j\| = P_j \|s_j\| \leq C P_j$ and $P_j^{1-q} \leq p_{\max}^{1-q}$ for $q \leq 1$, Cauchy–Schwarz gives, for every example i,

$$\nabla_{\boldsymbol{\theta}} P_i \cdot \mathbf{d}_B \;\leq\; \frac{1}{|B|} \sum_{j \in B} P_j^{-q}\, \|\nabla_{\boldsymbol{\theta}} P_i\|\, \|\nabla_{\boldsymbol{\theta}} P_j\| \;\leq\; C^2 p_{\max}^{2-q}, \qquad \|\mathbf{d}_B\| \;\leq\; C\, p_{\max}^{1-q}. \tag{12}$$

Each example's push on P_i is capped by the size of its own gradient, which is small at cold start: alignment between examples can redistribute gradient mass but not create it. The bound is attained only when all gradients are positively parallel with scores at the cap; any misalignment lowers it.

Proposition D.1 (Interference cannot beat the single-example bound)**.** *Under full-batch gradient flow $\dot{\boldsymbol{\theta}} = \mathbf{d}_{\{1,\ldots,N\}}$ on J_Q with $p_{\max}(0) < \delta \leq 1/2$, no example reaches success probability δ before time $\frac{1}{C^2} \int_{p_{\max}(0)}^{\delta} u^{-(2-q)}\, du$, the bound of Theorem 3.1 at the easiest example's initial probability.*

Proof. Every example attaining the maximum at time t has $\dot{P}_i \leq C^2 p_{\max}^{2-q}$ by Equation (12). As a maximum of finitely many C^1 functions, $p_{\max}$ has upper right Dini derivative $D^+ p_{\max}(t) = \max_{i : P_i(t) = p_{\max}(t)} \dot{P}_i(t) \leq C^2 p_{\max}(t)^{2-q}$. Hence $G(t) \triangleq \int_{p_{\max}(0)}^{p_{\max}(t)} u^{-(2-q)}\, du$ is continuous with $D^+ G \leq C^2$, so $G(t) \leq C^2 t$; at the first time $p_{\max}(t) = \delta$, $G(t) = \int_{p_{\max}(0)}^{\delta} u^{-(2-q)}\, du$. □

Proposition D.2 (Discrete minibatch steps cannot beat it either)**.** *Let $\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \eta\, \mathbf{d}_{B_k}$ for a step size $\eta > 0$ and any sequence of minibatches $B_k \subseteq \{1, \ldots, N\}$, with exact per-example gradients. Suppose each $\nabla_{\boldsymbol{\theta}} P_i$ is L-Lipschitz, let p_0 be the value of $p_{\max}$ at $\boldsymbol{\theta}_0$, and let $\eta \leq 2p_0^q/L$. Then the first step K at which some example reaches δ satisfies $K \geq \frac{1}{2\eta C^2} \int_{p_0}^{\delta} u^{-(2-q)}\, du$: the elapsed time ηK is at least half the gradient-flow bound. At $q = 0$ the step-size condition can be dropped by replacing the factor 2 with $1 + L\eta/2$.*

Proof. Write $p_k \triangleq p_{\max}(\boldsymbol{\theta}_k)$. L-smoothness and Equation (12) give, for every example, $P_i(\boldsymbol{\theta}_{k+1}) \leq P_i(\boldsymbol{\theta}_k) + \eta\, \nabla_{\boldsymbol{\theta}} P_i \cdot \mathbf{d}_{B_k} + \frac{L}{2}\eta^2 \|\mathbf{d}_{B_k}\|^2 \leq p_k + f(p_k)$ with $f(u) \triangleq \eta C^2 u^{2-q}\big(1 + \frac{L\eta}{2} u^{-q}\big)$. The iterates need not increase monotonically, so we track the running maximum $u_k \triangleq \max_{j \leq k} p_j$: since f is nonnegative and nondecreasing for $q \leq 1$, $u_{k+1} \leq u_k + f(u_k)$, and since $u_k \geq p_0$, the step-size condition gives $f(u_k) \leq 2\eta C^2 u_k^{2-q}$. As $u^{-(2-q)}$ is decreasing, $\int_{u_k}^{u_{k+1}} u^{-(2-q)}\, du \leq (u_{k+1} - u_k)\, u_k^{-(2-q)} \leq 2\eta C^2$. Summing over $k < K$, with $u_0 = p_0$ and $u_K \geq \delta$, gives $\int_{p_0}^{\delta} u^{-(2-q)}\, du \leq 2\eta C^2 K$. □

Because the argument bounds the progress of each step, it holds for every minibatch sequence, including non-monotone (choppy) trajectories, where dips in p only delay escape. It concerns the lower bound only: the matching upper bound of Theorem 3.2 does not extend to many examples, since conflicting examples can slow escape at every q. Instead, we provide empirical evidence of achievability at dataset scale (Table 1).

D.5 Noise-fitting rate under symmetric label noise

The cold-start escape rates (Theorems 3.1 and 3.2) measure how fast the model commits to correct supervision under the J_Q amplification $P_{\boldsymbol{\theta}}^{-q}$. The symmetric question is how fast the model commits to *incorrect* supervision: the same amplification drives both, giving the following dynamical formulation of robustness under label noise.

Noise-contamination setup. We work with a two-label categorical model, chosen to expose the mechanism in the simplest possible setting. For a single input $\mathbf{x}^*$, the model predicts one of two labels $\{c, k\}$ with probabilities $p_{\boldsymbol{\theta}}(c \mid \mathbf{x}^*) = p$ and $p_{\boldsymbol{\theta}}(k \mid \mathbf{x}^*) = 1 - p$. We instantiate the parameterization with the sigmoid $p = \sigma(\theta)$ used in Section D.1, under which $s \triangleq \nabla_\theta \log p = \tilde{p}$ and $\|s\|^2 = \tilde{p}^2$. The target label is *corrupted*: with probability $1 - \epsilon$ it equals the clean value c, and with probability $\epsilon \in (0, 1/2)$ it flips to the noise value k, giving $\tilde{\alpha} = (1-\epsilon, \epsilon)$. The restriction to two labels is cosmetic: in the N-label categorical model with symmetric noise $\tilde{\alpha} = (1 - \epsilon)\alpha + \epsilon \cdot \mathrm{Unif}$, conditioning on the two-subset $\{j^*, k\}$ containing the clean mode j^* and any fixed wrong label k reduces to this binary setting.

Let $p(t) = p_{\boldsymbol{\theta}}(c \mid \mathbf{x}^*)$ denote the clean-mode probability under gradient flow on $J_Q(\boldsymbol{\theta}) = \mathbb{E}_{y\sim\tilde{\alpha}}[\ell_q(p_{\boldsymbol{\theta}}(y \mid \mathbf{x}^*))]$, and let $\tilde{p}(t) = 1 - p(t)$ denote the noise contamination. The cold-start analysis (Theorem 3.2) assumed a non-vanishing score $\|s\| \geq c_* > 0$; the analogous lower bound fails near $p = 1$, where the sigmoid score vanishes linearly in $\tilde{p}$, so we substitute the actual scaling $\|s\|^2 = \tilde{p}^2$ rather than treating $\|s\|$ as a constant.

The escort asymptote. Differentiating $J(p) = (1 - \epsilon)\ell_q(p) + \epsilon\ell_q(1 - p)$ gives $J'(p) = -(1 - \epsilon)p^{-q} + \epsilon\tilde{p}^{-q}$. Gradient flow on the sigmoid yields

$$\dot{\tilde{p}} = -\dot{p} = [\epsilon\tilde{p}^{-q} - (1 - \epsilon)(1 - \tilde{p})^{-q}]\, p^2\, \tilde{p}^2. \tag{13}$$

For $q > 0$, the dynamics have a unique stable equilibrium at

$$\tilde{p}_*(q) \;=\; (\epsilon/(1 - \epsilon))^{1/q}\,(1 + o(1)) \quad \text{as } \epsilon \to 0, \tag{14}$$

obtained by solving $J'(p) = 0$ ($\|s\|^2$ cancels at equilibrium, so $\tilde{p}_*(q)$ does not depend on the parameterization). This equilibrium coincides with the static escort minimizer from Theorem 2.1 applied to $\tilde{\alpha}$: at $q = 1$, $\tilde{p}_*(1) = \epsilon$ (the model fits observed noise exactly); as $q \to 0$, $\tilde{p}_*(q) \to 0$ (the model concentrates on the clean mode, paralleling Corollary C.2). The escort is both where J_Q is minimized (static) and where gradient flow converges (dynamic).

The noise-to-clean ratio $\epsilon\tilde{p}^{-q}/[(1 - \epsilon)(1 - \tilde{p})^{-q}]$ is monotone decreasing in $\tilde{p}$ on $(0, 1)$: it diverges as $\tilde{p} \to 0$ (noise term dominates near the clean mode), equals 1 at $\tilde{p} = \tilde{p}_*(q)$ (equilibrium), and vanishes as $\tilde{p} \to 1$. So for $\tilde{p} \ll \tilde{p}_*(q)$ — the regime of small noise contamination — the noise term in Equation (13) dominates by an arbitrarily large factor. This drives the asymptotic scaling.

Proposition D.3 (Noise-fitting rate)**.** *Fix $q \in (0, 1]$. Under the setup above, starting from $\tilde{p}(0) = \tilde{p}_0$ with $\tilde{p}_0 \ll \tilde{p}_*(q)$, the time $T_q^{\mathrm{noise}}(\tilde{p}_0)$ to reach a fixed target η (with $\tilde{p}_0 \ll \eta \leq \tilde{p}_*(q)$, η independent of $\tilde{p}_0$) satisfies, as $\tilde{p}_0 \to 0$:*

$$T_q^{\mathrm{noise}}(\tilde{p}_0) = \Theta\left(\frac{\tilde{p}_0^{-(1-q)}}{(1 - q)\,\epsilon}\right) \text{ for } q \in (0, 1), \qquad T_1^{\mathrm{noise}}(\tilde{p}_0) = \Theta\left(\frac{\log(1/\tilde{p}_0)}{\epsilon}\right). \tag{15}$$

The speedup ratio for $0 < q < q' \leq 1$ diverges: $T_q^{\mathrm{noise}}(\tilde{p}_0)/T_{q'}^{\mathrm{noise}}(\tilde{p}_0) = \Theta(\tilde{p}_0^{-(q'-q)}) \to \infty$ as $\tilde{p}_0 \to 0$. At $q = 0$, adopting the convention $\tilde{p}^0 \equiv 1$, the dynamics Equation (13) reduce to $\dot{\tilde{p}} = -(1 - 2\epsilon)\, p^2\, \tilde{p}^2 < 0$ everywhere (for $\epsilon < 1/2$), so any positive $\tilde{p}_0$ decays monotonically toward 0: $T_0^{\mathrm{noise}}(\tilde{p}_0) = \infty$ for any target $\eta > \tilde{p}_0$.

Proof. By the noise-to-clean monotonicity established above, for any $K > 1$ there exists $\tilde{p}_K(q) = K^{-1/q}\,\tilde{p}_*(q)(1 + o(1))$ such that for $\tilde{p} \le \tilde{p}_K$, the noise term in Equation (13) exceeds K times the clean term. Combined with $p = 1 - \tilde{p} \to 1$ as $\tilde{p} \to 0$ and $\|s\|^2 = \tilde{p}^2$:

$$\dot{\tilde{p}} \;\in\; \big[(1 - \tfrac{1}{K})\,\epsilon\,\tilde{p}^{2-q}\,(1 + o(1)),\; \epsilon\,\tilde{p}^{2-q}\big].$$

Fix any $K > 1$ (e.g., $K = 2$). Separating variables, $\tilde{p}^{q-2}\,d\tilde{p} = \Theta(\epsilon)\,dt$. For $q \in (0, 1)$, integrating from $\tilde{p}_0$ to η with $\tilde{p}_0 \ll \eta \le \tilde{p}_K(q)$ gives

$$\frac{\tilde{p}_0^{-(1-q)} - \eta^{-(1-q)}}{1 - q} = \Theta(\epsilon T),$$

so $T_q^{\text{noise}}(\tilde{p}_0) = \Theta(\tilde{p}_0^{-(1-q)}/((1-q)\epsilon))$ as $\tilde{p}_0 \to 0$. (The integral from exactly $\tilde{p}_0 = 0$ diverges for $q \le 1$, so a positive starting contamination is required.) For $q = 1$, $\dot{\tilde{p}} = \Theta(\epsilon\,\tilde{p})$ gives $\tilde{p}(t) = \tilde{p}_0\,\exp(\Theta(\epsilon\,t))$, so $T_1^{\text{noise}}(\tilde{p}_0) = \Theta(\log(\eta/\tilde{p}_0)/\epsilon) = \Theta(\log(1/\tilde{p}_0)/\epsilon)$. The speedup ratio $T_q/T_{q'} = \Theta(\tilde{p}_0^{-(q'-q)})$ diverges for $q < q' \le 1$ as $\tilde{p}_0 \to 0$. □

Structural parallel with cold-start escape. Theorem 3.2 gives $T_q^{\text{escape}}(p_0) = \Theta(p_0^{-(1-q)}/(1-q))$ for $q < 1$ and $\Theta(\log(1/p_0))$ at q=1, with speedup ratio $\Theta(p_0^{-(q'-q)})$. Proposition D.3 gives $T_q^{\text{noise}}(\tilde{p}_0) = \Theta(\tilde{p}_0^{-(1-q)}/((1-q)\epsilon))$ and $\Theta(\log(1/\tilde{p}_0)/\epsilon)$, with speedup ratio $\Theta(\tilde{p}_0^{-(q'-q)})$ — the *exact dual*: same exponent in the small starting probability (p_0 for cold-start escape from clean, $\tilde{p}_0$ for noise-fitting escape from corruption), with the noise rate ϵ as the only additional rate factor. The same $P_{\boldsymbol{\theta}}^{-q}$ amplification accelerates commitment to clean and corrupted supervision by the same multiplicative factor. Static mode-seeking (Corollary C.2) is recovered as the $t \to \infty$ limit of Equation (13): $\tilde{p}(t) \to \tilde{p}_*(q) \to 0$ as $q \to 0$.

D.6 Near-optimality convergence (supplementary result)

Proposition D.4 (Near-optimality convergence is q-independent)**.** *Suppose that near optimality, $\|s(\boldsymbol{\theta})\|^2$ depends on $\boldsymbol{\theta}$ only through $P_{\boldsymbol{\theta}}$ (i.e., $\|s(\boldsymbol{\theta})\|^2 = h(P_{\boldsymbol{\theta}})$ for some function h). Then for $\epsilon_0 \ll 1$ and $\epsilon_1 < \epsilon_0$, the time to improve from $P_{\boldsymbol{\theta}} = 1 - \epsilon_0$ to $P_{\boldsymbol{\theta}} = 1 - \epsilon_1$ satisfies*

$$T_q(1 - \epsilon_0, 1 - \epsilon_1) = T_{q'}(1 - \epsilon_0, 1 - \epsilon_1)\big(1 + O(\epsilon_0)\big)$$

for all $q, q' \in [0, 1]$. That is, the convergence time is the same for all members of the J_Q family up to a correction that vanishes as $\epsilon_0 \to 0$.

Proof. Write $\epsilon = 1 - p$ with $\epsilon \ll 1$. From Equation (3), $\dot{\epsilon} = -(1-\epsilon)^{2-q}\,\|s(\boldsymbol{\theta})\|^2 < 0$. Since ϵ decreases over time, the convergence time from ϵ_0 to ϵ_1 is:

$$T_q = \int_{\epsilon_1}^{\epsilon_0} \frac{d\epsilon}{(1-\epsilon)^{2-q}\,\|s(\boldsymbol{\theta})\|^2}.$$

For any $q, q' \in [0, 1]$, the integrands of T_q and $T_{q'}$ differ by the factor $(1-\epsilon)^{q-q'}$. We bound this factor on $\epsilon \in [\epsilon_1, \epsilon_0]$ with $\epsilon_0 \ll 1$. Using the Taylor expansion $\log(1-\epsilon) = -\epsilon - \epsilon^2/2 - \cdots$:

$$\begin{aligned}\log(1-\epsilon)^{q-q'} &= (q - q')\log(1-\epsilon) \\ &= (q - q')\big(-\epsilon - \tfrac{\epsilon^2}{2} - \cdots\big).\end{aligned}$$

Since $|q - q'| \le 1$:

$$\big|\log(1-\epsilon)^{q-q'}\big| \le \epsilon + \tfrac{\epsilon^2}{2} + \cdots = O(\epsilon).$$

Exponentiating and using $e^x = 1 + x + O(x^2) = 1 + O(\epsilon)$ for $x = O(\epsilon)$, we get $(1-\epsilon)^{q-q'} = 1 + O(\epsilon)$. Since $\epsilon \le \epsilon_0$ on $[\epsilon_1, \epsilon_0]$, the integrands of T_q and $T_{q'}$ differ by a multiplicative $1 + O(\epsilon_0)$ factor, giving $T_q/T_{q'} = 1 + O(\epsilon_0)$. □

E Proofs and Pseudocode for Section 4: Monte Carlo Estimators

Theorem 4.1. *[Consistency and bias expansion] Fix a supervised example $(\mathbf{x}^*, \mathbf{y}^*)$ and assume:*

1. *$P_{\boldsymbol{\theta}} > 0$;*
2. *$\mathbb{E}[\|g_m\|^2] < \infty$;*
3. *$w_m \geq \epsilon$ a.s. for some $\epsilon > 0$.*

Then for any fixed $q \in [0, 1]$, the estimator is consistent: $\hat{\nabla}_{\boldsymbol{\theta}} \ell_q \xrightarrow{a.s.} \nabla_{\boldsymbol{\theta}} \ell_q$ as $M \to \infty$. Moreover, the leading-order bias is

$$\mathbb{E}\Big[\hat{\nabla}_{\boldsymbol{\theta}} \ell_q\Big] - \nabla_{\boldsymbol{\theta}} \ell_q \;=\; \frac{q}{M P_{\boldsymbol{\theta}}^{q+1}} \big[\tfrac{q+1}{2} \nabla_{\boldsymbol{\theta}} \ell_1 \, \mathbf{Var}(w_m) - \mathbf{Cov}(g_m, w_m)\big] + O(M^{-2}) \quad \text{as } M \to \infty. \tag{7}$$

Under additionally bounded marginal and per-trajectory scores ($\|\nabla_{\boldsymbol{\theta}} \log P_{\boldsymbol{\theta}}\| \leq C$, $\|\nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{y}^* \mid \mathbf{x}^*)\| \leq C'$), the bracketed term is $O(P_{\boldsymbol{\theta}})$, so the bias simplifies to $O(q/M P_{\boldsymbol{\theta}}^q)$.

Proof. We write

$$\mu_w \triangleq \mathbb{E}[w_m] = P_{\boldsymbol{\theta}}, \qquad \mu_g \triangleq \mathbb{E}[g_m] = \nabla_{\boldsymbol{\theta}} \ell_0(\boldsymbol{\theta}; \mathbf{x}^*, \mathbf{y}^*).$$

Define the smooth map

$$f(a, b) \triangleq b \, a^{-q},$$

for $a > 0$. Then

$$\hat{\nabla}_{\boldsymbol{\theta}} \ell_q(q, \boldsymbol{\theta}; \mathbf{x}^*, \mathbf{y}^*, M) = f(\bar{w}_M, \bar{g}_M),$$

while the target gradient is

$$\nabla_{\boldsymbol{\theta}} \ell_q(\boldsymbol{\theta}, q; \mathbf{x}^*, \mathbf{y}^*) = f(\mu_w, \mu_g) = \mu_g \, \mu_w^{-q}.$$

Almost sure convergence follows from the Strong Law of Large Numbers, since $\bar{w}_M \to \mu_w$ and $\bar{g}_M \to \mu_g$ almost surely, and f is continuous at (μ_w, μ_g) because $\mu_w = P_{\boldsymbol{\theta}} > 0$.

For the bias expansion, we exploit the linearity of f in its second argument: $f(a, b) = b \, a^{-q}$, so

$$\begin{aligned} f(\bar{w}_M, \bar{g}_M) &= \bar{g}_M \cdot h(\bar{w}_M) \\ &= \underbrace{\mu_g \, h(\bar{w}_M)}_{\text{first piece}} + \underbrace{(\bar{g}_M - \mu_g) \, h(\bar{w}_M)}_{\text{second piece}}, \end{aligned}$$

where $h(a) \triangleq a^{-q}$ is a scalar function whose derivatives $h^{(k)}(a) = (-q)(-q-1)\cdots(-q-k+1) \, a^{-(q+k)}$ depend only on a.

First piece. Expand $h(\bar{w}_M)$ to *third* order around μ_w, with $h'(a) = -qa^{-q-1}$, $h''(a) = q(q+1)a^{-q-2}$, $h'''(a) = -q(q+1)(q+2)a^{-q-3}$:

$$\begin{aligned} h(\bar{w}_M) = \; & \underbrace{h(\mu_w)}_{\mathbb{E}[\cdot] = \mu_w^{-q}} + \underbrace{h'(\mu_w)(\bar{w}_M - \mu_w)}_{\mathbb{E}[\cdot] = 0} + \underbrace{\tfrac{1}{2} h''(\mu_w)(\bar{w}_M - \mu_w)^2}_{\mathbb{E}[\cdot] = \frac{q(q+1)}{2M} \mu_w^{-q-2} \mathbf{Var}(w_m)} \\ & + \underbrace{\tfrac{1}{6} h'''(\mu_w)(\bar{w}_M - \mu_w)^3}_{\mathbb{E}[\cdot] = O(M^{-2}) \text{ via } \kappa_3/M^2} + \underbrace{R_M^{(1)}}_{\text{4th-order}}. \end{aligned}$$

Therefore:

$$\begin{aligned} \mu_g \; \mathbb{E}[h(\bar{w}_M)] = \mu_g \, \mu_w^{-q} + \frac{q(q+1)}{2M} \, \mu_g \, \mu_w^{-q-2} \, \mathbf{Var}(w_m) \\ + O(M^{-2}) + \mu_g \; \mathbb{E}[R_M^{(1)}]. \end{aligned}$$

Second piece. The factor $(\bar{g}_M - \mu_g) = O_p(M^{-1/2})$, so a *second*-order expansion of $h(\bar{w}_M)$ suffices. Multiplying $(\bar{g}_M - \mu_g)$ by each term of the expansion and taking expectations:

$$\begin{aligned}
&\mathbb{E}[(\bar{g}_M - \mu_g)\, h(\bar{w}_M)] \\
&= \underbrace{h(\mu_w)\ \mathbb{E}[\bar{g}_M - \mu_g]}_{=0} + \underbrace{h'(\mu_w)\ \mathbb{E}[(\bar{g}_M - \mu_g)(\bar{w}_M - \mu_w)]}_{=-\frac{q}{M}\mu_w^{-q-1}\mathbf{Cov}(g_m, w_m)} \\
&\quad + \underbrace{\tfrac{1}{2}h''(\mu_w)\ \mathbb{E}[(\bar{g}_M - \mu_g)(\bar{w}_M - \mu_w)^2]}_{=O(M^{-2}) \text{ via i.i.d. expansion}} + \underbrace{\mathbb{E}[R_M^{(2)}]}_{\text{3rd-order remainder}} .
\end{aligned}$$

For the cross moment, expand $\mathbb{E}[(\bar{g}_M - \mu_g)(\bar{w}_M - \mu_w)^2] = M^{-3}\sum_{i,j,k}\mathbb{E}[(g_i - \mu_g)(w_j - \mu_w)(w_k - \mu_w)]$. By independence, the only nonzero index pattern is $i = j = k$ (all others vanish because $\mathbb{E}[g_i - \mu_g] = 0$ or $\mathbb{E}[w_j - \mu_w] = 0$). The M surviving terms give $\mathbb{E}[(g_m - \mu_g)(w_m - \mu_w)^2]/M^2 = O(M^{-2})$, since $|(w_m - \mu_w)^2| \le 1$ and $\mathbb{E}[\|g_m\|] < \infty$ (Assumption 2). The remainder has the form $R_M^{(2)} = (\bar{g}_M - \mu_g) \cdot O(|\bar{w}_M - \mu_w|^3)$.

Combining. Adding the two pieces and substituting $\mu_w = P_{\boldsymbol{\theta}}$, $\mu_g = \nabla_{\boldsymbol{\theta}}\ell_0$, $\nabla_{\boldsymbol{\theta}}\ell_1 = {}^{\nabla_{\boldsymbol{\theta}}\ell_0}/_{P_{\boldsymbol{\theta}}}$:

$$\begin{aligned}
\mathbb{E}\Big[\hat{\nabla}_{\boldsymbol{\theta}}\ell_q(q, \boldsymbol{\theta}; \mathbf{x}^*, \mathbf{y}^*, M)\Big] &= \nabla_{\boldsymbol{\theta}}\ell_q(\boldsymbol{\theta}, q; \mathbf{x}^*, \mathbf{y}^*) \\
&\quad + \frac{q}{M P_{\boldsymbol{\theta}}^{q+1}} \cdot \left[\frac{q+1}{2}\nabla_{\boldsymbol{\theta}}\ell_1(\boldsymbol{\theta}; \mathbf{x}^*, \mathbf{y}^*)\, \mathbf{Var}(w_m) - \mathbf{Cov}(g_m, w_m)\right] \\
&\quad + \mathbb{E}[R_M],
\end{aligned} \tag{16}$$

where $R_M = \mu_g R_M^{(1)} + R_M^{(2)}$.

Remainder bound. Write $\mathbb{E}[R_M] = \mathbb{E}[R_M \cdot \mathbf{1}_A] + \mathbb{E}[R_M \cdot \mathbf{1}_{A^c}]$ where $A = \{\bar{w}_M \ge P_{\boldsymbol{\theta}}/2\}$.

On A. The derivatives of h are bounded on $\{a \ge P_{\boldsymbol{\theta}}/2\}$: $|h^{(k)}(a)| \le C_k$.

For $R_M^{(1)}$ (the *fourth*-order scalar remainder), the integral form gives $|R_M^{(1)}| \le C_4|\bar{w}_M - \mu_w|^4$ on A. Since $w_m \in [0,1]$, $\mathbb{E}[|\bar{w}_M - \mu_w|^4] = O(M^{-2})$, so $\mathbb{E}[|R_M^{(1)}| \cdot \mathbf{1}_A] = O(M^{-2})$.

For $R_M^{(2)} = (\bar{g}_M - \mu_g) \cdot O(|\bar{w}_M - \mu_w|^3)$ on A (the *third*-order remainder from the second piece, a vector quantity), Cauchy–Schwarz gives $\mathbb{E}[\|R_M^{(2)}\| \cdot \mathbf{1}_A] \le C_3\sqrt{\mathbb{E}[\|\bar{g}_M - \mu_g\|^2]}\sqrt{\mathbb{E}[(\bar{w}_M - \mu_w)^6]} = O(M^{-1/2})\, O(M^{-3/2}) = O(M^{-2})$, using Assumption 2 and the boundedness of w_m.

On A^c. Assumption 3 gives $\bar{w}_M \ge \epsilon > 0$, so $|h(\bar{w}_M)| \le \epsilon^{-q}$ everywhere and $\|f(\bar{w}_M, \bar{g}_M)\| \le \epsilon^{-q}\|\bar{g}_M\|$. Therefore $\|R_M\| \le \|f(\bar{w}_M, \bar{g}_M)\| + \|T_M\| \le C\,\epsilon^{-q}(1 + \|\bar{g}_M\|)$, where T_M collects the (bounded) Taylor terms. Again by Cauchy–Schwarz,

$$\mathbb{E}[\|R_M\| \cdot \mathbf{1}_{A^c}] \le C\,\epsilon^{-q}\sqrt{\mathbb{E}[(1 + \|\bar{g}_M\|)^2]}\,\sqrt{P(A^c)}.$$

The first factor is $O(1)$ by Assumption 2. For the second, since $w_m \in [0,1]$ are i.i.d. with mean $P_{\boldsymbol{\theta}}$, Hoeffding's inequality with $t = P_{\boldsymbol{\theta}}/2$ gives $P(A^c) = P(\bar{w}_M - P_{\boldsymbol{\theta}} \le -P_{\boldsymbol{\theta}}/2) \le \exp(-MP_{\boldsymbol{\theta}}^2/2)$. Thus $\mathbb{E}[\|R_M\| \cdot \mathbf{1}_{A^c}]$ decays faster than any polynomial in M.

Combining: $\mathbb{E}[R_M] = O(M^{-2})$, so the leading-order bias is the explicit formula above.

Bound on the bracketed coefficient. In Equation (16), the prefactor ${}^{q}/_{(MP_{\boldsymbol{\theta}}^{q+1})}$ has $P_{\boldsymbol{\theta}}^{-(q+1)}$ scaling, but the bracket $[{}^{(q+1)}/_2\,\nabla_{\boldsymbol{\theta}}\ell_1\,\mathbf{Var}(w_m) - \mathbf{Cov}(g_m, w_m)]$ scales as $O(P_{\boldsymbol{\theta}})$, so one factor of $P_{\boldsymbol{\theta}}$ cancels. Specifically:

- $\mathbf{Var}(w_m) \le \mathbb{E}[w_m^2] \le \mathbb{E}[w_m] = P_{\boldsymbol{\theta}}$ since $w_m \in [0,1]$.
- $\nabla_{\boldsymbol{\theta}}\ell_1 = -\nabla_{\boldsymbol{\theta}}\log P_{\boldsymbol{\theta}} = -s$ is bounded under the bounded-score assumption used in Theorem 3.1.

- Under bounded per-trajectory score $\|\nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{y}^* \mid \mathbf{x}^*)\| \leq C'$ (which follows from bounded weights and Lipschitz activations), $\|g_m\| \leq C' w_m$, and Cauchy–Schwarz gives $\|\mathbf{Cov}(g_m, w_m)\| \leq \sqrt{\mathbf{Var}(g_m)\,\mathbf{Var}(w_m)} \leq \sqrt{C'^2 P_{\boldsymbol{\theta}} \cdot P_{\boldsymbol{\theta}}} = O(P_{\boldsymbol{\theta}})$.

Hence the bracket is bounded by ${}^{(q+1)}/_2\, O(P_{\boldsymbol{\theta}}) + O(P_{\boldsymbol{\theta}}) = O(P_{\boldsymbol{\theta}})$ (the ${}^{(q+1)}/_2$ multiplier is bounded by 1 for $q \in [0, 1]$ and absorbs into the constant), and the leading-order bias is ${}^{q}/_{(M P_{\boldsymbol{\theta}}^{q+1})} \cdot O(P_{\boldsymbol{\theta}}) = O({}^{q}/_{(M P_{\boldsymbol{\theta}}^{q})})$, yielding Equation (7). The bias scales with the same $P_{\boldsymbol{\theta}}^{-q}$ exponent as the cold-start amplification factor. □

E.1 RLOO CONTROL VARIATE DERIVATION

We derive the RLOO estimator (19) from the plug-in estimator (6). Using the chain rule, g_m from (4) decomposes into a score-function term and a pathwise term:

$$g_m = -\, w_m\, \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{z}^{(m)} \mid \mathbf{x}^*) - \nabla_{\boldsymbol{\theta}} w_m. \tag{17}$$

Substituting into the plug-in estimator isolates the score-function component:

$$\hat{\nabla}_{\boldsymbol{\theta}}^{\text{plug-in}} \ell_q = \frac{1}{M} \sum_{m=1}^{M} \left[\frac{-w_m}{(\bar{w}_M)^q} \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{z}^{(m)} \mid \mathbf{x}^*) - \frac{\nabla_{\boldsymbol{\theta}} w_m}{(\bar{w}_M)^q} \right]. \tag{18}$$

Since $\mathbb{E}[\nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{z}^{(m)} \mid \mathbf{x}^*)] = 0$, we can subtract any baseline from the score-function coefficient $-w_m/(\bar{w}_M)^q$ without changing the expected value, provided the baseline does not depend on $\mathbf{z}^{(m)}$.

We use a leave-one-out approximation. Let $\bar{w}_{\neg m} = \frac{1}{M-1} \sum_{j \neq m} w_j$. Replacing w_m with $\bar{w}_{\neg m}$ in the coefficient, the batch mean collapses to $\bar{w}_{\neg m}$, giving a surrogate coefficient of $-(\bar{w}_{\neg m})^{1-q}$. Subtracting this baseline yields the RLOO estimator

$$\hat{\nabla}_{\boldsymbol{\theta}}^{\text{RLOO}} \ell_q = \frac{1}{M} \sum_{m=1}^{M} \left[\underbrace{-\left(\frac{w_m}{(\bar{w}_M)^q} - (\bar{w}_{\neg m})^{1-q} \right)}_{\text{centered weight}} \cdot \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{z}^{(m)} \mid \mathbf{x}^*) - \frac{\nabla_{\boldsymbol{\theta}} w_m}{(\bar{w}_M)^q} \right]. \tag{19}$$

Endpoint recovery. At $q = 0$, the centered weight evaluates to $w_m - \bar{w}_{\neg m}$, and the score-function term becomes $-(w_m - \bar{w}_{\neg m}) \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{z}^{(m)} \mid \mathbf{x}^*)$, exactly recovering the REINFORCE leave-one-out (RLOO) estimator standard in RLVR. At $q = 1$, the centered weight is $w_m/\bar{w}_M - 1$; since $\sum_{m=1}^{M} (w_m/\bar{w}_M - 1) = 0$, this acts as a self-normalizing baseline that strictly centers the importance weights across the batch.

Proposition E.1 (RLOO bias preservation)**.** *Under the assumptions of Theorem 4.1, the RLOO estimator* (19) *satisfies the same bias expansion as the plug-in estimator* (6)*.*

Proof. The RLOO estimator (19) differs from the plug-in estimator (18) by subtracting $(\bar{w}_{\neg m})^{1-q}$ from the score-function coefficient $w_m/(\bar{w}_M)^q$ for each sample m. Denoting $s_m = \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{z}^{(m)} \mid \mathbf{x}^*)$, the difference in expectations is

$$\Delta = \frac{1}{M} \sum_{m=1}^{M} \mathbb{E}[(\bar{w}_{\neg m})^{1-q}\, s_m].$$

Since $\bar{w}_{\neg m} = \frac{1}{M-1} \sum_{j \neq m} w_j$ is a function of $\{\mathbf{z}^{(j)}\}_{j \neq m}$ only, and $s_m = \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{z}^{(m)} \mid \mathbf{x}^*)$ is a function of $\mathbf{z}^{(m)}$ only, the independence of the i.i.d. samples gives

$$\mathbb{E}[(\bar{w}_{\neg m})^{1-q}\, s_m] = \underbrace{\mathbb{E}[(\bar{w}_{\neg m})^{1-q}]}_{} \cdot \underbrace{\mathbb{E}[s_m]}_{=0} = 0,$$

where $\mathbb{E}[s_m] = \mathbb{E}_{\mathbf{z} \sim p_{\boldsymbol{\theta}}}[\nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathbf{x}^*)] = 0$ is the standard score-function identity. Therefore $\Delta = 0$ and the two estimators have identical expectations for every M. □

E.2 Endpoint recovery

Proposition E.2 (Endpoint recovery for GARL and PAFT)**.** *Fix a supervised example* $(\mathbf{x}^*, \mathbf{y}^*)$ *with* $P_{\boldsymbol{\theta}} > 0$*.*

1. ***GARL at*** $q = 0$ *recovers Rao–Blackwellized REINFORCE (Williams, 1992; Zhou et al., 2026):*

$$\hat{\nabla}_{\boldsymbol{\theta}} \ell_q \big|_{q=0} = \bar{g}_M = \frac{1}{M} \sum_{m=1}^{M} \big(-w_m \, \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{z}^{(m)}, \mathbf{y}^* \mid \mathbf{x}^*) \big),$$

which is unbiased for $\nabla_{\boldsymbol{\theta}} \ell_0$ *by Equation* (5)*. Each* g_m *marginalizes out the output* $\mathbf{y}$ *given* $\mathbf{z}^{(m)}$ *analytically via* $w_m = p_{\boldsymbol{\theta}}(\mathbf{y}^* \mid \mathbf{x}^*, \mathbf{z}^{(m)})$*, rather than relying on a sampled output and binary reward.*

2. ***GARL at*** $q = 1$ *recovers the IWAE gradient estimator (Burda et al., 2015), a self-normalized importance sampling (SNIS) estimator for* $\nabla_{\boldsymbol{\theta}} \log P_{\boldsymbol{\theta}}$*:*

$$\hat{\nabla}_{\boldsymbol{\theta}} \ell_q \big|_{q=1} = \frac{\bar{g}_M}{\bar{w}_M} = \frac{\sum_m w_m \left(-\nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{z}^{(m)}, \mathbf{y}^* \mid \mathbf{x}^*) \right)}{\sum_m w_m}.$$

3. ***PAFT at*** $q = 0$ *reduces to posterior-resampled SFT scaled by* $P_{\boldsymbol{\theta}}$*:*

$$\hat{\nabla}_{\mathrm{PAFT}} \big|_{q=0} = -\bar{w}_M \cdot \frac{1}{K} \sum_{k=1}^{K} \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{z}^{(r_k)}, \mathbf{y}^* \mid \mathbf{x}^*).$$

The factor $\bar{w}_M \approx P_{\boldsymbol{\theta}}$ *downweights hard instances so aggressively that this endpoint is overly conservative in practice. Unlike the other three endpoints, it does not correspond to a standard method.*

4. ***PAFT at*** $q = 1$ *recovers the EM gradient update with E-step posterior samples (Dempster et al., 1977) / TRICE (Phan et al., 2023):*

$$\hat{\nabla}_{\mathrm{PAFT}} \big|_{q=1} = -\frac{1}{K} \sum_{k=1}^{K} \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{z}^{(r_k)}, \mathbf{y}^* \mid \mathbf{x}^*).$$

The attenuation vanishes: $(\bar{w}_M)^{1-1} = 1$*, so all instances contribute equally, and the gradient is uniform SFT on approximate posterior samples.*

Proof. Each case follows by substituting $q = 0$ or $q = 1$ into the GARL estimator (6) or PAFT estimator (9) and simplifying $(\bar{w}_M)^0 = 1$. □

E.3 PAFT bias and variance

Proposition E.3 (PAFT has the same bias as GARL)**.** *Under the assumptions of Theorem 4.1,* $\mathbb{E}[\hat{\nabla}_{\mathrm{PAFT}}] = \mathbb{E}[\hat{\nabla}_{\mathrm{GARL}}]$ *for all* M*. In particular, the PAFT estimator inherits the same leading bias expansion as in Equation* (7)*, simplifying to* $O(q/MP_{\boldsymbol{\theta}}^q)$ *under bounded marginal and per-trajectory scores.*

Proof. Conditional on the prior samples $\mathrm{pool} = \{(\mathbf{z}^{(m)}, w_m)\}_{m=1}^{M}$, the factor $(\bar{w}_M)^{1-q}$ is deterministic. The importance-resampled average satisfies

$$\mathbb{E}\left[\frac{1}{K} \sum_{k=1}^{K} f(\mathbf{z}^{(r_k)}) \;\middle|\; \mathrm{pool} \right] = \sum_{m=1}^{M} \frac{w_m}{\sum_j w_j} \, f(\mathbf{z}^{(m)}) = \hat{\mu}_{\mathrm{SNIS}},$$

where $f(\mathbf{z}) = \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{y}^* \mid \mathbf{x}^*)$. Therefore

$$\begin{aligned}
\mathbb{E}[\hat{\nabla}_{\mathrm{PAFT}} \mid \mathrm{pool}] &= -(\bar{w}_M)^{1-q} \cdot \hat{\mu}_{\mathrm{SNIS}} \\
&= -(\bar{w}_M)^{1-q} \cdot \frac{\sum_m w_m f_m}{M \bar{w}_M} \\
&= \frac{1}{(\bar{w}_M)^q} \cdot \frac{1}{M} \sum_m (-w_m f_m) \\
&= \frac{\bar{g}_M}{(\bar{w}_M)^q} = \hat{\nabla}_{\mathrm{GARL}}.
\end{aligned}$$

Taking outer expectations by the tower property: $\mathbb{E}[\hat{\nabla}_{\mathrm{PAFT}}] = \mathbb{E}[\hat{\nabla}_{\mathrm{GARL}}]$. □

Proposition E.4 (GARL has strictly lower variance than PAFT)**.** *Under the same setup,* $\mathbf{Var}(\hat{\nabla}_{\mathrm{PAFT}}) \geq \mathbf{Var}(\hat{\nabla}_{\mathrm{GARL}})$*, with equality only when* $\mathbf{Var}(\hat{\nabla}_{\mathrm{PAFT}} \mid \mathrm{pool}) = 0$ *almost surely.*

Proof. By Proposition E.3, $\mathbb{E}[\hat{\nabla}_{\mathrm{PAFT}} \mid \mathrm{pool}] = \hat{\nabla}_{\mathrm{GARL}}$. The law of total variance gives

$$\begin{aligned}
\mathbf{Var}(\hat{\nabla}_{\mathrm{PAFT}}) &= \mathbf{Var}\big(\mathbb{E}[\hat{\nabla}_{\mathrm{PAFT}} \mid \mathrm{pool}]\big) + \mathbb{E}\big[\mathbf{Var}(\hat{\nabla}_{\mathrm{PAFT}} \mid \mathrm{pool})\big] \\
&= \mathbf{Var}(\hat{\nabla}_{\mathrm{GARL}}) + \underbrace{\mathbb{E}\big[\mathbf{Var}(\hat{\nabla}_{\mathrm{PAFT}} \mid \mathrm{pool})\big]}_{\geq 0},
\end{aligned}$$

with equality iff $\mathbf{Var}(\hat{\nabla}_{\mathrm{PAFT}} \mid \mathrm{pool}) = 0$ a.s. This holds when, for each pool realization, all resampled trajectories produce the same gradient — e.g., when a single trajectory dominates the importance weights. In the non-degenerate case, the inequality is strict. □

E.4 Pseudocode for GARL and PAFT

Update normalization. The per-sample weight $w_m/(\bar{w}_M)^q$ (the effective reward under the RL view) has maximum M^q, so the centered advantage c_m in Equation (19) can range up to M^q in magnitude. To keep the per-sample advantage uniformly bounded as q varies, the algorithms Algorithms 1 and 2 divide by M^q, yielding $c_m/M^q \in [-1, 1]$. The mathematical estimators Equations (9) and (19) target $\nabla_{\boldsymbol{\theta}} \ell_q$ directly; the algorithm-side $1/M^q$ is equivalent to applying a q-independent learning rate to the bounded-advantage form (vs. a q-dependent learning rate to the unscaled form).

Algorithm 1 GARL: per-example J_Q gradient with RLOO control variate. *Numerical stability*: $w_m = \prod_t p_{\boldsymbol{\theta}}(y_t^* \mid \cdot)$ underflows for long $\mathbf{y}^*$ in linear-space arithmetic, so w_m, $\bar{w}_M$, $\bar{w}_{\neg m}$, and c_m should be computed in log-space (e.g., LogSumExp); the pathwise term $\nabla_{\boldsymbol{\theta}} w_m/(\bar{w}_M)^q$ should be implemented as $\frac{w_m}{(\bar{w}_M)^q} \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{y}^* \mid \mathbf{x}^*, \mathbf{z}^{(m)})$ (log-derivative trick), with the coefficient computed in log-space before being applied to the log-probability gradient.

Require: Example $(\mathbf{x}^*, \mathbf{y}^*)$, interpolation parameter $q \in [0, 1]$, number of latent samples $M \geq 2$ (for the leave-one-out baseline)

Sample latent trajectories $\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(M)} \sim p_{\boldsymbol{\theta}}(\cdot \mid \mathbf{x}^*)$
for $m = 1, \ldots, M$ **do**
  $w_m \leftarrow p_{\boldsymbol{\theta}}(\mathbf{y}^* \mid \mathbf{x}^*, \mathbf{z}^{(m)})$ ▷ likelihood weight
  $\nabla_{\boldsymbol{\theta}} w_m \leftarrow \nabla_{\boldsymbol{\theta}}\, p_{\boldsymbol{\theta}}(\mathbf{y}^* \mid \mathbf{x}^*, \mathbf{z}^{(m)})$ ▷ pathwise gradient of output likelihood
end for
$\bar{w}_M \leftarrow \frac{1}{M} \sum_{m=1}^M w_m$ ▷ batch mean (estimates $P_{\boldsymbol{\theta}}$)
for $m = 1, \ldots, M$ **do**
  $\bar{w}_{\neg m} \leftarrow \frac{1}{M-1} \sum_{j \neq m} w_j$ ▷ leave-one-out mean
  $c_m \leftarrow \frac{w_m}{(\bar{w}_M)^q} - (\bar{w}_{\neg m})^{1-q}$ ▷ centered weight (RLOO baseline)
  $\hat{g}_m \leftarrow -c_m \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{z}^{(m)} \mid \mathbf{x}^*) - \frac{\nabla_{\boldsymbol{\theta}} w_m}{(\bar{w}_M)^q}$ ▷ score-function + pathwise terms
end for
return $\hat{g} \leftarrow \frac{1}{M^q} \cdot \frac{1}{M} \sum_{m=1}^M \hat{g}_m$ ▷ per-example gradient estimate, rescaled by $1/M^q$ to bound per-sample advantage uniformly in q

Algorithm 2 PAFT: per-example J_Q gradient via importance resampling. *Numerical stability*: the resampling step should be implemented with a categorical distribution parameterized by log-weights, e.g., $\mathrm{Categorical}(\mathrm{logits} = [\log w_1, \ldots, \log w_M])$, to avoid division-by-zero when all w_m underflow.

Require: Example $(\mathbf{x}^*, \mathbf{y}^*)$, interpolation parameter $q \in [0, 1]$, prior samples M, resampled trajectories K

1: Sample latent trajectories $\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(M)} \sim p_{\boldsymbol{\theta}}(\cdot \mid \mathbf{x}^*)$
2: **for** $m = 1, \ldots, M$ **do**
3: $\quad w_m \leftarrow p_{\boldsymbol{\theta}}(\mathbf{y}^* \mid \mathbf{x}^*, \mathbf{z}^{(m)})$ ▷ likelihood weight (same as GARL)
4: **end for**
5: $\bar{w}_M \leftarrow \frac{1}{M} \sum_{m=1}^{M} w_m$ ▷ batch mean (estimates $P_{\boldsymbol{\theta}}$)
6: Resample indices $r_1, \ldots, r_K \sim \mathrm{Categorical}(w_1 / \sum_j w_j, \ldots, w_M / \sum_j w_j)$
7: $\hat{g} \leftarrow -\frac{(\bar{w}_M)^{1-q}}{M^q K} \sum_{k=1}^{K} \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\mathbf{z}^{(r_k)}, \mathbf{y}^* \mid \mathbf{x}^*)$ ▷ attenuated SFT on coherent rationales, rescaled by $1/M^q$ for advantage bounding
8: **return** $\hat{g}$ ▷ rescaled per-example gradient estimate (matching Alg. 1's $1/M^q$ convention)

F Additional Experimental Details

Subset construction. We sample subsets from Huggingface datasets: FinQA from `dreamerdeo/finqa`, HotPotQA from `hotpotqa/hotpot_qa`, and MuSiQue from `bdsaglam/musique`. We construct training, validation, and test subsets by retaining instances whose pre-tokenization input length (in characters) falls below predefined caps. The caps are 8000, 4000, and 10000 characters for FinQA, HotPotQA, and MuSiQue respectively. The resulting train/val/test subset sizes are 6145/872/1132, 9067/342/343, and 9985/579/445 for the 3 datasets respectively.

Prompts. Every input is a two-turn chat, a system message and a user message (Figure 2), rendered with the model's chat template with thinking enabled. With prompted inputs, the user message is a task instruction, a blank line, and the example; without prompts, as at cold start, it is the example alone, formatted the same way. The system message is the same in both settings. The Gemma 4 E4B instructions do not mention `<think>` tags, since Gemma delimits its thinking with special tokens. The completion that GARL and PAFT score, $\mathtt{t}_{\text{end}}$`\n<answer>`$\mathbf{y}^*$`</answer>` (Section 5.1), does not depend on the prompt.

Training setup. We do not apply KL regularization to a reference policy, following the VeriFree setup (Zhou et al., 2026); Liu et al. (2025) found KL does not improve performance in this regime. Per-rationale token budgets force the thinking-end token (`</think>` for Qwen, `<channel|>` for Gemma) once the budget is exhausted (Muennighoff et al., 2025); see *Generation lengths* below. We use the AdamW optimizer (Loshchilov & Hutter, 2019) for all experiments. Training batch size is 64, and learning rate is set to 5×10^{-7} for Qwen 3 0.6B (higher learning rate was unstable in preliminary experiments), and 1×10^{-6} for Qwen 3 8B experiments respectively. Gemma 4 E4B runs follow the Qwen 3 0.6B protocol, including its learning rate. We train for 2 epochs for all datasets, with a constant learning rate (no warmup or decay). Rollouts during training use temperature 1.0 (with top-k/top-p sampling disabled).

Model selection. We evaluate on the validation sets every 50 steps, and also at the end of training. We select the checkpoint that performs best on the `m@16` metric.

Generation lengths. We cap the maximum generation lengths to be 4096 for FinQA, 3072 for HotPotQA, and 2048 for MuSiQue. In addition, we allocate 128 tokens at the end of generation for the answer.

Compute. Each run uses one 8-GPU (NVIDIA A100 80GB) machine. A training step, excluding validation, takes about 2–4.5 minutes for Qwen 3 0.6B, 3–5 minutes for Qwen 3 8B, and 5–8 minutes for Gemma 4 E4B.

System message (all models, with and without prompts)

```
You are a helpful assistant.
```

Instruction, FinQA (Qwen 3)

```
You are given a table and surrounding texts from a financial document,
    ↪ followed by a question. Think step by step, and provide your
    ↪ brief and concise reasoning between <think> and </think> tokens.
    ↪ Then, provide the final answer (i.e., just one numerical value),
    ↪ bracketed between <answer> and </answer> immediately after.
```

Instruction, FinQA (Gemma 4)

```
You are given a table and surrounding texts from a financial document,
    ↪ followed by a question. Think step by step, and then provide the
    ↪ final answer (i.e., just one numerical value), bracketed between
    ↪ <answer> and </answer>.
```

Instruction, HotPotQA and MuSiQue (Qwen 3)

```
You are given a question and a set of context paragraphs. Answer the
    ↪ question based on the provided context. Think step by step, and
    ↪ provide your reasoning between <think> and </think> tokens.
    ↪ Then, provide your answer between <answer> and </answer>.
```

Instruction, HotPotQA (Gemma 4)

```
You are given a question and a set of context paragraphs. Answer the
    ↪ question based on the provided context. Think step by step, and
    ↪ then provide your answer between <answer> and </answer>.
```

Example, FinQA

```
pre_text:
row 1: {sentence}
row 2: {sentence}
...

table:
row 1: {cell} | {cell} | ...
row 2: {cell} | {cell} | ...
...

post_text:
['{sentence}', '{sentence}', ...]
question:
{question}
```

Example, HotPotQA and MuSiQue

```
Context:
[{title}] {paragraph}

[{title}] {paragraph}

...

Question: {question}
```

Figure 2: Model inputs, verbatim from the data files; training, validation, and test use the same templates. With prompts, the user message is the instruction, a blank line, and the example; without prompts, it is the example alone. Braces mark fields filled from each example, and ↪ marks a line wrapped for display.

Table 3: Test m@16 (Qwen 3 0.6B; test at the best-validation checkpoint; evaluation uses the reshaped reward for all rows). Base = un-adapted model with the same prompts. GRPO is trained with the matched reward (3-seed mean, as in Table 1) or the reshaped reward (3-seed mean on FinQA, 2-seed mean on HotPotQA, single-seed on MuSiQue). Unprompted GARL numbers are 3-seed mean from Table 1. All other rows are single-seed. GARL at q=0 recovers RB-RLOO (Zhou et al., 2026). Unprompted GARL entries at $q > 0$ on HotPotQA and MuSiQue are peak-before-collapse: validation accuracy later falls to zero, except that the HotPotQA q=0.75 run ended at step 89, before its next evaluation, after its gradient fell to near zero. FinQA GARL is stable. After their best checkpoints, the PAFT q=0.75 runs lose up to 30% (HotPotQA) and 41% (MuSiQue) of their peak validation accuracy but stay far from zero. Best per benchmark in bold.

Method	FinQA	HotPotQA	MuSiQue
Base (no training, prompted)	12.6	22.2	8.9
GRPO (matched reward, prompted)	27.8	34.0	15.4
GRPO (reshaped reward, prompted)	37.1	55.4	34.0
GARL ($q = 0$ (RB-RLOO), prompted)	38.3	21.6	9.1
GARL ($q = 0.25$, prompted)	**38.7**	22.9	24.3
GARL ($q = 0.75$, prompted)	37.6	46.8	19.7
PAFT ($q = 0.25$, prompted)	26.6	47.0	9.0
PAFT ($q = 0.75$, prompted)	28.6	47.9	17.5
GARL ($q = 0.75$, unprompted)	38.6	**57.4**	**35.6**

G ADDITIONAL RESULTS WITH PROMPTED INPUTS

The prompted setting (Section 5.1) tests whether GARL and PAFT help when $P_{\boldsymbol{\theta}}$ is not negligible and GRPO already makes progress; this section expands on RQ4 (Section 5.5). Table 3 reports test m@16 for Qwen 3 0.6B at the default learning rate (5×10^{-7}), with the same best-validation checkpoint selection as the main text.

J_Q estimators beat the matched GRPO baseline, but beat reshaped-reward GRPO only on FinQA. Against matched-reward GRPO, which trains toward the same success event as GARL and PAFT (Section 5.1), the best J_Q checkpoint gains +10.9 m@16 on FinQA (GARL, q=0.25), +13.9 on HotPotQA (PAFT, q=0.75), and +8.9 on MuSiQue (GARL, q=0.25, before collapse). Reshaped-reward GRPO, which trains on the evaluation's own reward function, is much stronger with prompted inputs: GARL beats it only on FinQA, by 0.5 to 1.6 points (37.6–38.7 vs. 37.1), and on HotPotQA and MuSiQue reshaped-reward GRPO scores above every GARL and PAFT entry (55.4 vs. at most 47.9; 34.0 vs. at most 24.3). On FinQA, GARL is stable at every tested q and m@16 is roughly flat across $q \in [0, 0.75]$, consistent with the bias and noise-fitting costs of high q (Theorem 4.1 and Proposition D.3) outweighing its amplification once reward is not sparse. At q=0, GARL recovers RB-RLOO, which beats matched-reward GRPO on FinQA (+10.5) but not on HotPotQA (−12.4) or MuSiQue (−6.3); raising q lifts the best checkpoint on those two benchmarks (HotPotQA 21.6 → 46.8 at q=0.75; MuSiQue 9.1 → 24.3 at q=0.25).

Prmopted GARL training destabilizes after its best checkpoint on HotPotQA and MuSiQue. On HotPotQA, GARL's validation accuracy falls to zero at q=0.25 (it peaks at step 50 at 30.6, falls to 9.7 at step 100, and is zero by step 150); the q=0.75 run ended at step 89, before its next evaluation, after its gradient norm had risen about 70-fold and then fallen to near zero as responses lengthened to about 2,700 tokens. On MuSiQue it falls to zero in every run that learned, between steps 200 and 250 at q=0.25 and between steps 50 and 100 at q=0.75. At q=0, GARL barely trains on either benchmark: its validation accuracy stays within two points of its starting value. In the collapses we could inspect (MuSiQue at the default learning rate, HotPotQA at half of it), responses run to the generation cap and the gradient goes to zero; where we tracked the gradient norm per step, it spikes first. Matched-reward GRPO also degrades late in training: on HotPotQA, validation m@16 peaks near 37.4 around step 100 and ends near 5.0; on MuSiQue it falls from about 15 to 3–5, and on FinQA by 4–8 points (two seeds each). Reshaped-reward GRPO instead improves throughout training on all three benchmarks and ends at its best checkpoint. GARL's failure is the most abrupt:

on MuSiQue, validation accuracy goes from its peak to exactly zero between consecutive evaluations, 50 steps apart, and on HotPotQA at $q{=}0.25$ within two.

PAFT degrades, but does not destabilize. PAFT's validation accuracy also declines after its best checkpoint, but gradually and far from zero. At $q{=}0.75$ it falls from 55.0 (step 100) to a low of 38.7 on HotPotQA (-30% relative) and from 25.1 (step 150) to 14.7 on MuSiQue (-41%), ending at 47.7 and 18.3; on FinQA the drop from peak is 20% at $q{=}0.25$ and 5% at $q{=}0.75$. Its responses also run to the generation cap on HotPotQA and MuSiQue, yet its accuracy stays well above zero. Under best-validation selection, PAFT leads GARL only on HotPotQA (47.9 vs. 46.8); GARL's pre-collapse checkpoint scores higher on MuSiQue (24.3 vs. 17.5), and GARL leads on FinQA at every q. PAFT's advantage is therefore robustness rather than a higher score: it degrades instead of collapsing, which matters when no validation set is available for checkpoint selection.

H Training Dynamics Figures

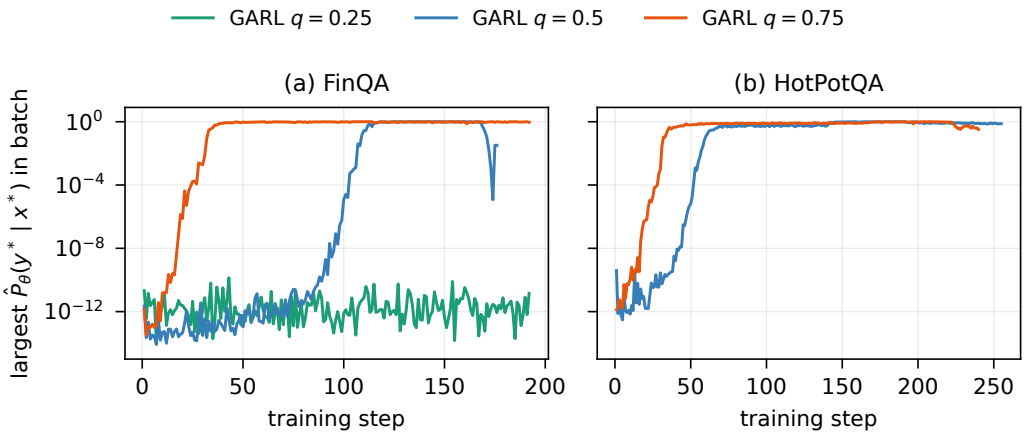


Figure 3: Cold-start escape on Gemma 4 E4B, one seed per run, on (a) FinQA and (b) HotPotQA: a lower bound on the largest per-prompt Monte Carlo estimate of $P_{\boldsymbol{\theta}}(\mathbf{y}^* \mid \mathbf{x}^*)$ in each training batch (log scale), which reflects the easiest prompt in the batch. With $w_m = p_{\boldsymbol{\theta}}(\mathbf{y}^* \mid \mathbf{x}^*, \mathbf{z}^{(m)})$, the estimate for rollout m's prompt is the leave-one-out mean $\bar{w}_{\neg m}$. The runs did not log w_m, so we recover this from the logged minimum advantage in each batch, $\min c_m / M^q$: since $c_m \geq -(\bar{w}_{\neg m})^{1-q}$, with equality when $w_m = 0$, the plotted $|\min c_m|^{1/(1-q)}$ is at most the largest $\bar{w}_{\neg m}$ in the batch. GARL at q=1 is omitted because its advantages depend only on ratios of the w_m. On both benchmarks, q=0.75 escapes (its estimate rises to near 1) before q=0.5, and both outperform prompted GRPO in Table 1; on FinQA, q=0.25 does not escape within the training budget, and its accuracy stays at zero. This is the ordering that Section 3 predicts for gradient flow from a small p_0: the escape time decreases with q ($T_q \geq T_{q'}$ for $q < q'$, and $T_q / T_{q'} \to \infty$ as $p_0 \to 0$ by Theorem 3.2), and by Theorem 3.1 it is at least of order $p_0^{-(1-q)}/(1-q)$, which is large at low q.

I LIMITATIONS AND FUTURE DIRECTIONS

Limitations. The majority of experiments, and all multi-seed results, use Qwen 3 0.6B; Qwen 3 8B and Gemma 4 E4B runs are single-seed, and all reported runs use a fixed q. The cold-start theorems are scale-agnostic, and the cold-start ordering replicates at Qwen 3 8B on all three benchmarks and on Gemma 4 E4B on FinQA and HotPotQA (Section 5). GARL can become unstable at our default learning rate, with prompted inputs (Section G). As implemented, J_Q estimators teacher-force one canonical completion, so unlike GRPO they do not benefit from a verifier that accepts other response formats. Annealed-q schedules, including the three-phase recipe below, are untested. The convergence analysis is stylized: the lower bound survives interference between examples and discrete minibatch steps, but assumes exact per-example gradients (Section D.4), the matching upper bound is for a single example under gradient flow, and both assume a bounded score and exact-match supervision; general rewards are open.

A three-phase post-training recipe. The continuum prescribes a regime-dependent recipe: at cold start ($P_{\boldsymbol{\theta}} \approx 0$), GARL at large q (≥ 0.75, scaling up with model size) breaches the $\Omega(1/p_0)$ bottleneck (PAFT degenerates here); with prompted inputs, PAFT at intermediate q is the more stable estimator; as $P_{\boldsymbol{\theta}} \to 1$, the bias shrinks and annealing $q \to 0$ recovers the unbiased RB-RLOO estimator. Validating these switches empirically is future work.

Multi-example dynamics. Apart from the lower bound (Section D.4), our convergence analysis considers a single example. Across examples, the dynamics on each p_i involve the kernel $K_{ij} = \nabla_{\boldsymbol{\theta}} P_i \cdot \nabla_{\boldsymbol{\theta}} P_j$. Its interplay with the q-dependent weighting P_j^{-q} (potentially via NTK theory) could give matching upper bounds and characterize how dataset-level coverage emerges from gradient-level amplification.

Annealing and richer posterior sampling. Principled schedule design adaptive to the current $P_{\boldsymbol{\theta}}$, and automatic switching between GARL and PAFT, remain open. PAFT's importance resampling from the prior pool fails at cold start (vanishing attenuation and particle degeneracy); learned proposals, MCMC, or infilling models conditioned on both $\mathbf{x}^*$ and $\mathbf{y}^*$ could extend PAFT to lower-$P_{\boldsymbol{\theta}}$ regimes.

BROADER IMPACTS

This work is methodological: we propose a loss family and corresponding gradient estimators for training reasoning language models, using publicly available checkpoints (Qwen 3) and benchmarks (FinQA, HotPotQA, MuSiQue); no new pre-trained models or datasets are released. The J_Q continuum and its estimators (GARL, PAFT) enable post-training without annotated rationales, lowering the data bar for adapting reasoning models to specialized domains, low-resource languages, or settings where rationale annotations are expensive or unavailable. As with any post-training improvement, our methods could in principle be applied to fine-tune models for harmful applications; the same dual-use considerations apply to any RL-based post-training method (*e.g.*, GRPO, RLHF), and our contributions at the level of the training objective remain compatible with existing safety-relevant training procedures.