CM-PRM: CONTROLLED-MARGIN CONTRASTIVE LEARNING FOR PROCESS REWARD MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Process reward models (PRMs) provide fine-grained step-level supervision for complex reasoning tasks. Existing discriminative PRMs offer strong step-level discrimination but rely on large-scale step-level annotations, whereas generative PRMs are more annotation-efficient and interpretable but often lose discriminative capability when trained with existing preference optimization objectives. To bridge this gap, we study generative PRM training from a conditional mutual information perspective and formulate an optimization objective by maximizing a variational lower bound to capture the dependence between process states and generated evaluations. From this perspective, we identify two potential failure modes of these objectives: *Synchronous Collapse*, where margin-driven optimization allows both chosen and rejected rewards to decrease as their margin grows, weakening process-level discrimination; and *Log-ratio Inflation*, where insufficient control over log-ratio scales can over-reinforce particular evaluation trajectories, encouraging overreliance on local analysis at the expense of other relevant evidence. To address these failure modes, we propose CM-PRM, a controlled-margin contrastive learning framework that decouples the optimization signals for correct and incorrect evaluation trajectories through a Pearson-χ^2 variational objective and introduces quadratic regularization to control log-ratio scales and stabilize reward outputs. Experiments on standard benchmarks and test-time scaling tasks show that CM-PRM outperforms compared PRMs overall; ablation studies and training-dynamics analyses show that its joint design mitigates simultaneous decline of chosen and rejected rewards and curbs excessive growth in log-ratio magnitudes.

1 INTRODUCTION

Complex reasoning tasks require models to generate long chains of intermediate steps, where early errors can derail the final answer. Outcome Reward Models (ORMs) supervise final answers but offer limited feedback on intermediate reasoning steps (Lightman et al., 2024; Wang et al., 2024b). Process Reward Models (PRMs) address this limitation by providing step-level evaluation and feedback. Among PRMs, discriminative PRMs(Zhang et al., 2025b; Xi et al., 2026) directly assign scalar rewards to reasoning steps, whereas generative PRMs (Zhao et al., 2026; Zhang et al., 2026a) produce evaluation trajectories comprising natural-language analyses and final correctness judgments (Figure 1). Although these evaluation trajectories make the judgment process inspectable and support data-efficient learning, generative PRMs still struggle to produce decisive, robust judgments. To this end, this paper focuses on enhancing the discriminative capability of generative PRMs.

To improve the discriminative capability of generative PRMs, recent work has explored supervised learning of evaluation trajectories and preference optimization over trajectories with correct and incorrect final judgments (She et al., 2025; Whitehouse et al., 2025). Nevertheless, reliable evaluation remains challenging. Figure 2 illustrates two evaluation failure phenomena: *insufficient verification*, where the model accepts a reasoning step without adequately checking its validity, and *overreliance on local analysis*, where it reaches a judgment based on local analysis without integrating relevant reasoning evidence. These phenomena motivate closer examination of the training objective: a larger margin between evaluation trajectories with correct and incorrect judgments does not by itself specify how strongly each trajectory is reinforced or suppressed. To provide this individual control, we propose CM-PRM, a controlled-margin contrastive learning framework for training generative PRMs. Next, we analyze these phenomena and derive CM-PRM's design principles.

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.

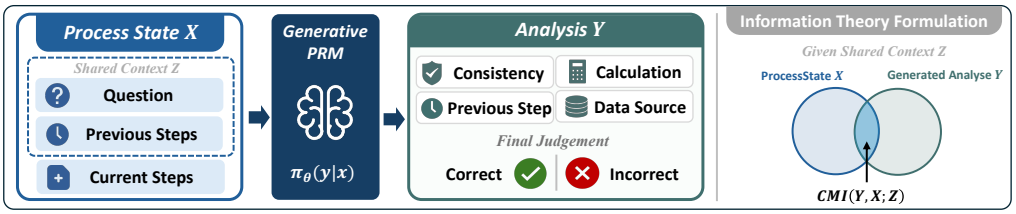


Figure 1: Overview of a generative PRM, which generates an analysis and a correctness judgment for the current reasoning step given a process state.

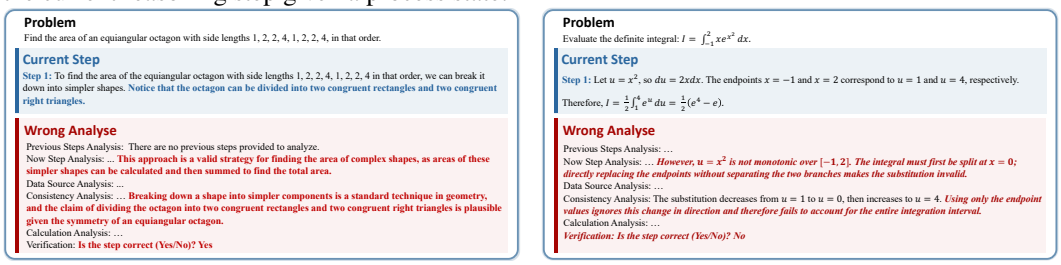


Figure 2: Examples of evaluation failure phenomena: insufficient verification (left) and overreliance on local analysis (right).

Q1: How can generative PRMs make well-verified, decisive judgments?

Insufficient verification highlights the need to increase the likelihood of evaluation trajectories with correct final judgments. Sampled rejected trajectories cannot exhaustively cover diverse incorrect evaluations. Suppressing these samples is insufficient; training must reinforce chosen evaluations with correct judgments. In DPO-style training, both gradient coefficients depend on the relative margin, allowing the objective to improve even as both log-ratios decline, provided the rejected one falls faster. Such simultaneous likelihood decreases, termed **Synchronous Collapse**, have been observed previously (Pal et al., 2024). Thus, preference separation alone does not ensure reinforcement of sampled chosen evaluations. This motivates our first design principle: regulate the chosen and rejected learning signals separately. From a mutual information variational perspective, we derive a Pearson-χ^2-based objective that separates the chosen and rejected gradient coefficients from the relative margin, providing distinct signals to reinforce chosen evaluations and suppress rejected ones.

Q2: How can generative PRMs avoid overreliance on local analysis?

Overreliance on local analysis highlights the need to integrate relevant evidence rather than let a partial analysis dominate the final judgment. A correct final judgment does not ensure a sound analysis throughout, motivating control over how strongly individual trajectories are reinforced. Separating the learning signals alone leaves their scales insufficiently controlled: in the Pearson objective, the chosen coefficient is constant and does not respond to changes in its own log-ratio, permitting excessive growth that we term **Log-ratio Inflation**. This motivates our second design principle: complement coefficient-level separation with individual log-ratio scale control. CM-PRM uses separate quadratic terms to make each coefficient respond to its own log-ratio and establish finite coordinate-wise targets. Reinforcement or suppression weakens as each log-ratio approaches its target and reverses direction after crossing it, providing explicit feedback to correct scale deviations.

Guided by these principles, CM-PRM integrates separate learning signals and log-ratio scale control for generative PRM training. Theoretically, it preserves the early-stage direction of standard preference optimization, provides sample-wise feedback toward finite log-ratio targets, and admits an empirical estimator variance bound under stated assumptions. Empirically, process-evaluation and reward-guided search experiments demonstrate its effectiveness, while ablations and training-dynamics analyses support the joint design.

2 Rethinking Generative PRMs through Information-Theory

2.1 Problem Formulation of Generative PRMs

Given a problem q and a reasoning sequence $s_{1:K}$, we define the shared context at step t as $z_t = (q, s_{<t})$ and the process state as $x_t = (z_t, s_t)$, where s_t is the current step to be verified. Let Z

and X denote the corresponding random variables. Unlike discriminative PRMs that output scalar rewards, a generative PRM π_θ generates a complete evaluation trajectory $Y = (A, \widehat{G}) \sim \pi_\theta(\cdot \mid X)$, comprising a natural-language analysis A and a final judgment $\widehat{G}$. Let G denote the ground-truth correctness label of the current step and define $C = \mathbf{1}\{\widehat{G} = G\}$. Thus, C indicates whether the evaluation reaches the correct final judgment, rather than whether the current step is correct or every statement in the analysis has been verified. Our goal is to learn a generation policy $\pi_\theta(Y \mid X)$ that produces informative analyses and accurate step-level judgments.

2.2 Reformulating Generative PRMs via Conditional Mutual Information

Reliable process evaluation requires final judgments to be grounded in the current reasoning step while accounting for the problem context and preceding reasoning. We use conditional f-mutual information $I_f(X; Y \mid Z)$ to characterize the statistical dependence between process states and evaluation trajectories. Since exact computation is generally intractable in high-dimensional text spaces, we consider a variational lower bound under a fixed offline data distribution (Nguyen et al., 2010; Nowozin et al., 2016):

$$I_f(X;Y \mid Z) := D_f(P_{X,Y|Z} || P_{X|Z}P_{Y|Z}) \geq \sup_\theta J_f(T_\theta) := \mathbb{E}_{P_{X,Y|Z}} T_\theta(X,Y) - \mathbb{E}_{P_{X|Z}P_{Y|Z}} f^*(T_\theta(X,Y)). \quad (1)$$

where $P_{X,Y|Z}$ is the joint distribution of states and matched analyses, $P_{X|Z}P_{Y|Z}$ is the product of conditional marginals corresponding to independent pairing, and f^* is the convex conjugate of f.

To encourage accurate, evidence-grounded analyses, we draw on contrastive learning (Oord et al., 2018; Hjelm et al., 2019) and maximize this lower bound to learn a critic T_θ that captures state–analysis associations. Following prior work (Xiao et al., 2025; Lv et al., 2025), we set $T_\theta(x, y) = \tau \log \frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}$, with $\tau > 0$ (see proof in Appendix A.1). This parameterization links the variational objective to each evaluation trajectory's log-ratio relative to the reference model, allowing us to examine both separation between trajectories and the learning signal assigned to each trajectory. By reinforcing state-matched evaluations while contrasting them with mismatched ones, the objective encourages the PRM to generate evaluations grounded in evidence from the current step.

2.3 The Margin Trap in Preference-Optimized Generative PRMs

Preference optimization learns from paired evaluations with correct and incorrect final judgments. For a given state, the chosen trajectory y_w has $c_w = 1$, while the rejected trajectory y_l has $c_l = 0$ (She et al., 2025; Whitehouse et al., 2025). Through two-sample InfoNCE, we connect DPO to the variational perspective and examine its gradient coefficients. The algebraic connection is direct; however, interpreting the objective as a CMI lower bound when rejected trajectories are drawn from a state-specific distribution requires an additional sampling condition.

Theorem 1 (DPO as CMI Lower-Bound Objective). For $f(t) = t \log t$, two-sample InfoNCE provides the following conditional f-mutual information lower bound (Oord et al., 2018):

$$I(X;Y \mid Z) \geq J_{\text{InfoNCE}} := \sup_\theta \mathbb{E}_{\substack{(x,y_w)\sim P_{X,Y|z} \\ y_l^{\text{iid}} \sim P_{Y|z}}} \left[\log \frac{2e^{T_\theta(x,y_w)}}{e^{T_\theta(x,y_w)} + e^{T_\theta(x,y_l^{\text{iid}})}} \right]. \quad (2)$$

Here, y_w is sampled jointly with the current state, whereas y_l^{iid} is drawn independently from the conditional marginal and randomly paired with the current state. Proof is in Appendix A.2. In preference data, rejected trajectories $y_l \sim q(\cdot \mid x, z)$ are state-specific hard negative with incorrect final judgments, rather than randomly paired negatives.. Define

$$h_w = \log \frac{\pi_\theta(y_w \mid x)}{\pi_{\text{ref}}(y_w \mid x)}, \qquad h_l = \log \frac{\pi_\theta(y_l \mid x)}{\pi_{\text{ref}}(y_l \mid x)}. \quad (3)$$

Replacing y_l^{iid} with y_l and substituting $T_\theta(x, y) = \tau \log \frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}$ yields

$$J^{\text{HG}}_{\text{InfoNCE}}(T_\theta) = \log 2 + \mathbb{E}_{\substack{(x,y_w)\sim P_{X,Y|z} \\ y_l \sim q(\cdot|x,z)}} \log \sigma(\tau(h_w - h_l)) = \log 2 + J_{\text{DPO}}(\theta). \quad (4)$$

Motivated by Bradley–Terry preference utilities in DPO (Rafailov et al., 2023), we formalize $T_\theta(x, y_l) \succeq_{\text{st}} T_\theta(x, y_l^{\text{iid}})$ conditional on x, z, where $y_l \sim q(\cdot \mid x, z)$ and $y_l^{\text{iid}} \sim P_{Y|z}$ are sampled independently of the positive evaluation. Since the InfoNCE objective decreases with the negative score, we obtain (proof in Appendix A.3):

$$\boxed{I(X;Y \mid Z) \geq J_{\text{InfoNCE}} \geq J^{\text{HG}}_{\text{InfoNCE}}(T_\theta) = \log 2 + J_{\text{DPO}}(\theta).} \quad (5)$$

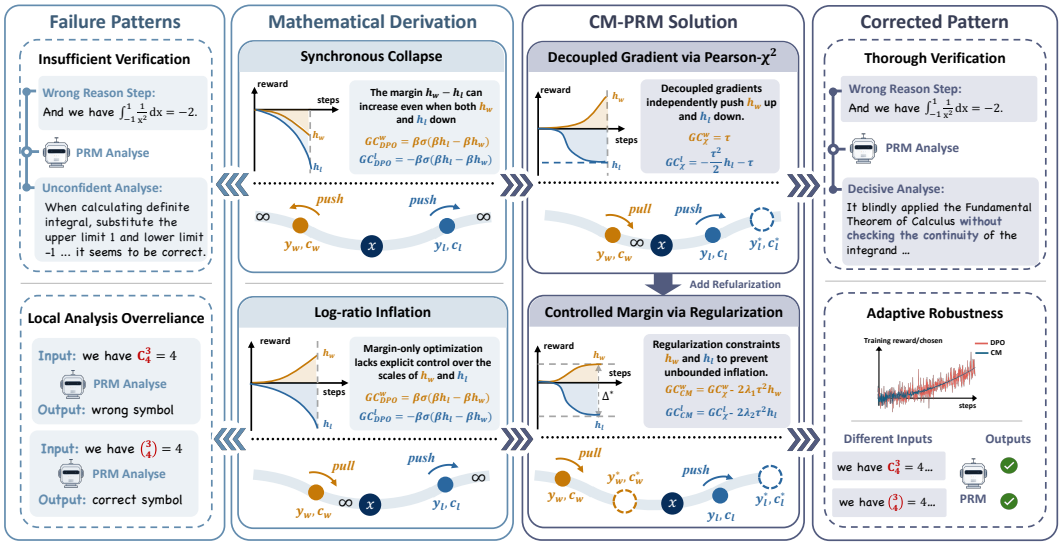


Figure 4: Overview of CM-PRM. PO-based generative PRM training suffers from two failure modes: Synchronous Collapse and Log-ratio Inflation. CM-PRM addresses them with decoupled gradient updates and controlled-margin regularization, yielding more decisive and robust process evaluations.

Under the above condition, minimizing the DPO loss is equivalent to maximizing a CMI lower bound; however, its gradient coefficients depend only on the preference pair's relative margin:

$$\nabla_\theta j_{\mathrm{DPO}} = \mathrm{GC}_w \nabla_\theta \log \pi_\theta(y_w \mid x) + \mathrm{GC}_l \nabla_\theta \log \pi_\theta(y_l \mid x), \quad \mathrm{GC}_w = -\mathrm{GC}_l = \tau\sigma(\tau(h_l - h_w)). \tag{6}$$

The gradient coefficients depend solely on $h_w - h_l$ and do not regulate the two log-ratios separately. We term this limitation the **Margin Trap**, which permits two potential failure modes (Figure 3).

First, DPO can improve its objective if h_l falls faster even when both h_w and h_l decline, resulting in **Synchronous Collapse**: with a frozen reference policy, the margin grows while chosen trajectories become less likely. Since rejected samples cannot cover all incorrect evaluations, suppressing them alone cannot ensure reinforcement of chosen trajectories, motivating margin-independent gradient coefficients. Second, DPO lacks separate log-ratio scale constraints, allowing h_w and h_l to grow excessively, resulting in **Log-ratio Inflation**. This can over-reinforce evaluations, causing final judgments to rely too heavily on local analysis while overlooking other evidence.

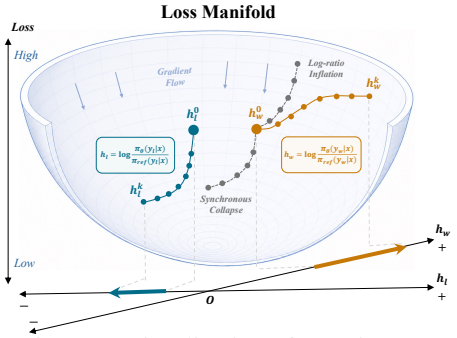


Figure 3: Visualization of Margin Trap

These limitations motivate us to revisit the conditional-information variational framework for an alternative training objective. The objective should decouple the chosen and rejected gradient coefficients from the relative margin while separately controlling their log-ratio scales.

3 Controlled-Margin Contrastive Learning for PRM

We introduce CM-PRM, a controlled-margin contrastive learning framework for generative PRMs (Figure 4). Following the preceding analysis, we derive a Pearson-χ^2 objective that separates chosen and rejected gradient coefficients from the relative margin. We then add quadratic penalties that make each coefficient respond to its own log-ratio, providing individual scale control toward finite coordinate-wise targets.

3.1 Breaking the Margin Trap with Pearson χ^2

Generative PRMs learn reliable step-level judgments from correct and incorrect analyses. However, as discussed in sec 2.3, the probability of correct trajectories may decrease even as the preference

margin grows. We therefore use the f-divergence variational framework in Eq. 1 to regulate their update signals separately, so that reinforcement of correct evaluations no longer depends solely on the margin. Retaining $T_\theta(x, y_l) \succeq_{\mathrm{st}} T_\theta(x, y_l^{\mathrm{iid}})$ from sec 2.3, we have:

$$I_f(X;Y \mid Z) = D_f(P_{X,Y|Z} || P_{X|Z} P_{Y|Z}) \geq J_f^{\mathrm{HG}} := \sup_\theta \mathbb{E}_{(x,y_w)\sim P_{X,Y|Z}} T_\theta(y_w \mid x) - \mathbb{E}_{\substack{x\sim P_{X|Z} \\ y_l \sim q(\cdot|x,Z)}} f^*(T_\theta(y_l \mid x)), \quad (7)$$

where f is a convex generator and f^* is its convex conjugate. The positive and negative samples y_w and y_l follow the same definitions and sampling distributions as in Section 2.3. For notational simplicity, we subsequently denote J^{HG} by J.

Within this family, we choose Pearson-χ^2 divergence because it decouples the chosen and rejected gradient coefficients from the relative margin while preserving an early suppression signal for rejected evaluations. Its quadratic form (Table 1) yields linear gradient coefficients and facilitates the individual log-ratio control introduced below. See Appendix B for detailed comparisons.

Table 1: Conjugates and NWJ bounds for f-divergences.

Divergence	$\mathbf{f^*(t)}$	**NWJ bound of** $D_f(P \parallel Q)$
KL	e^{t-1}	$\mathbb{E}_P[T] - \mathbb{E}_Q\left[e^{T-1}\right]$
Reverse KL	$-1 - \log(-t)$	$\mathbb{E}_P[T] - \mathbb{E}_Q[-1 - \log(-T)]$
JS	$-\log(2 - e^t)$	$\mathbb{E}_P[T] - \mathbb{E}_Q\left[-\log(2 - e^T)\right]$
SH	$\frac{t}{1-t}$	$\mathbb{E}_P[T] - \mathbb{E}_Q\left[\frac{T}{1-T}\right]$
Pearson-χ^2	$t + \frac{t^2}{4}$	$\mathbb{E}_P[T] - \mathbb{E}_Q\left[T + \frac{T^2}{4}\right]$

For Pearson-χ^2, define $\bar{f}^*(t) = t + t^2/4$ for all $t \in \mathbb{R}$. This extension upper-bounds the exact conjugate, so using it preserves Eq. 7 under the stated sampling condition. Substituting $T_\theta(x, y) = \tau h_\theta(x, y)$ gives

$$J_{\chi^2}(\theta) = \mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}} \underbrace{\left[\tau h_w - \tau h_l - \frac{\tau^2}{4} h_l^2\right]}_{j_{\chi^2}(\theta)}, \quad (8)$$

where $\mathcal{D}$ draws $z \sim P_Z$, $(x, y_w) \sim P_{X,Y|Z}$, and $y_l \sim q(\cdot \mid x, z)$. The log-ratios follow Eq. 3. For a single pair, let $g_w = \nabla_\theta \log \pi_\theta(y_w \mid x)$ and $g_l = \nabla_\theta \log \pi_\theta(y_l \mid x)$. Then

$$\nabla_\theta j_{\chi^2} = \mathrm{GC}_w g_w + \mathrm{GC}_l g_l, \qquad \mathrm{GC}_w = \tau, \quad \mathrm{GC}_l = -\tau - \frac{\tau^2}{2} h_l. \quad (9)$$

The two update signals are no longer coupled through the relative margin $h_w - h_l$. However, the chosen coefficient $\mathrm{GC}_w = \tau$ neither strengthens when h_w declines nor weakens as it grows, whereas the rejected coefficient already responds to h_l. Pearson therefore provides asymmetric individual feedback.

3.2 Controlled-Margin Contrastive Objective

Pearson-χ^2 contrastive learning separates the chosen and rejected gradient coefficients, but leaves the chosen log-ratio without explicit scale control, permitting Log-ratio Inflation. Excessive reinforcement may over-amplify particular analytical patterns in preference data, since a correct final judgment does not ensure a sound analysis throughout.

To control individual log-ratio scales, we introduce separate quadratic penalties and define the per-pair controlled-margin objective:

$$j_{\mathrm{CM}} = \tau h_w - \tau h_l - \frac{\tau^2}{4} h_l^2 \underbrace{-\lambda_1 \tau^2 h_w^2 - \lambda_2 \tau^2 h_l^2}_{\text{Penalty terms}}, \quad (10)$$

with $\lambda_1 > 0$ and $\lambda_2 \geq 0$. By negating this objective, we obtain the training loss and rewrite it using unified hyperparameters:

$$\boxed{\mathcal{L}_{\mathrm{CM}} = -\tau \log \frac{\pi_\theta(y_w|x)}{\pi_{\mathrm{ref}}(y_w|x)} + \tau \log \frac{\pi_\theta(y_l|x)}{\pi_{\mathrm{ref}}(y_l|x)} + \frac{\beta\tau^2}{2}\left[\log \frac{\pi_\theta(y_w|x)}{\pi_{\mathrm{ref}}(y_w|x)}\right]^2 + \frac{\gamma\tau^2}{2}\left[\log \frac{\pi_\theta(y_l|x)}{\pi_{\mathrm{ref}}(y_l|x)}\right]^2,} \quad (11)$$

where $\beta = 2\lambda_1$ and $\gamma = \frac{1}{2} + 2\lambda_2$. For the objective j_{CM}, the corresponding gradient coefficients are

$$\mathrm{GC}_w = \tau - 2\lambda_1 \tau^2 h_w, \quad \mathrm{GC}_l = -\tau - 2\left(\frac{1}{4}\tau^2 + \lambda_2 \tau^2\right) h_l. \quad (12)$$

As h_w increases, $-2\lambda_1\tau^2 h_w$ reduces the positive update coefficient, thereby weakening reinforcement of the chosen trajectory. Both coefficients thus respond to their own log-ratios, providing separate feedback on reinforcement and suppression.

Viewed as a function of the independent log-ratio coordinates (h_w, h_l), the loss is strictly convex for $\beta, \gamma, \tau > 0$, with a positive-definite Hessian. It therefore admits a unique stationary minimizer in the log-ratio space (proof in Appendix C.1). The optimal log-ratios and margin are

$$h_w^* = \frac{1}{\beta\tau}, \quad h_l^* = -\frac{1}{\gamma\tau}, \quad \Delta^* = h_w^* - h_l^* = \frac{1}{\beta\tau} + \frac{1}{\gamma\tau}. \tag{13}$$

GC_w and GC_l vanish at the coordinate-wise optima, yielding a finite target margin Δ^*. By separately regulating chosen and rejected log-ratios, CM-PRM combines gradient-coefficient decoupling with scale control, aiming to curb excessive inflation and improve discrimination of locally plausible reasoning steps that contain critical errors.

3.3 Discussion: Theoretical Properties of the Controlled-Margin Objective

In this section, we analyze three theoretical properties of CM-PRM: directional effectiveness, first-order consistency with standard preference optimization, and bounded variance of the empirical estimator.

- **Directional Effectiveness of Optimization.** The CM objective $\mathcal{J}_{\mathrm{CM}}$ preserves the desired push-pull direction in the log-ratio space:

Proposition 1 ***(Directional Effectiveness).*** *When $h_w < h_w^*$ and $h_l > h_l^*$, gradient-ascent updates in the independent log-ratio coordinates increase h_w and decrease h_l.*

In early fine-tuning, $h_w, h_l \approx 0$, yielding $\mathrm{GC}_w \approx \tau > 0,\ \mathrm{GC}_l \approx -\tau < 0$. The regularization terms have not yet canceled the base contrastive signal. These gradient coefficients provide reinforcement and suppression signals for chosen and rejected outputs, respectively.

- **Early-stage Alignment and Sample-wise Corrections.** At the early training stage, CM shares the leading optimization direction of DPO (Rafailov et al., 2023) and IPO (Azar et al., 2024) after matching their gradient scales. As the log-ratios depart from zero, CM applies separate corrections based on each trajectory's own log-ratio.

Proposition 2 ***(Early-stage Equivalence and Late-stage Adaptivity).*** *For a preference pair, write $\Delta = h_w - h_l$. With $\mathcal{L}_{\mathrm{DPO}} = -\log\sigma(2\tau\Delta)$ and $\mathcal{L}_{\mathrm{IPO}} = (\Delta - \tau/2)^2$ to match the leading gradient scales , the gradients decompose as*

$$\nabla_\theta \mathcal{L}_k = g_0 + R_k, \qquad g_0 = -\tau(g_w - g_l), \quad k \in \{\mathrm{DPO}, \mathrm{IPO}, \mathrm{CM}\}, \tag{14}$$

with corrections (see details in Appendix C.2)

$$R_{\mathrm{DPO}} = \tau\tanh(\tau\Delta)(g_w - g_l), \quad R_{\mathrm{IPO}} = 2\Delta(g_w - g_l), \quad R_{\mathrm{CM}} = \tau^2(\beta h_w g_w + \gamma h_l g_l). \tag{15}$$

For bounded g_w, g_l, each correction is $O(\rho)$ as $\rho := \max\{|h_w|, |h_l|\} \to 0$.

The methods thus align near the reference policy but differ in how their corrections depend on the log-ratios: DPO and IPO use the relative margin, while CM responds to each trajectory separately.

- **Variance Boundedness of Empirical Estimators.** Under bounded log-ratio scales and the assumptions in Appendix C.3, the empirical objective evaluated at the estimated parameters satisfies the asymptotic variance bound:

$$\mathrm{Var}\left[\hat{J}_{\mathrm{CM}}^{n,m}\right] = \mathcal{O}\left(\frac{1}{n} + \frac{1}{m}\right). \tag{16}$$

Under these assumptions, bounded log-ratio scales limit the impact of extreme samples and help stabilize empirical objective estimation.

4 Experiments

In this section, we evaluate CM-PRM on process evaluation and PRM-guided search. Through controlled comparisons, training-dynamics analysis, and fine-grained evaluation, we examine how individual trajectory control affects training behavior and judgment quality. We address 3 questions:

RQ1: Can CM-PRM outperform other PRMs in process evaluation and PRM-guided search?

RQ2: How do gradient decoupling and controlled-margin regularization in CM-PRM jointly improve the training dynamics and discriminative capability of generative PRMs?

RQ3: Can CM-PRM's explicit analyses effectively support correctness judgments and enable competitive process evaluation under limited supervision?

Table 2: PRMBench performance. ★ marks our **CM-PRM**; † and ‡ denote results from (She et al., 2025) and (Pala et al., 2025), respectively. Bold/underline indicate the best/second-best PRM results. * and ** indicate significant differences over Vanilla DPO baseline with $p < 0.05$ and $p < 0.01$.

MODEL	# Samples	Simplicity			Soundness					Sensitivity				Overall
		NR.	NCL.	**Avg.**	ES	SC.	DC.	CI	**Avg.**	PS	DR.	MS.	**Avg.**	
LLM-as-judge, Proprietary language models														
GPT-4o†	-	57.0	62.4	59.7	72.0	69.7	70.7	71.1	70.9	62.5	65.7	99.2	75.8	66.8
o1-mini†	-	65.6	63.7	64.6	74.5	67.7	73.8	72.3	72.1	61.8	64.8	100.0	75.5	68.8
LLM-as-judge, Open-source language models														
Qwen-2.5-Math-72B‡	-	55.3	54.9	55.1	55.5	71.6	58.1	59.1	61.1	47.4	53.8	100.0	67.1	57.4
QwQ-Preview-32B‡	-	57.2	55.6	56.4	67.4	72.3	66.2	66.9	68.2	57.8	62.7	100.0	73.5	63.6
Discriminative PRMs														
Math-Shepherd-7B†	445K	44.0	50.3	47.1	49.4	44.5	41.3	47.7	45.7	47.2	48.6	86.1	60.7	47.0
Math-PSA-7B†	860K	47.6	55.1	51.3	56.5	49.4	47.1	54.2	51.8	51.7	54.1	88.9	64.9	52.3
RLHFflow-Mistral-8B†	273K	46.1	47.3	46.7	56.6	55.1	54.4	63.8	57.5	51.5	56.2	97.9	68.5	54.4
RLHFflow-DeepSeek-8B†	250K	46.4	48.9	47.6	55.7	55.0	53.2	66.2	57.5	49.0	55.4	99.8	68.1	54.2
Llemma-PRM800k-7B†	350K	49.3	53.4	51.4	56.4	47.1	46.7	53.3	50.9	51.0	53.5	93.6	66.0	52.0
Skywork-PRM-7B†	unk	35.7	41.2	38.4	36.7	29.1	30.6	34.4	32.7	36.8	37.4	88.8	54.3	36.2
ReasonEval-7B†	350K	61.0	50.1	55.5	62.1	65.9	61.5	66.0	63.9	55.6	58.0	99.5	71.0	60.0
Qwen2.5-Math-7B-PRM800K†	264K	48.6	47.8	48.2	62.1	59.4	58.7	68.5	62.2	52.9	64.0	**99.8**	72.2	58.3
Qwen2.5-Math-PRM-7B†	∼1.8M	49.0	55.1	52.1	71.8	67.3	66.3	**78.5**	71.0	57.6	69.1	99.7	75.5	65.5
PathFinder-PRM-7B‡	400K	52.1	**65.8**	**58.9**	73.1	68.7	66.3	75.0	70.8	61.7	69.8	99.2	76.9	67.7
Generative PRMs														
GenPRM-7B (Qwen)	22.3K	48.5	53.7	51.1	59.5	53.4	54.3	57.9	56.3	57.7	57.0	95.6	70.1	56.3
R-PRM-7B-DPO†	558K	52.2	58.2	55.2	72.1	69.1	68.9	75.0	71.2	61.2	69.5	99.1	76.6	66.8
Vanilla Ablation Baselines														
Backbone w/ SFT	289K	50.2	62.1	56.2	67.3	66.0	61.6	68.0	65.7	59.5	65.5	98.2	74.4	63.5
Backbone w/ SFT+Verdict	417K	31.3	38.9	35.1	36.0	29.1	30.0	30.2	31.3	36.4	35.5	56.1	42.7	34.0
Backbone w/ SFT+DPO	417K	50.8	63.4	57.1	71.5	64.5	66.8	69.4	68.1	59.1	67.6	98.8	75.2	65.1
Backbone w/ SFT+DPO+Reg.	417K	49.4	55.4	52.4	67.5	64.6	62.2	63.9	64.6	55.8	61.6	98.0	71.8	61.1
Backbone w/ SFT+IPO	417K	49.1	53.7	51.4	71.9	70.3	64.9	74.0	70.3	60.1	69.4	95.9	75.1	65.0
Backbone w/ SFT+IPO+Reg.	417K	48.5	48.8	48.7	68.4	69.4	63.2	64.5	66.4	57.6	65.0	98.5	73.7	61.7
Backbone w/ SFT+Pearson	417K	49.8	46.4	48.1	61.9	64.3	59.0	67.6	63.2	52.8	63.9	98.1	71.6	59.3
Backbone w/ SFT+Pearson+Reg^{+}.	417K	47.6	55.5	51.6	63.9	63.6	63.9	65.2	64.2	56.1	62.3	98.1	72.2	60.8
Backbone w/ SFT+Pearson+Reg^{-}.	417K	47.2	53.8	50.5	66.2	62.5	61.8	65.5	64.0	56.6	63.2	98.5	72.8	60.6
Our Model														
★ **CM-PRM**	417K	**53.7****	64.1 *	**58.9****	**73.6****	**71.1***	**70.3***	77.9*	**73.2****	**62.7****	**70.9****	99.5 *	**77.7****	**69.1****

4.1 Experimental Settings

Training. We use Qwen2.5-Math-7B-Instruct (Yang et al., 2024) as the backbone and follow the data setting of R-PRM (She et al., 2025), comprising 289K SFT samples and 269K preference samples. We first perform SFT on the 289K samples to learn the process-evaluation format, then conduct controlled-margin contrastive learning on a subset totaling 128K chosen and rejected samples. Unless otherwise specified, we set $\beta = 30$, $\gamma = 60$, and $\tau = 1.0$. Details are provided in Appendix F.1.

Evaluation. Following the evaluation settings of R-PRM (She et al., 2025), we evaluate CM-PRM on static process evaluation and PRM-guided search. For static evaluation, we report F1 scores on PRMBench (Song et al., 2025) and ProcessBench (Zheng et al., 2025) in Table 2 and Appendix G.1, respectively. For search, Qwen2.5-7B-Instruct generates candidate continuations at each step, and the PRM selects the highest-reward candidate until a final answer is reached. Search performance is measured by final-answer accuracy. Dataset details are provided in Appendix F.3.

Baselines. We compare CM-PRM with LLM-as-a-Judge, discriminative PRMs, and generative PRMs; model descriptions appear in the experimental tables and Appendix F.2. We conduct three groups of ablations: to assess the contribution of analysis to step-correctness judgments, we introduce Verdict-only, trained with the same CM objective, which outputs judgments without analysis; to examine how gradient update mechanisms affect PRM discrimination and error localization, we compare DPO, IPO, and CM-PRM against SFT; to investigate the sources of robust evaluation, we compare Pearson-χ^2-CL and its two one-sided regularization variants with the full CM-PRM, alongside DPO, IPO, and their regularized variants. All training-objective variants differ only in their objectives and objective-specific hyperparameters, with other training settings matching CM-PRM.

4.2 Performance on Process Evaluation and Guided Search

To answer RQ1, we evaluate whether CM-PRM provides stronger process-level rewards in static evaluation (PRMBench) and test-time search (PRM-guided greedy search with a policy model).

On PRMBench, CM-PRM achieves an average F1 of 69.1, outperforming other PRM baselines overall (Table 2). Its *Soundness* and *Sensitivity* scores reach 73.2 and 77.7, respectively, suggesting stronger discrimination between valid and flawed reasoning steps, including locally plausible errors.

Table 3: PRM-guided greedy search performance. ★ marks our **CM-PRM**; † and ‡ denote results from (She et al., 2025) and (Pala et al., 2025). Bold/underline indicates thebest/second-best PRM results. * and ** indicate significant differences over Vanilla DPO baseline with $p < 0.05$, $p < 0.01$.

Setting	AIME24	AMC23	MATH	Olympiad	College	Minerva	Avg.
pass@1†	11.2	47.8	73.0	38.0	38.6	37.2	41.0
major@8†	20.0	57.5	79.6	47.0	41.5	42.7	48.0
pass@8†	33.3	82.5	88.8	58.5	47.5	57.7	61.4
Discriminative PRM (prm@8)							
Math-Shepherd-7B†	13.3	52.5	74.6	38.5	36.5	41.2	42.8
Math-PSA-7B†	6.7	57.5	79.8	42.5	41.0	39.3	44.5
RLHFflow-PRM-Mistral-8B†	10.0	57.5	73.4	37.5	38.0	41.2	42.9
RLHFflow-PRM-DeepSeek-8B†	13.3	52.5	74.8	39.5	37.0	40.8	43.0
Lemma-PRM800k-7B†	13.3	57.5	73.8	40.0	36.5	38.2	43.2
Skywork-PRM-7B†	10.0	57.5	77.8	41.5	39.0	43.4	44.9
ReasonEval-7B†	3.3	55.0	73.0	37.5	35.5	37.9	40.4
Qwen2.5-Math-7B-PRM800K†	**23.3**	45.0	78.2	42.0	35.5	38.6	43.8
Qwen2.5-Math-PRM-7B†	16.7	60.0	81.0	43.5	39.0	40.4	46.8
PathFinder-PRM-7B‡	<u>20.0</u>	62.5	78.8	36.5	**55.0**	36.7	48.3
Generative PRM (prm@8)							
R-PRM-7B-DPO†	16.7	**70.0**	80.0	<u>46.5</u>	39.5	<u>43.4</u>	<u>49.4</u>
GenPRM-7B (Qwen)	10.0	55.0	76.1	43.5	45.5	34.6	44.1
Vanilla Ablation Baselines (prm@8)							
Backbone w/ SFT	13.3	57.5	76.6	43.5	39.0	42.0	45.3
Backbone w/ SFT+Verdict	11.2	42.5	71.4	40.5	44.0	27.4	39.5
Backbone w/ SFT+DPO	13.3	62.5	<u>81.6</u>	46.5	40.5	43.0	49.0
Backbone w/ SFT+DPO+Reg	13.3	50.0	74.8	49.0	48.5	42.6	46.4
Backbone w/ SFT+IPO	13.3	60.0	77.8	48.0	47.0	39.8	47.7
Backbone w/ SFT+IPO+Reg	10.0	47.5	72.6	49.0	36.5	40.5	42.7
Backbone w/ SFT+Pearson	13.3	54.5	76.8	46.0	50.5	40.2	46.9
Backbone w/ SFT+Pearson+Reg^{+}	13.3	55.0	77.2	45.4	51.2	41.6	47.3
Backbone w/ SFT+Pearson+Reg^{-}	13.3	60.0	78.4	46.6	49.8	37.0	47.5
Our Model (prm@8)							
★ **CM-PRM**	<u>20.0</u>*	<u>65.0</u>*	**83.6***	**52.0****	<u>53.0</u>*	**47.7***	**53.6***

In PRM-guided greedy search, CM-PRM achieves an average accuracy of 53.6 (Table 3), outperforming all PRM-guided methods. Its process-evaluation capability transfers to search guidance, enabling selection among reasoning continuations rather than merely judging fixed steps.

4.3 Gradient Decoupling and Controlled-Margin Optimization

To answer RQ2, we examine controlled-margin contrastive learning through ablation studies and training-dynamics analysis. Comparisons include DPO, IPO, Pearson-χ^2-CL and its two one-sided regularization variants, and the quadratically regularized variants of DPO and IPO. All variants share the same SFT initialization, data, common configuration, and budget, differing only in their objectives and objective-specific hyperparameters.

Limitations of standard preference optimization. DPO and IPO underperform CM-PRM on PRMBench and guided search (Tables 2, 3). Their reward margins increase while chosen and rejected rewards decline together, with rejected rewards falling faster (Figure 6). Thus, a growing margin does not necessarily indicate reinforcement of the sampled chosen evaluations, highlighting the limitation of relying on relative preference separation alone.

Complementarity of gradient decoupling and regularization. Pearson-χ^2-CL decouples the chosen and rejected gradient coefficients, yet its chosen reward still declines in later training. With shared model parameters, coefficient decoupling does not guarantee independent trajectory updates, while its constant $GC_w = \tau$ cannot respond to the chosen log-ratio. The two one-sided regularization variants modestly outperform Pearson but remain below CM-PRM. Adding quadratic regularization to DPO and IPO also fails to improve performance under our settings. Together, these results support CM-PRM's joint design of gradient decoupling and separate log-ratio control.

Effect of regularization strength. We track rewards, margins, and accuracy across (β, γ) settings (Figure 5). With the default $(\beta, \gamma) = (30, 60)$, CM-PRM raises the chosen reward while lowering the rejected reward, avoiding the synchronous decline observed with standard preference objectives. Moreover, larger coefficients yield smoother but slower reward dynamics, consistent with stronger scale constraints and target log-ratios closer to zero. Together with the task results, these curves support controlled-margin regularization as a means of regulating individual trajectories beyond their relative separation, while improving process evaluation. Appendix G.2 examines the effect of τ.

4.4 The Role of Analysis in Evaluation Trajectories

To answer RQ3, we examine the role of analysis within evaluation trajectories from two perspectives. First, analysis quality matters for judgment accuracy. CM-PRM outperforms Verdict-only on PRMBench and in guided search, but generating analyses alone is insufficient: on ProcessBench,

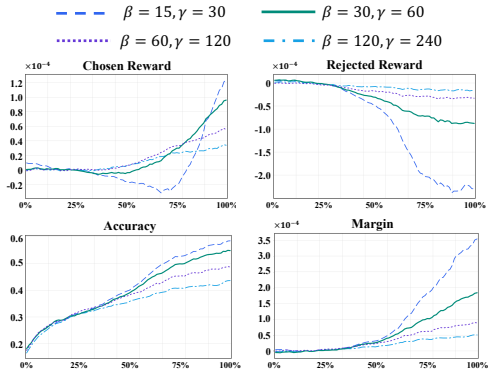


Figure 5: Training dynamics of CM-PRM under different controlled-margin hyperparameters.

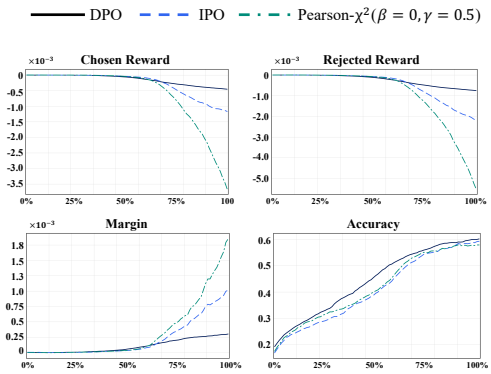


Figure 6: Training dynamics of DPO, IPO and Pearson-χ^2 varients.

SFT, DPO, IPO, and DPO+Reg. outperform Verdict-only, whereas Pearson and IPO+Reg. underperform it (Appendix G.1). These results suggest that effectively trained analyses support correctness judgments, while unsuitable training can diminish their benefits and perform worse than judging without analyses. Second, analysis-based evaluators show promise under limited supervision. Using 289K SFT samples and 128K preference samples, CM-PRM achieves 69.1 on PRMBench, exceeding Qwen2.5-Math-PRM-7B, which uses about 1.8M step-level samples, by 3.6 F1 points (Figure 12). Although supervision formats differ, this suggests that learning to use correctness-relevant evidence may support stronger judgments with fewer samples; see Appendix G.3.

5 Related Work

Process Reward Modeling. PRMs provide step-level supervision for reasoning steps and are more informative than ORMs, which only score final answers (Lightman et al., 2024; Uesato et al., 2022). Step rewards enable error detection and support inference procedures such as best-of-N selection and guided search (Wang et al., 2024b; Zhang et al., 2025b). Since reliable evaluation requires distinguishing subtle correctness differences among locally plausible steps, our work focuses on improving PRM discrimination under limited process supervision.

Discriminative Process Reward Models. Discriminative PRMs cast process evaluation as step-level classification, outputting a scalar correctness score for each reasoning step. Prior work improves reward calibration (Zhang et al., 2025a), credit assignment (Cheng et al., 2025), verification quality (Tan et al., 2026), and uncertainty modeling (Cao et al., 2025). However, scalar rewards compress heterogeneous reasoning errors into one score, making it difficult to capture locally plausible but globally flawed continuations and potentially encouraging superficial heuristics (Pala et al., 2025). Strong discriminative PRMs often require large-scale step-level annotations or costly synthetic labeling pipelines (Wang et al., 2024b; Zhang et al., 2025b; 2026b; Xi et al., 2026).

Generative Process Reward Models. Generative PRMs evaluate reasoning steps through natural-language analyses and explicit judgments, improving interpretability and enabling inference-time scaling through sampled evaluation trajectories (Zhao et al., 2026). LLM-as-a-judge is a common form (Gu et al., 2026), but its effectiveness depends heavily on model scale, prompt design, and inference budget, while small open-source judges often have unstable accuracy and output formats (Zheng et al., 2025; Bavaresco et al., 2025). Recent generative PRMs, such as GenPRM (Zhao et al., 2026), R-PRM (She et al., 2025), and ThinkPRM (Khalifa et al., 2026), learn process evaluation through SFT or RFT-style training. However, they may produce unconfident judgments or be sensitive to minor context perturbations, motivating objectives that preserve discriminative confidence and robustness.

6 Conclusion and Future Work

We presented CM-PRM, a controlled-margin contrastive learning framework for generative process reward models. By mitigating reward-margin-induced gradient coupling and log-ratio inflation, CM-PRM improves the stability and discriminative capability of process evaluation. Experiments on static process evaluation and PRM-guided test-time scaling show consistent gains over existing generative PRMs, while ablation studies and training-dynamics analyses further support the role of the controlled-margin mechanism. This work focuses on mathematical reasoning, where step boundaries and correctness criteria are relatively clear. Future work will explore more open-ended process supervision, including code generation, scientific reasoning, and long-document rule verification, using global context, external tool feedback, or domain-specific knowledge for reliable step evaluation.

AI USE STATEMENT

(This section is **required** and does not count toward the page limit.)

We used generative AI tools to assist with language editing and polishing, including improvements to grammar, clarity, and readability. The authors reviewed and revised the AI-assisted text to ensure that it accurately reflected the methods, results, and intended claims of the study. We take full responsibility for the final content of this paper, including all text, technical claims, and research artifacts.

ETHICS STATEMENT

(This section is **recommended** and does not count toward the page limit.)

This work aims to improve process evaluation in mathematical reasoning. We have not identified significant ethical risks or potential harmful applications directly associated with this study. The model may nevertheless produce incorrect judgments, and its outputs should not be treated as guarantees of mathematical correctness.

REPRODUCIBILITY STATEMENT

(This section is **recommended** and does not count toward the page limit.)

We provide the methodological and experimental details needed to support reproduction of our results. The main text specifies the CM-PRM objective and its theoretical properties, with supporting derivations, assumptions, and proofs in Appendices A–C. Appendix E provides the inference prompts, while Appendix F describes the training configuration, baseline and ablation implementations, hyper-parameter selection, and evaluation setup. Additional experimental results are reported in Appendix G. We also submit the implementation code as supplementary material to facilitate verification and reproduction of the reported experiments.

REFERENCES

Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Remi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. A general theoretical paradigm to understand learning from human preferences. In Proc. of AISTATS, pp. 4447–4455, 2024.

Anna Bavaresco, Raffaella Bernardi, Leonardo Bertolazzi, Desmond Elliott, Raquel Fernández, Albert Gatt, Esam Ghaleb, Mario Giulianelli, Michael Hanna, Alexander Koller, et al. Llms instead of human judges? a large scale empirical study across 20 nlp evaluation tasks. In Proc. of ACL, volume 2, pp. 238–255, 2025.

Lang Cao, Renhong Chen, Yingtian Zou, Chao Peng, Huacong Xu, Yuxian Wang, Wu Ning, Qian Chen, Mofan Peng, Zijie Chen, et al. More bang for the buck: Process reward modeling with entropy-driven uncertainty. arXiv preprint arXiv:2503.22233, 2025.

Jie Cheng, Gang Xiong, Ruixi Qiao, Lijun Li, Chao Guo, Junle Wang, Yisheng Lv, and Fei-Yue Wang. Stop summation: Min-form credit assignment is all process reward model needs for reasoning. In Proc. of NeurIPS, volume 38, pp. 131646–131671, 2025.

Hanze Dong, Wei Xiong, Bo Pang, Haoxiang Wang, Han Zhao, Yingbo Zhou, Nan Jiang, Doyen Sahoo, Caiming Xiong, and Tong Zhang. RLHF workflow: From reward modeling to online RLHF. Trans. Mach. Learn. Res., 2024.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, et al. A survey on llm-as-a-judge. The Innovation, 7(6), 2026.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, et al. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. In Proc. of ACL, volume 1, pp. 3828–3850, 2024a.

Jujie He, Tianwen Wei, Rui Yan, Jiacai Liu, Chaojie Wang, Yimeng Gan, Shiwen Tu, Chris Yuhao Liu, Liang Zeng, Xiaokun Wang, Boyang Wang, Yongcong Li, Fuxiang Zhang, Jiacheng Xu, Bo An, Yang Liu, and Yahui Zhou. Skywork-o1 open series, November 2024b.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. In Proc. of NeurIPS, volume 1, 2021.

R. Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Philip Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. In Proc. of ICLR, 2019.

Muhammad Khalifa, Rishabh Agarwal, Lajanugen Logeswaran, Jaekyeom Kim, Hao Peng, Moontae Lee, Honglak Lee, and Lu Wang. Process reward models that think. Trans. Mach. Learn. Res., 2026.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In Proc. of ICLR, 2024.

Xufei Lv, Kehai Chen, Haoyuan Sun, Xuefeng Bai, Min Zhang, and Houde Liu. The hidden link between rlhf and contrastive learning. arXiv preprint arXiv:2506.22578, 2025.

XuanLong Nguyen, Martin J Wainwright, and Michael I Jordan. Estimating divergence functionals and the likelihood ratio by convex risk minimization. IEEE Transactions on Information Theory, 56(11):5847–5861, 2010.

Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-gan: Training generative neural samplers using variational divergence minimization. Proc. of NeurIPS, 29, 2016.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748, 2018.

Arka Pal, Deep Karkhanis, Samuel Dooley, Manley Roberts, Siddartha Naidu, and Colin White. Smaug: Fixing failure modes of preference optimisation with dpo-positive. arXiv preprint arXiv:2402.13228, 2024.

Tej Deep Pala, Panshul Sharma, Amir Zadeh, Chuan Li, and Soujanya Poria. Error typing for smarter rewards: Improving process reward models with error-aware hierarchical supervision. In Proc. of EMNLP Findings, pp. 14940–14954, 2025.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In Proc. of NeurIPS, 2023.

Shuaijie She, Junxiao Liu, Yifeng Liu, Jiajun Chen, Xin Huang, and Shujian Huang. R-PRM: Reasoning-driven process reward modeling. In Proc. of EMNLP, pp. 13438–13451, 2025.

Mingyang Song, Zhaochen Su, Xiaoye Qu, Jiawei Zhou, and Yu Cheng. Prmbench: A fine-grained and challenging benchmark for process-level reward models. In Proc. of ACL, volume 1, pp. 25299–25346, 2025.

Zhiqing Sun, Longhui Yu, Yikang Shen, Weiyang Liu, Yiming Yang, Sean Welleck, and Chuang Gan. Easy-to-hard generalization: Scalable alignment beyond human supervision. In Proc. of NeurIPS, 2024.

Xiaoyu Tan, Tianchu Yao, Chao Qu, Bin Li, Minghao Yang, Dakuan Lu, Haozhe Wang, Xu Yinghui, and Xihe Qiu. Aurora: Automated training framework of universal process reward models via ensemble prompting and reverse verification. In Proc. of KDD, pp. 1378–1389, 2026.

Zhengyang Tang, Xingxing Zhang, Benyou Wang, and Furu Wei. Mathscale: Scaling instruction tuning for mathematical reasoning. In Proc. of ICML, pp. 47885–47900, 2024.

Jonathan Uesato, Nate Kushman, Ramana Kumar, Francis Song, Noah Siegel, Lisa Wang, Antonia Creswell, Geoffrey Irving, and Irina Higgins. Solving math word problems with process- and outcome-based feedback. In Proc. of NeurIPS Workshop, 2022. URL https://arxiv.org/abs/2211.14275.

Jun Wang, Meng Fang, Ziyu Wan, Muning Wen, Jiachen Zhu, Anjie Liu, Ziqin Gong, Yan Song, Lei Chen, Lionel M Ni, et al. Openr: An open source framework for advanced reasoning with large language models. arXiv preprint arXiv:2410.09671, 2024a.

Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. Math-shepherd: Verify and reinforce llms step-by-step without human annotations. In Proc. of ACL, volume 1, pp. 9426–9439, 2024b.

Chenxi Whitehouse, Tianlu Wang, Ping Yu, Xian Li, Jason Weston, Ilia Kulikov, and Swarnadeep Saha. J1: Incentivizing thinking in llm-as-a-judge via reinforcement learning. arXiv preprint arXiv:2505.10320, 2025.

Zhiheng Xi, Chenyang Liao, Guanyu Li, Zhihao Zhang, Wenxiang Chen, Binghai Wang, Senjie Jin, Yuhao Zhou, Jian Guan, Wei Wu, et al. Agentprm: Process reward models for llm agents via step-wise promise and progress. In Proc. of WWW, pp. 4184–4195, 2026.

Shijie Xia, Xuefeng Li, Yixin Liu, Tongshuang Wu, and Pengfei Liu. Evaluating mathematical reasoning beyond accuracy. In Proc. of AAAI, volume 39, pp. 27723–27730, 2025.

Teng Xiao, Zhen Ge, Sujay Sanghavi, Tian Wang, Julian Katz-Samuels, Marc Versage, Qingjun Cui, and Trishul Chilimbi. Infopo: On mutual information maximization for large language model alignment. In Proc. of NAACL, pp. 11699–11711, 2025.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, Keming Lu, Mingfeng Xue, Runji Lin, Tianyu Liu, Xingzhang Ren, and Zhenru Zhang. Qwen2.5-math technical report: Toward mathematical expert model via self-improvement. arXiv preprint arXiv:2409.12122, 2024.

Hanning Zhang, Pengcheng Wang, Shizhe Diao, Yong Lin, Rui Pan, Hanze Dong, Dylan Zhang, Pavlo Molchanov, and Tong Zhang. Entropy-regularized process reward model. Trans. Mach. Learn. Res., 2025a.

Jianghangfan Zhang, Yibo Yan, Kening Zheng, Xin Zou, Song Dai, and Xuming Hu. Gm-prm: A generative multimodal process reward model for multimodal mathematical reasoning. In Proc. of the 4th Workshop on ALVR, pp. 139–154, 2026a.

Lingyin Zhang, Jun Gao, Xiaoxue Ren, and Ziqiang Cao. The bidirectional process reward model. In Proc. of the ACL, pp. 12564–12580, 2026b.

Zhenru Zhang, Chujie Zheng, Yangzhen Wu, Beichen Zhang, Runji Lin, Bowen Yu, Dayiheng Liu, Jingren Zhou, and Junyang Lin. The lessons of developing process reward models in mathematical reasoning. In Proc. of ACL Findings, pp. 10495–10516, 2025b.

Jian Zhao, Runze Liu, Kaiyan Zhang, Zhimu Zhou, Junqi Gao, Dong Li, Jiafei Lyu, Zhouyi Qian, Biqing Qi, Xiu Li, et al. Genprm: Scaling test-time compute of process reward models via generative reasoning. In Proc. of AAAI, volume 40, pp. 34932–34940, 2026.

Chujie Zheng, Zhenru Zhang, Beichen Zhang, Runji Lin, Keming Lu, Bowen Yu, Dayiheng Liu, Jingren Zhou, and Junyang Lin. Processbench: Identifying process errors in mathematical reasoning. In Proc. of ACL, volume 1, pp. 1009–1024, 2025.

A Understanding the CM-PRM Training Objective

A.1 Energy-Based Interpretation of the Log-Ratio Critic

This appendix explains how the critic used in the main text admits an energy-based model (EBM) interpretation as a reward score. Rather than defining it as a conditional probability, we treat it as

a sample-level scoring function. Given a process state x and a candidate complete evaluation y, the critic assigns a scalar score $T(x, y)$ describing its reweighting strength relative to the reference model.

Consistent with the main text, A denotes the natural-language analysis, $\widehat{G}$ denotes the final judgment, and $Y = (A, \widehat{G})$ denotes the complete evaluation. The shared context is denoted by Z, and the process state is $X = (Z, S_t)$. The critic therefore scores a state–evaluation pair (x, y); since x already contains the shared context z, we omit an additional conditioning argument for z. Throughout, the scoring unit is the complete evaluation, rather than only its final judgment or an individual analysis segment.

From the EBM perspective, a conditional generation policy can be obtained by reweighting a reference model with a sample-level critic. Given a frozen reference model $\pi_{\mathrm{ref}}(y \mid x)$, the critic induces

$$\pi_T(y \mid x) = \frac{\pi_{\mathrm{ref}}(y \mid x) \exp(T(x, y))}{Z_T(x)}, \qquad Z_T(x) = \mathbb{E}_{Y \sim \pi_{\mathrm{ref}}(\cdot|x)}[\exp(T(x, Y))], \tag{17}$$

assuming that the normalizer is finite and positive. The reference model provides the base distribution, while the critic adjusts the relative weights of candidate evaluations. A higher score corresponds to a larger exponential reweighting factor relative to the reference model. Equivalently, the energy function is $E_T(x, y) = -\log \pi_{\mathrm{ref}}(y \mid x) - T(x, y)$.

Taking the log-ratio between π_T and π_{ref} gives

$$\log \frac{\pi_T(y \mid x)}{\pi_{\mathrm{ref}}(y \mid x)} = T(x, y) - \log Z_T(x). \tag{18}$$

Since $Z_T(x)$ depends only on the process state, it cancels when comparing two evaluations under the same state. Thus, for a chosen evaluation $y_w = (a_w, \widehat{g}_w)$ and a rejected evaluation $y_l = (a_l, \widehat{g}_l)$,

$$T(x, y_w) - T(x, y_l) = \log \frac{\pi_T(y_w \mid x)}{\pi_{\mathrm{ref}}(y_w \mid x)} - \log \frac{\pi_T(y_l \mid x)}{\pi_{\mathrm{ref}}(y_l \mid x)}. \tag{19}$$

The critic-score difference therefore equals the difference between the induced policy's log-ratios relative to the reference model. This provides an interpretation of the critic as an energy-based reward score rather than a conditional probability.

We parameterize the critic using the log-ratio between the current and reference models:

$$T_\theta(x, y) = \tau \log \frac{\pi_\theta(y \mid x)}{\pi_{\mathrm{ref}}(y \mid x)}, \tag{20}$$

where $\tau > 0$ is a scaling coefficient. We assume that both policies are positive and normalized over the output space under consideration. The score is positive when the current model assigns a higher probability to an evaluation than the reference model, and negative otherwise.

When $\tau = 1$, this parameterization gives an exact EBM representation of π_θ:

$$\pi_\theta(y \mid x) = \pi_{\mathrm{ref}}(y \mid x) \exp(T_\theta(x, y)). \tag{21}$$

Since π_θ is normalized, $Z_{T_\theta}(x) = 1$. When $\tau \neq 1$, provided that the normalizer is finite, the critic induces a geometrically reweighted distribution:

$$\pi_{T_\theta}(y \mid x) \propto \pi_{\mathrm{ref}}(y \mid x)^{1-\tau} \pi_\theta(y \mid x)^\tau. \tag{22}$$

The critic retains its energy-based interpretation, but under the $\exp(T)$ reweighting convention above, its induced distribution generally differs from π_θ.

For chosen and rejected evaluations, define

$$h_w = \log \frac{\pi_\theta(y_w \mid x)}{\pi_{\mathrm{ref}}(y_w \mid x)}, \qquad h_l = \log \frac{\pi_\theta(y_l \mid x)}{\pi_{\mathrm{ref}}(y_l \mid x)}. \tag{23}$$

Then $T_\theta(x, y_w) = \tau h_w$ and $T_\theta(x, y_l) = \tau h_l$. Pairwise CPC/DPO objectives depend only on score differences, whereas the Pearson and controlled-margin objectives also depend on absolute scores. Although adding a state-dependent offset to a general EBM critic leaves its normalized distribution

unchanged, it changes the latter objectives. We therefore retain the log-ratio scores determined by the current policy, reference policy, and scaling coefficient, rather than arbitrarily recentering them.

In summary, the critic is a sample-level scoring function over complete evaluations $Y = (A, \widehat{G})$, not a conditional probability. Its value is determined by the current model's log-ratio relative to the reference model, providing a basis for expressing preference-training objectives and analyzing the optimization dynamics of h_w and h_l. This EBM interpretation connects the scoring function to the generation policy.

A.2 The Two-Sample InfoNCE Lower Bound

Given a shared context z, the positive sample is drawn jointly with the current state, $(x, y_w) \sim P_{X,Y|z}$, while the remaining $N-1$ negative samples are drawn independently from $P_{Y|z}$. The conditional InfoNCE lower bound is

$$I(X;Y \mid Z) \geq \sup_{\theta} \mathbb{E}_{\substack{z\sim P_Z,\ (x,y_w)\sim P_{X,Y|z} \\ y_{l,1:N-1}^{\mathrm{iid}} \overset{\text{i.i.d.}}{\sim} P_{Y|z}}} \left[\log \frac{e^{T_\theta(x,y_w)}}{\frac{1}{N}\left(e^{T_\theta(x,y_w)} + \sum_{i=1}^{N-1} e^{T_\theta(x,y_{l,i}^{\mathrm{iid}})}\right)} \right]. \tag{24}$$

This follows by applying the standard InfoNCE bound to the conditional distribution given z and then averaging over Z (Oord et al., 2018). The negative samples are independent of the positive pair conditioned on z.

When $N = 2$, there is only one random negative sample y_l^{iid}, giving

$$I(X;Y \mid Z) \geq J_{\mathrm{InfoNCE}} := \sup_{\theta} \mathbb{E}_{\substack{z\sim P_Z,\ (x,y_w)\sim P_{X,Y|z} \\ y_l^{\mathrm{iid}}\sim P_{Y|z}}} \left[\log \frac{2e^{T_\theta(x,y_w)}}{e^{T_\theta(x,y_w)} + e^{T_\theta(x,y_l^{\mathrm{iid}})}} \right]. \tag{25}$$

Dividing both the numerator and denominator by $e^{T_\theta(x,y_w)}$ yields

$$\log \frac{2e^{T_\theta(x,y_w)}}{e^{T_\theta(x,y_w)} + e^{T_\theta(x,y_l^{\mathrm{iid}})}} = \log 2 + \log \sigma\big(T_\theta(x,y_w) - T_\theta(x,y_l^{\mathrm{iid}})\big). \tag{26}$$

Thus, the two-sample InfoNCE bound can be written as

$$J_{\mathrm{InfoNCE}} = \log 2 + \sup_{\theta} \mathbb{E} \log \sigma\big(T_\theta(x,y_w) - T_\theta(x,y_l^{\mathrm{iid}})\big). \tag{27}$$

This expression reveals the connection between InfoNCE and pairwise preference objectives: both apply a log-sigmoid function to the score difference between positive and negative samples.

A.3 Hard-Negative Sampling and the DPO Lower Bound

Random negatives y_l^{iid} are drawn from $P_{Y|z}$, whereas hard negatives y_l are drawn from the state-specific distribution $q(\cdot \mid x, z)$. Both objectives use the same positive-sample distribution, and negative sampling is independent of the positive sample conditioned on x, z.

We motivate score dominance through the Bradley–Terry (BT) model, which expresses preference probabilities in terms of utility differences:

$$P_\theta(y_a \succ y_b \mid x) = \sigma(r_\theta(x,y_a) - r_\theta(x,y_b)). \tag{28}$$

Under the implicit reward parameterization used in DPO,

$$r_\theta(x,y) = \tau \log \frac{\pi_\theta(y \mid x)}{\pi_{\mathrm{ref}}(y \mid x)} + b_\theta(x) = T_\theta(x,y) + b_\theta(x). \tag{29}$$

Thus, utilities and critic scores induce the same ordering for fixed x.

Relevant but incorrect evaluations may retain greater relevance and plausibility than randomly mismatched evaluations, motivating higher preference utilities. Accordingly, we consider the regime in which hard-negative implicit utilities first-order stochastically dominate random-negative utilities conditional on x, z:

$$r_\theta(x,y_l) \succeq_{\mathrm{st}} r_\theta(x,y_l^{\mathrm{iid}}). \tag{30}$$

Since the offset $b_\theta(x)$ is shared, the same stochastic ordering holds for the critic scores.

By the coupling characterization of first-order stochastic dominance, the two conditional score distributions admit a coupling that preserves their respective distributions and satisfies

$$T_\theta(x, y_l^{\text{iid}}) \le T_\theta(x, y_l) \qquad \text{almost surely.} \tag{31}$$

This coupling can be constructed independently of the positive sample conditional on x, z, allowing both objectives to use the same positive sample.

Fixing the same positive sample y_w, monotonicity of the exponential function gives

$$e^{T_\theta(x,y_w)} + e^{T_\theta(x,y_l^{\text{iid}})} \le e^{T_\theta(x,y_w)} + e^{T_\theta(x,y_l)}. \tag{32}$$

Since the two contrastive terms have the same numerator,

$$\log \frac{2e^{T_\theta(x,y_w)}}{e^{T_\theta(x,y_w)} + e^{T_\theta(x,y_l^{\text{iid}})}} \ge \log \frac{2e^{T_\theta(x,y_w)}}{e^{T_\theta(x,y_w)} + e^{T_\theta(x,y_l)}}. \tag{33}$$

Taking expectations shows that the random-negative objective is no smaller than the hard-negative objective. The former is also bounded above by its supremum over the critic parameters, hence

$$J_{\text{InfoNCE}} \ge J^{\text{HG}}_{\text{InfoNCE}}(T_\theta). \tag{34}$$

Finally, substituting $T_\theta(x, y) = \tau \log[\pi_\theta(y \mid x)/\pi_{\text{ref}}(y \mid x)]$ and using h_w, h_l from Eq. 3, we obtain

$$J^{\text{HG}}_{\text{InfoNCE}}(T_\theta) = \mathbb{E}_{\mathcal{D}_q} \log \frac{2e^{\tau h_w}}{e^{\tau h_w} + e^{\tau h_l}} = \log 2 + \mathbb{E}_{\mathcal{D}_q} \log \sigma(\tau(h_w - h_l)) = \log 2 + J_{\text{DPO}}(\theta). \tag{35}$$

Combining this identity with the InfoNCE bound derived above yields

$$\boxed{I(X; Y \mid Z) \ge J_{\text{InfoNCE}} \ge J^{\text{HG}}_{\text{InfoNCE}}(T_\theta) = \log 2 + J_{\text{DPO}}(\theta).} \tag{36}$$

Since $\log 2$ is independent of the model parameters, minimizing the DPO loss is equivalent to maximizing this hard-negative contrastive lower bound.

B f-DIVERGENCE SELECTION FOR CM-PRM

B.1 GRADIENT STRUCTURE OF DIFFERENT DIVERGENCES

Let the log-density ratio generated by the model be

$$h(y|x) = \log \frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}, \tag{37}$$

where h_w and h_l denote the log-density ratios of the positive and negative samples, respectively. In the variational representation of f-divergence, we parameterize the critic as

$$T(y, x) = \tau h(y|x), \tag{38}$$

where τ is a temperature coefficient.

Given the NWJ variational form

$$J_f = \mathbb{E}_P[T] - \mathbb{E}_Q[f^*(T)], \tag{39}$$

the pairwise objective for a positive sample y_w and a negative sample y_l can be written as

$$J_f(h_w, h_l) = \tau h_w - f^*(\tau h_l). \tag{40}$$

We compare different choices of f-divergence by analyzing their effective gradients with respect to h_w and h_l.

Pearson-χ^2 Divergence. For the Pearson-χ^2 divergence, the conjugate function is

$$f^*(t) = t + \frac{t^2}{4}, t \geq -2. \tag{41}$$

The exact conjugate equals -1 for $t < -2$. We extend the quadratic branch to all real values to construct the training objective. This extension upper-bounds the exact conjugate and therefore preserves the lower-bound property under the stated negative-sampling replacement condition. Substituting $T = \tau h$ gives

$$J_{\chi^2} = \tau h_w - \left(\tau h_l + \frac{\tau^2}{4} h_l^2\right) = \tau h_w - \tau h_l - \frac{\tau^2}{4} h_l^2. \tag{42}$$

Taking partial derivatives yields

$$\frac{\partial J_{\chi^2}}{\partial h_w} = \tau, \qquad \frac{\partial J_{\chi^2}}{\partial h_l} = -\tau - \frac{\tau^2}{2} h_l. \tag{43}$$

The positive sample receives a constant update coefficient, while the negative-sample gradient depends only on its own log-density ratio h_l. Therefore, the optimization signals for positive and negative samples are decoupled from the relative margin $h_w - h_l$.

At the beginning of training, the policy is close to the reference model, so $h_w \approx 0$ and $h_l \approx 0$. In this region, the negative-sample gradient of Pearson-χ^2 is approximately $-\tau$. Once optimization starts to suppress the negative sample and makes $h_l < 0$, its magnitude becomes

$$\left|-\tau - \frac{\tau^2}{2} h_l\right| = \tau - \frac{\tau^2}{2}|h_l|, \qquad h_l < 0. \tag{44}$$

Thus, near the initial training stage, Pearson-χ^2 maintains a relatively strong negative-sample update while still avoiding relative-margin coupling. Its polynomial form also makes it convenient to combine with quadratic controlled-margin regularization.

KL Divergence. For the KL divergence, the conjugate function is

$$f^*(t) = e^{t-1}. \tag{45}$$

Substituting $T = \tau h$ gives

$$J_{\mathrm{KL}} = \tau h_w - e^{\tau h_l - 1}. \tag{46}$$

The corresponding partial derivatives are

$$\frac{\partial J_{\mathrm{KL}}}{\partial h_w} = \tau, \qquad \frac{\partial J_{\mathrm{KL}}}{\partial h_l} = -\tau e^{\tau h_l - 1}. \tag{47}$$

This objective also decouples the positive and negative samples because the negative-sample gradient depends only on h_l. However, when $h_l \approx 0$, its magnitude is

$$\left|-\tau e^{\tau h_l - 1}\right| \approx \frac{\tau}{e}, \tag{48}$$

which is much smaller than the Pearson-χ^2 gradient magnitude τ near initialization. Therefore, KL-NWJ provides a weaker early-stage suppression signal for negative samples. In addition, its exponential form can lead to rapidly changing gradients when h_l becomes large.

Reverse KL Divergence. For the reverse KL divergence, the conjugate function is

$$f^*(t) = -1 - \log(-t), \tag{49}$$

which requires $t < 0$. Under the unified log-ratio critic $T = \tau h$, the objective becomes

$$J_{\mathrm{RKL}} = \tau h_w - [-1 - \log(-\tau h_l)] = \tau h_w + 1 + \log(-\tau h_l), \qquad \tau h_l < 0. \tag{50}$$

Since the constant term does not affect optimization, we equivalently use

$$\widetilde{J}_{\mathrm{RKL}} = \tau h_w + \log(-\tau h_l), \qquad h_l < 0. \tag{51}$$

The corresponding partial derivatives are

$$\frac{\partial \widetilde{J}_{\mathrm{RKL}}}{\partial h_w} = \tau, \qquad \frac{\partial \widetilde{J}_{\mathrm{RKL}}}{\partial h_l} = \frac{1}{h_l}. \tag{52}$$

This objective also decouples the chosen and rejected samples because the negative-sample gradient depends only on h_l. However, it introduces a strict domain constraint $h_l < 0$ under the linear critic $T = \tau h$. This is problematic near initialization, where $h_l \approx 0$: the objective is not defined at $h_l = 0$, and the negative-sample gradient magnitude

$$\left| \frac{1}{h_l} \right| \tag{53}$$

can become arbitrarily large as $h_l \to 0^-$. Therefore, reverse KL is less compatible with the unified linear log-ratio critic.

Jensen-Shannon Divergence. For the Jensen-Shannon divergence, the conjugate function is

$$f^*(t) = -\log(2 - e^t), \tag{54}$$

which requires $t < \log 2$. Therefore, the pairwise objective is

$$J_{\mathrm{JS}} = \tau h_w + \log(2 - e^{\tau h_l}), \qquad \tau h_l < \log 2. \tag{55}$$

Taking partial derivatives gives

$$\frac{\partial J_{\mathrm{JS}}}{\partial h_w} = \tau, \qquad \frac{\partial J_{\mathrm{JS}}}{\partial h_l} = -\frac{\tau e^{\tau h_l}}{2 - e^{\tau h_l}}. \tag{56}$$

At $h_l = 0$, the negative-sample gradient magnitude is also τ. However, when the negative sample starts to be suppressed and $h_l < 0$, we have

$$\left| -\frac{\tau e^{\tau h_l}}{2 - e^{\tau h_l}} \right| = \frac{\tau e^{-\tau |h_l|}}{2 - e^{-\tau |h_l|}} \approx \tau - 2\tau^2 |h_l|, \qquad h_l < 0. \tag{57}$$

This decreases faster than the Pearson-χ^2 gradient magnitude

$$\tau - \frac{\tau^2}{2} |h_l|. \tag{58}$$

Thus, in the early stage after negative-sample suppression begins, the JS-NWJ objective provides a weaker negative-sample update than Pearson-χ^2. It also has a restricted valid region because the logarithmic term requires $\tau h_l < \log 2$.

Squared Hellinger Divergence. For the squared Hellinger divergence, the conjugate function is

$$f^*(t) = \frac{t}{1 - t}, \tag{59}$$

which requires $t < 1$. Substituting $T = \tau h$ gives

$$J_{\mathrm{SH}} = \tau h_w - \frac{\tau h_l}{1 - \tau h_l}, \qquad \tau h_l < 1. \tag{60}$$

The corresponding partial derivatives are

$$\frac{\partial J_{\mathrm{SH}}}{\partial h_w} = \tau, \qquad \frac{\partial J_{\mathrm{SH}}}{\partial h_l} = -\frac{\tau}{(1 - \tau h_l)^2}. \tag{61}$$

This objective also yields a self-dependent negative-sample gradient and avoids explicit dependence on the relative margin $h_w - h_l$. However, it introduces the domain constraint $\tau h_l < 1$, and its rational form may lead to large gradients near the boundary. We therefore do not adopt it as the main divergence choice in CM-PRM.

B.2 Second-Order Structure of Pearson-χ^2

Pearson-χ^2 has an exact quadratic structure: its generator is a quadratic polynomial, and its conjugate likewise contains no third- or higher-order terms near the critic origin. This preserves the local second-order form shared by smooth divergences while yielding linear gradient coefficients and constant curvature, providing a simple objective structure for controlling positive and negative updates.

Specifically, let $r = dP/dQ = 1 + \delta$. For a generator satisfying $f(1) = 0$ and sufficient smoothness near 1,

$$f(1+\delta) = f'(1)\delta + \frac{f''(1)}{2}\delta^2 + O(|\delta|^3). \tag{62}$$

Since $\mathbb{E}_Q[\delta] = 0$, we obtain

$$D_f(P\|Q) = \frac{f''(1)}{2}\mathbb{E}_Q[\delta^2] + O\big(\mathbb{E}_Q[|\delta|^3]\big)\,. \tag{63}$$

Thus, when the density ratio is close to 1, smooth f-divergences share a scaled Pearson-χ^2 divergence as their leading second-order term. For $f(r) = (r-1)^2$, this second-order expression is exact, with no higher-order remainder.

Let s denote the expansion variable. We expand the conjugates of KL, JS, and squared Hellinger around the critic origin. Since the domain of the reverse-KL conjugate excludes the origin, we instead expand it at $t = -1 + s$ around $s = 0$.

For KL, the Taylor expansion of the exponential gives

$$f^*_{\mathrm{KL}}(s) = e^{s-1} = \frac{1}{e}\left(1 + s + \frac{s^2}{2} + \frac{s^3}{6}\right) + O(s^4). \tag{64}$$

Hence, beyond the constant, linear, and quadratic terms, the exponential introduces higher-order corrections.

For reverse KL, using $-\log(1-s) = \sum_{k=1}^{\infty} s^k/k$, we obtain

$$f^*_{\mathrm{RKL}}(-1+s) = -1 - \log(1-s) = -1 + s + \frac{s^2}{2} + \frac{s^3}{3} + O(s^4). \tag{65}$$

Its logarithmic structure likewise produces nonzero third- and higher-order terms.

For JS, let $u = e^s - 1 = s + s^2/2 + s^3/6 + O(s^4)$. Then

$$\begin{aligned} f^*_{\mathrm{JS}}(s) &= -\log(2 - e^s) = -\log(1-u) \\ &= u + \frac{u^2}{2} + \frac{u^3}{3} + O(u^4) \\ &= s + s^2 + s^3 + O(s^4). \end{aligned} \tag{66}$$

For squared Hellinger, the geometric-series expansion gives

$$f^*_{\mathrm{SH}}(s) = \frac{s}{1-s} = s(1 + s + s^2 + \cdots) = s + s^2 + s^3 + O(s^4). \tag{67}$$

In contrast, the Pearson-χ^2 conjugate satisfies

$$f^*_{\chi^2}(s) = s + \frac{s^2}{4}, \qquad s \geq -2, \tag{68}$$

exactly on its quadratic branch containing the origin. Within this branch, all third- and higher-order derivatives vanish. Substituting $s = \tau h$ yields the pairwise objective

$$J_{\chi^2} = \tau h_w - \tau h_l - \frac{\tau^2}{4}h_l^2, \tag{69}$$

whose negative-sample gradient and curvature are

$$\frac{\partial J_{\chi^2}}{\partial h_l} = -\tau - \frac{\tau^2}{2}h_l, \qquad \frac{\partial^2 J_{\chi^2}}{\partial h_l^2} = -\frac{\tau^2}{2}. \tag{70}$$

The negative-sample update coefficient therefore varies linearly with the log-ratio, while the curvature remains constant. Other divergences introduce nonlinear updates and varying curvature through higher-order terms, including exponential curvature for KL and increasing curvature near the respective domain boundaries for reverse KL, JS, and squared Hellinger.

Pearson therefore provides a structurally simple quadratic contrastive objective that naturally accommodates the controlled-margin quadratic regularization introduced subsequently. Adding quadratic control preserves this structure, allowing explicit analysis of the positive and negative update coefficients and their target scales.

C Proofs of Theoretical Properties of CM-PRM

C.1 Coordinate-wise Optimum and Target Margin

To simplify the notation, let the log density-ratios of the positive and negative samples be $h_w = \log \frac{\pi_\theta(y_w|x)}{\pi_{\text{ref}}(y_w|x)}$ and $h_l = \log \frac{\pi_\theta(y_l|x)}{\pi_{\text{ref}}(y_l|x)}$, respectively. According to the definition in the main text, the objective function can be abbreviated as:

$$\mathcal{L}_{\text{CM}} = -\tau h_w + \tau h_l + \frac{\beta\tau^2}{2} h_w^2 + \frac{\gamma\tau^2}{2} h_l^2. \tag{71}$$

Proof of Bounded Margin:
Taking the partial derivatives of $\mathcal{L}_{\text{CM}}$ with respect to h_w and h_l yields:

$$\frac{\partial \mathcal{L}_{\text{CM}}}{\partial h_w} = -\tau + \beta\tau^2 h_w, \tag{72}$$

$$\frac{\partial \mathcal{L}_{\text{CM}}}{\partial h_l} = \tau + \gamma\tau^2 h_l. \tag{73}$$

Since the second-order derivatives of the objective function are $\frac{\partial^2 \mathcal{L}_{\text{CM}}}{\partial h_w^2} = \beta\tau^2 > 0$ and $\frac{\partial^2 \mathcal{L}_{\text{CM}}}{\partial h_l^2} = \gamma\tau^2 > 0$, its Hessian matrix is a positive-definite diagonal matrix. Setting the above partial derivatives to 0 yields the unique global stable stationary point:

$$h_w^* = \frac{1}{\beta\tau}, \qquad h_l^* = -\frac{1}{\gamma\tau}. \tag{74}$$

Consequently, the margin at the stationary point is $\Delta^* = h_w^* - h_l^* = \frac{1}{\beta\tau} + \frac{1}{\gamma\tau}$, which proves the bounded nature of the optimization margin. This identifies the unique minimizer and finite target margin in the space of independent log-ratio coordinates. With shared model parameters, this result does not guarantee that all sample-wise log-ratios remain bounded throughout training or simultaneously attain their coordinate-wise optima.

Proof of Directional Validity:
Using the chain rule, the gradient of the objective function with respect to the parameters θ, $\nabla_\theta \mathcal{L}_{\text{CM}}$, can be expanded as a linear combination of the log-likelihood gradients:

$$\nabla_\theta \mathcal{L}_{\text{CM}} = \underbrace{(\beta\tau^2 h_w - \tau)}_{\text{GC}(x, y_w)} \nabla_\theta \log \pi_\theta(y_w \mid x) + \underbrace{(\tau + \gamma\tau^2 h_l)}_{\text{GC}(x, y_l)} \nabla_\theta \log \pi_\theta(y_l \mid x), \tag{75}$$

where $\text{GC}(\cdot)$ represents the Gradient Coefficient. Under the common initialization state during model training where $\pi_\theta \approx \pi_{\text{ref}}$, we have $h_w \approx 0$ and $h_l \approx 0$. Substituting these into the above equation gives:

$$\text{GC}(x, y_w) \approx -\tau < 0, \qquad \text{GC}(x, y_l) \approx \tau > 0. \tag{76}$$

When performing gradient descent (i.e., updating along the direction of $-\nabla_\theta \mathcal{L}_{\text{CM}}$), the overall gradient coefficient for the positive sample log-likelihood is positive, and the overall gradient coefficient for the negative sample log-likelihood is negative. These coefficients provide the intended reinforcement and suppression signals.

The decoupling above concerns the chosen and rejected gradient coefficients, not the actual probability changes under shared model parameters. Parameter updates remain subject to interactions between sample gradients. Thus, the coefficients provide reinforcement and suppression signals but do not guarantee monotonic increases and decreases in the corresponding log-ratios. We assess the resulting training dynamics empirically.

C.2 First-Order Equivalence with DPO/IPO and Adaptive Corrections

For notational convenience, define

$$g_w := \nabla_\theta \log \pi_\theta(y_w \mid x), \qquad g_l := \nabla_\theta \log \pi_\theta(y_l \mid x), \tag{77}$$

and

$$\Delta := h_w - h_l. \tag{78}$$

Derivation. We write the policy gradients of DPO, IPO, and CM in the unified form

$$\nabla_\theta \mathcal{L} = -\tau g_w + \tau g_l + \mathcal{O}(\rho), \tag{79}$$

where the first two terms are the shared first-order part, and $\mathcal{O}(\rho)$ denotes the method-specific higher-order correction term.

For DPO, the loss is

$$\mathcal{L}_{\mathrm{DPO}} = -\log \sigma(\beta\Delta). \tag{80}$$

Its gradient is

$$\nabla_\theta \mathcal{L}_{\mathrm{DPO}} = -\beta\sigma(-\beta\Delta)(g_w - g_l). \tag{81}$$

By setting $\tau = \beta/2$, we obtain

$$\nabla_\theta \mathcal{L}_{\mathrm{DPO}} = -\tau g_w + \tau g_l + \big(\beta\sigma(-\beta\Delta) - \tau\big)(-g_w + g_l). \tag{82}$$

Using the identity

$$2\sigma(z) - 1 = \tanh(z/2), \tag{83}$$

the higher-order term can be written as

$$\mathcal{O}_{\mathrm{DPO}}(\rho) = \tau \tanh\big(\tau\Delta\big)(g_w - g_l). \tag{84}$$

For IPO, its gradient can be rearranged as

$$\nabla_\theta \mathcal{L}_{\mathrm{IPO}} = (-\tau + 2\Delta)g_w + (\tau - 2\Delta)g_l. \tag{85}$$

Extracting the shared first-order part gives

$$\nabla_\theta \mathcal{L}_{\mathrm{IPO}} = -\tau g_w + \tau g_l + 2\Delta(g_w - g_l), \tag{86}$$

and hence

$$\mathcal{O}_{\mathrm{IPO}}(\rho) = 2(h_w - h_l)(g_w - g_l). \tag{87}$$

For CM, the loss is

$$\mathcal{L}_{\mathrm{CM}} = -\tau h_w + \tau h_l + \lambda_1 \tau^2 h_w^2 + \left(\frac{1}{4} + \lambda_2\right)\tau^2 h_l^2. \tag{88}$$

Its gradient is

$$\nabla_\theta \mathcal{L}_{\mathrm{CM}} = -\tau g_w + \tau g_l + 2\tau^2 \left[\lambda_1 h_w g_w + \left(\frac{1}{4} + \lambda_2\right) h_l g_l\right]. \tag{89}$$

Therefore,

$$\mathcal{O}_{\mathrm{CM}}(\rho) = 2\tau^2 \left[\lambda_1 h_w g_w + \left(\frac{1}{4} + \lambda_2\right) h_l g_l\right]. \tag{90}$$

Early-stage As $\rho \to 0$. Let

$$\rho := \sup_{(x,y)\in\mathcal{D}} |h_\theta(x, y)|, \tag{91}$$

denote the upper bound of the absolute log-density ratio. Assume there exists a constant $G > 0$ such that

$$\|g_w\| \le G, \qquad \|g_l\| \le G. \tag{92}$$

Since $|h_w| \le \rho$ and $|h_l| \le \rho$, we have

$$|\Delta| = |h_w - h_l| \le 2\rho. \tag{93}$$

For IPO,

$$\|\mathcal{O}_{\mathrm{IPO}}(\rho)\| = 2|\Delta|\,\|g_w - g_l\| \le 2(2\rho)(2G) = 8G\rho. \tag{94}$$

For DPO, using $|\tanh z| \le |z|$, we have

$$\|\mathcal{O}_{\mathrm{DPO}}(\rho)\| \le \tau|\tau\Delta|\,\|g_w - g_l\| \le \tau^2(2\rho)(2G) = 4\tau^2 G\rho. \tag{95}$$

For CM,

$$\|\mathcal{O}_{\mathrm{CM}}(\rho)\| \le 2\tau^2\left[\lambda_1|h_w|\|g_w\| + \left(\frac{1}{4}+\lambda_2\right)|h_l|\|g_l\|\right] \le 2\tau^2 G\left[\lambda_1 + \frac{1}{4} + \lambda_2\right]\rho. \tag{96}$$

Therefore, the higher-order terms of all three methods are at most $O(\rho)$. When $\rho \to 0$, they vanish and become negligible relative to the shared first-order part. This shows that DPO, IPO, and CM have the same first-order optimization form at the early stage of training, and CM does not lose early-stage optimization efficiency due to its additional higher-order term.

C.3 Variance Bound of the CM Empirical Estimator

We derive an asymptotic variance bound for the CM empirical objective under explicit sampling, boundedness, and estimator regularity assumptions. Let P_w and P_l denote the distributions of chosen and rejected context–evaluation pairs, respectively, and let Var_w and Var_l denote variances under these distributions. Define

$$\begin{aligned} A_\theta(x,y) &= \tau h_\theta(x,y) - \lambda_1\tau^2 h_\theta(x,y)^2, \\ B_\theta(x,y) &= \tau h_\theta(x,y) + \left(\frac{1}{4}+\lambda_2\right)\tau^2 h_\theta(x,y)^2. \end{aligned} \tag{97}$$

The population and empirical objectives are

$$\begin{aligned} J(\theta) &= \mathbb{E}_w[A_\theta] - \mathbb{E}_l[B_\theta], \\ \hat{J}^{n,m}(\theta) &= \frac{1}{n}\sum_{i=1}^{n} A_\theta(x_i^w, y_{w,i}) - \frac{1}{m}\sum_{j=1}^{m} B_\theta(x_j^l, y_{l,j}). \end{aligned} \tag{98}$$

Let $\theta^\star$ be a population maximizer and $\hat{\theta}$ an interior stationary point of the empirical objective. Write $\hat{J}^{n,m}_{\mathrm{CM}} = \hat{J}^{n,m}(\hat{\theta})$. The population optimum need not attain all coordinate-wise optimal log-ratios characterized in Appendix C.1.

We assume i.i.d. sampling within each group, while allowing dependence between the two groups to accommodate preference pairs sharing the same context. Furthermore, assume

$$|h_{\theta^\star}(x,y_w)| \le M_w < \infty, \qquad |h_{\theta^\star}(x,y_l)| \le M_l < \infty \tag{99}$$

almost surely under P_w and P_l, respectively.

To control the contribution of parameter estimation to the variance, assume that the empirical objective is twice continuously differentiable and that, almost surely, its Hessian has operator norm at most a deterministic constant H, independent of sample size, along the line segment joining $\hat{\theta}$ and $\theta^\star$. We additionally assume

$$\mathbb{E}\|\hat{\theta} - \theta^\star\|^4 = O(n'^{-2}), \qquad n' = \min\{n,m\}. \tag{100}$$

Since $\nabla_\theta \hat{J}^{n,m}(\hat{\theta}) = 0$, a Taylor expansion yields

$$\begin{aligned} \hat{J}^{n,m}(\hat{\theta}) &= \hat{J}^{n,m}(\theta^\star) + r_{n,m}, \\ |r_{n,m}| &\le \frac{H}{2}\|\hat{\theta} - \theta^\star\|^2. \end{aligned} \tag{101}$$

Consequently, $\mathbb{E}[r_{n,m}^2] = O(n'^{-2}) = o(1/n')$, providing the mean-square remainder control required for the variance analysis.

Problem

Find the area of an equiangular octagon with side lengths 1, 2, 2, 4, 1, 2, 2, 4, in that order.

Now Step

Step 1: To find the area of the equiangular octagon with side lengths 1, 2, 2, 4, 1, 2, 2, 4 in that order, we can break it down into simpler shapes. **Notice that the octagon can be divided into two congruent rectangles and two congruent right triangles.**

Wrong Analyse

Previous Steps Analysis: There are no previous steps provided to analyze.
Now Step Analysis: The Now Step proposes breaking down the equiangular octagon into simpler geometric shapes to calculate its area. ***This approach is a valid strategy for finding the area of complex shapes, as areas of these simpler shapes can be calculated and then summed to find the total area.***
Data Source Analysis: ...
Consistency Analysis: The approach suggested in the Now Step is consistent with geometric principles for calculating areas of complex shapes. ***Breaking down a shape into simpler components is a standard technique in geometry, and the claim of dividing the octagon into two congruent rectangles and two congruent right triangles is plausible given the symmetry of an equiangular octagon.***
Calculation Analysis: …
Verification: Is the step correct (Yes/No)? Yes

Figure 7: Case 1 - Insufficient Verification

Let $A = A_{\theta^\star}$ and $B = B_{\theta^\star}$, with empirical means $\bar{A}_n$ and $\bar{B}_m$, respectively. Then

$$\hat{J}^{n,m}_{\mathrm{CM}} - J(\theta^\star) = (\bar{A}_n - \mathbb{E}_w[A]) - (\bar{B}_m - \mathbb{E}_l[B]) + r_{n,m}. \tag{102}$$

Under the boundedness assumptions, define

$$K_w = \tau M_w + \lambda_1 \tau^2 M_w^2, \qquad K_l = \tau M_l + \left(\frac{1}{4} + \lambda_2\right) \tau^2 M_l^2. \tag{103}$$

We have $|A| \le K_w$ and $|B| \le K_l$, implying

$$\mathrm{Var}_w(A) \le K_w^2, \qquad \mathrm{Var}_l(B) \le K_l^2. \tag{104}$$

Allowing dependence between the groups and using $\mathrm{Var}(U - V) \le 2\,\mathrm{Var}(U) + 2\,\mathrm{Var}(V)$, we obtain

$$\begin{aligned} \mathrm{Var}(\bar{A}_n - \bar{B}_m) &\le \frac{2\,\mathrm{Var}_w(A)}{n} + \frac{2\,\mathrm{Var}_l(B)}{m} \\ &\le \frac{2K_w^2}{n} + \frac{2K_l^2}{m}. \end{aligned} \tag{105}$$

The remainder has variance $o(1/n')$. By the Cauchy–Schwarz inequality, its covariance with the leading term is also $o(1/n')$. Therefore, setting $C_1 = 2K_w^2$ and $C_2 = 2K_l^2$ gives

$$\boxed{\mathrm{Var}\left[\hat{J}^{n,m}_{\mathrm{CM}}\right] \le \frac{C_1}{n} + \frac{C_2}{m} + o\left(\frac{1}{n'}\right).} \tag{106}$$

This completes the proof under the stated assumptions. This variance bound is conditional on the stated assumptions. Quadratic regularization provides an optimization mechanism for controlling log-ratio scales but does not, by itself, guarantee that these assumptions hold.

Problem

Evaluate the definite integral: $I = \int_{-1}^{2} xe^{x^2}\,dx$.

Now Step

Step 1: Let $u = x^2$, so $du = 2xdx$. The endpoints $x = -1$ and $x = 2$ correspond to $u = 1$ and $u = 4$, respectively. Therefore, $I = \frac{1}{2}\int_1^4 e^u\,du = \frac{1}{2}(e^4 - e)$.

Wrong Analyse

Previous Steps Analysis: This is the first step, so there are no previous steps to assess.

Now Step Analysis: The step introduces $u = x^2$, correctly obtains $du = 2x\,dx$, and maps the original endpoints to $u = 1$ and $u = 4$. ***However, $u = x^2$ is not monotonic over $[-1, 2]$. The integral must first be split at $x = 0$; directly replacing the endpoints without separating the two branches makes the substitution invalid.***

Data Source Analysis: The integrand xe^{x^2} and the bounds $[-1,2]$ are taken directly from the problem …

Consistency Analysis: The substitution decreases from $u = 1$ to $u = 0$, then increases to $u = 4$. ***Using only the endpoint values ignores this change in direction and therefore fails to account for the entire integration interval.***

Calculation Analysis: This calculation does not resolve the substitution issue identified above, so it does not establish the validity of the original step.

Verification: Is the step correct (Yes/No)? No

Figure 8: Local Analysis Overreliance

D Failure Cases

This appendix presents two representative case studies illustrating typical failure modes of generative PRMs in process evaluation. The first case shows an *Unconfident Evaluation*, in which the model issues an overly loose judgment without sufficient verification. The second case shows *Local Analysis Overreliance*, where the model treats a local analysis result as sufficient grounds for a final judgment without considering other relevant evidence that establishes the step's validity.

Case 1: Insufficient Verification The insufficient-verification case is shown in Figure 7. The error in this analysis is that it only affirms the general strategy of decomposing a complex geometric shape into simpler components, without verifying whether the given equiangular octagon can indeed be decomposed into two congruent rectangles and two congruent right triangles. An equiangular octagon does not automatically imply this specific geometric decomposition. The analysis also fails to check the side-length order, angle relations, and dimensional consistency of the proposed components, and therefore prematurely judges an unverified construction as correct.

Case 2: Overreliance on Local Analysis. Figure 8 illustrates an incorrect judgment driven by overreliance on a local analysis result. The evaluation correctly observes that $u = x^2$ is non-monotonic over $[-1, 2]$, but treats this observation as sufficient to invalidate the substitution. It overlooks that the integrand contains the corresponding derivative factor, so the substitution remains valid by the chain rule without requiring monotonicity. Equivalently, splitting at $x = 0$ yields $\frac{1}{2}\int_1^0 e^u\,du + \frac{1}{2}\int_0^4 e^u\,du = \frac{1}{2}\int_1^4 e^u\,du$, confirming the original step. The failure therefore lies not in identifying non-monotonicity, but in binding this local finding to a negative verdict without considering the mathematical evidence that establishes the step's validity.

Prompt of CM-PRM

```
### Instruction:
You are an excellent math teacher. Please verify the correctness of the Now Step.

You first need to analyze the Now Step and the Previous Steps and then summarize based on your analysis.

Analysis:
You need to analyze the following aspects.
**Previous Steps Analysis**: You need to analyze the Previous Steps step by step. For each step, you need to first explain what the current step is doing, then you try to find any error in the current step.
**Now Step Analysis**: You first need to explain what the Now Step is doing, and then point out which part of the Question it is trying to solve or which part of the information it states.
**Data Source Analysis**: First you need to find out what data are used in the Now Step, and then you need to determine whether the source of the data is reasonable and correct. When you judge whether the source of a data is reasonable and correct, you need to specify the specific source of this data: such as which part of the question, or which content of the previous step; and then determine the source and current use is consistent, the Now Step is used correctly.
**Consistency Analysis**: You need to check that the Now Step is consistent with the contents of the Previous Steps, and then you need to check that all the information inside the Now Step is consistent.
**Calculation Analysis**: If the Now Step involves any calculations, such as addition, subtraction, multiplication, division, equations, modulo operations, etc., you will first need to perform a check on the calculation, such as a reverse operation, to see if the calculation was done correctly, and then analyze the results of your check to see if there was an error in the calculation.

Conclusion:
Please verify the correctness of the Now Step based on your analysis, if there is any error in the Now Step then the Now Step is wrong and vice versa the Now Step is correct. At the end of the Conclusion, when you give your final answer, write it in the form "Verification: Is the step correct (Yes/No)? X", where X is either Yes or No.

Question: {},
Previous Steps: {}
Now Step: {)
Please carefully analyze the correctness of the Now Step.

Reply:

### Response:
```

Figure 9: Prompt used for CM-PRM inference. Following R-PRM (She et al., 2025), the prompt asks the model to analyze the current reasoning step from multiple verification aspects and produce a final Yes/No correctness judgment.

Prompt of Policy Model in Greedy Guided Search

```
You are Qwen, created by Alibaba Cloud. You are a helpful math problem solving assistant. Please reason step by step,
and put your final answer within \\boxed{{}}.


{question}
{previous_steps}
```

Figure 10: Prompt used for policy-model generation in PRM-guided greedy search. The policy model generates step-by-step reasoning continuations and outputs the final answer in boxed format.

E Inference Prompts for CM-PRM Evaluation and Guided Search

This appendix presents the prompts used during inference. Figure 9 shows the prompt for CM-PRM evaluation. Given a question, previous reasoning steps, and the current step, CM-PRM is required to analyze the step from multiple aspects, including previous-step consistency, data sources, logical consistency, and calculation correctness, and then output an explicit Yes/No verification judgment. Figure 10 shows the prompt for the policy model used in PRM-guided greedy search. The policy model, Qwen2.5-7B-Instruct, generates step-by-step reasoning continuations and is instructed to place the final answer in `\boxed{}`.

F Training Configuration

F.1 Two-Stage CM-PRM Training Implementation

We train CM-PRM on 2 A100 40G GPUs and use Qwen2.5-Math-7B-Instruct as the base model. The training process consists of two stages: supervised fine-tuning (SFT) and controlled-margin contrastive learning (CM-CL). In the SFT stage, we search the learning rate from $\{5 \times 10^{-6}, 1 \times 10^{-5}, 5 \times 10^{-5}\}$ and select 1×10^{-5} as the final value. We train the model for one epoch with a batch size of 128. We use DeepSpeed ZeRO-2 for training acceleration. We also apply LoRA to reduce the training cost, with rank $r = 32$ and $\alpha = 128$.

In the CM-CL stage, we search the learning rate from $\{5 \times 10^{-7}, 1 \times 10^{-6}, 5 \times 10^{-6}\}$ and select 1×10^{-6}. For the controlled-margin hyperparameters (β, γ), we search over $\{(15, 30), (30, 60), (60, 120), (120, 240)\}$ and use $(30, 60)$ in the final model. This stage is also trained for one epoch with a batch size of 128. We use DeepSpeed ZeRO-2, LoRA, and gradient checkpointing to improve training efficiency and reduce memory usage. The LoRA configuration is the same as in the SFT stage, with $r = 32$ and $\alpha = 128$. The CM-CL training code is implemented based on the TRL library.

For methods evaluated repeatedly in this work, reported results are averaged over 10 independent evaluation runs using random seeds $\{42, 123, 2026, 3407, 7777, 17, 314, 1024, 4096, 10007\}$. Each method uses a fixed trained checkpoint across all runs, with evaluation trajectories resampled in each run. Across-run variability therefore reflects inference-time generation randomness rather than variability from retraining. All methods use the same test samples and matched random seeds.

For CM-PRM, the across-run standard deviations on PRMBench, ProcessBench, and PRM-guided Greedy Search are 0.7, 1.0, and 2.4, respectively. Statistical significance is assessed using two-sided paired t-tests on the evaluation results from the 10 matched-seed runs, against the baselines specified in the corresponding table captions. * and ** indicate $p < 0.05$ and $p < 0.01$, respectively.

Following R-PRM's validation-set setup, we use 20K samples from the R-PRM dataset for validation, but select checkpoints based on F1 rather than validation loss. We evaluate every 100 training steps and retain the checkpoint with the highest validation F1. The final test sets are not used for model selection. For DPO and IPO, we search their respective objective hyperparameters over $\{0.05, 0.1, 0.5\}$ and select 0.1 for both methods. All other shared training settings remain unchanged.

F.2 Baselines and Ablation Variants

We organize the comparison methods into four categories: LLM-as-a-Judge, discriminative PRMs, generative PRMs, and ablation variants. The first three assess CM-PRM against existing process-evaluation methods, while the fourth examines the roles of explicit analysis trajectories, training objectives, and regularization.

LLM-as-a-Judge. We include GPT-4o, o1-mini, Qwen2.5-Math-72B-Instruct, and QwQ-Preview-32B, prompting them to evaluate reasoning steps and produce correctness judgments. These models cover open-source and proprietary models at different scales, enabling comparisons with prompt-based generative evaluators.

Discriminative PRMs. We include Math-Shepherd (Wang et al., 2024b), Math-PSA (Wang et al., 2024a), RLHFlow-Mistral, RLHFlow-DeepSeek (Dong et al., 2024), Skywork-PRM-7B (He et al.,

2024b), ReasonEval-7B (Xia et al., 2025), Llemma-PRM800K-7B (Sun et al., 2024), Qwen2.5-Math-7B-PRM800K, Qwen2.5-Math-PRM-7B (Zhang et al., 2025b), and PathFinder-PRM-7B (Pala et al., 2025). These methods encompass reward models based on Monte Carlo estimation and PRMs trained with large-scale step-level supervision, enabling comparisons between generative evaluation and discriminative evaluation that directly assigns step-level rewards.

Generative PRMs. We select GenPRM (Zhao et al., 2026) and R-PRM (She et al., 2025) as the methods most directly comparable to CM-PRM. These models evaluate reasoning steps by generating natural-language analyses and correctness judgments. To reduce the influence of backbone differences, we reimplement GenPRM with Qwen2.5-Math-7B-Instruct, replacing its original DeepSeek-R1-Distill backbone.

Ablation Variants. To examine the roles of explicit analysis trajectories and training mechanisms, we consider the following variants:

- **SFT:** The model after supervised fine-tuning but before subsequent preference optimization. It generates analysis trajectories and final judgments, providing a baseline for measuring performance changes from further training.
- **Verdict:** A variant that outputs only correct/incorrect judgments without generating explicit analysis trajectories. It compares direct correctness prediction with analysis-based judgment, providing empirical evidence for the role of analysis trajectories in process evaluation.
- **DPO and IPO:** Variants trained from the SFT checkpoint using standard preference optimization objectives to assess how different optimization methods affect generative process evaluation.
- **DPO + Reg. and IPO + Reg.:** Variants that add quadratic regularization to the corresponding objectives to examine the effects of directly regularizing standard preference optimization.
- **Pearson:** A variant using the Pearson-χ^2 contrastive objective that retains its inherent quadratic term for rejected samples but excludes additional controlled-margin regularization, examining the effect of gradient-coefficient decoupling alone.
- **Pearson+Reg$^+$ and Pearson+Reg$^-$:** These one-sided variants isolate the two additional quadratic penalties in the CM objective. Pearson+Reg$^+$ uses $(\beta, \gamma) = (30, 0.5)$, adding chosen-side regularization while retaining Pearson's inherent rejected-side quadratic term. Pearson+Reg$^-$ uses $(\beta, \gamma) = (0, 60)$, adding only the additional rejected-side regularization. The nonzero regularization coefficients are inherited from the full CM-PRM rather than tuned separately.

DPO, IPO, their regularized counterparts, Pearson, and the two one-sided Pearson variants are initialized from the same SFT checkpoint as CM-PRM. Except for their objectives and objective-specific hyperparameters, these variants use the same training data, common training configuration, validation-based checkpoint selection, and training budget as CM-PRM.

DPO+Reg. and IPO+Reg. augment their respective base losses with quadratic regularization:

$$\mathcal{L}_{m+\mathrm{Reg.}} = \mathcal{L}_m + \frac{\beta\tau^2}{2} h_w^2 + \frac{\gamma\tau^2}{2} h_l^2, \qquad m \in \{\mathrm{DPO}, \mathrm{IPO}\}.$$

For DPO and IPO, we search the objective-specific hyperparameter β over $\{0.05, 0.1, 0.5\}$. Using the same validation set and F1-based selection criterion described above, we select $\beta = 0.1$ for both methods. We then fix this parameter and tune the quadratic regularization strength separately for DPO+Reg. and IPO+Reg. To limit the search budget, we fix $\beta/\gamma = 2$ and consider$(\beta, \gamma) \in \{(1.5, 3), (3, 6), (6, 12), (12, 24)\}$. The selected configurations are $(\beta, \gamma) = (3, 6)$ for DPO+Reg. and $(6, 12)$ for IPO+Reg. All hyperparameters and checkpoints are selected using the validation set; the test sets are not used for model selection.

We checked the Verdict-only variant's output format, verdict parsing, initialization and training transition, label distribution, and evaluation thresholds, and found no anomalies. The observed degradation is specific to our experimental setup.

F.3 Benchmarks for Static Evaluation and PRM-Guided Search

For static process evaluation, we use PRMBench Song et al. (2025) and ProcessBench Zheng et al. (2025):

• **ProcessBench** Zheng et al. (2025) contains 3,400 samples and evaluates whether a PRM can identify the first erroneous step in mathematical reasoning solutions. This benchmark directly tests error-localization ability in step-by-step reasoning.

• **PRMBench** Song et al. (2025) contains 6,216 samples and 83,456 step-level labels. It evaluates PRMs along three dimensions: *Simplicity*, *Soundness*, and *Sensitivity*, which test whether a PRM can detect redundant reasoning, make self-consistent judgments, and remain robust to contextual perturbations.

For PRM-guided search evaluation, we evaluate on *AIME24*, *AMC23*[1], *MATH* Hendrycks et al. (2021), *OlympiadBench* He et al. (2024a), *College Math* Tang et al. (2024), and *Minerva Math* Uesato et al. (2022). Following Pala et al. (2025); She et al. (2025), we use their subset of 200 test samples for OlympiadBench and College Math.

G Extented Experiments

G.1 ProcessBench Experiment

To assess CM-PRM's error-localization ability, we evaluate its performance in identifying the first erroneous step on ProcessBench. As shown in Table 4, CM-PRM achieves an average F1 score of 71.3, outperforming all compared generative PRMs while maintaining strong performance across the four subsets. Among discriminative PRMs, CM-PRM outperforms all listed models except Qwen2.5-Math-PRM-7B, trailing it by only 2.2 points. Meanwhile, CM-PRM uses 289K SFT samples and 128K preference samples, compared with approximately 1.8M step-level supervision samples used by the latter. These results demonstrate that CM-PRM achieves competitive process-level discrimination and error localization under limited supervision.

The ablation results further highlight the importance of the training design. SFT, DPO, and IPO achieve average F1 scores of 63.9, 66.9, and 61.1, respectively, all below CM-PRM, indicating that standard preference optimization does not necessarily improve process-level discrimination. Pearson-χ^2-CL without additional controlled-margin regularization achieves only 34.0 and exhibits a pronounced imbalance between the two classes: its scores on correct solutions approach or reach 100 across all four subsets, whereas its error-localization scores range from only 10.4 to 26.0. This suggests that gradient-coefficient decoupling alone may still leave the model biased toward accepting reasoning steps, compromising effective error detection.

Adding only chosen-side or rejected-side regularization to Pearson-χ^2-CL raises the average F1 to 64.0 and 63.8, respectively. Both variants substantially improve error localization compared with unregularized Pearson, but remain below CM-PRM (71.3). By comparison, quadratic regularization reduces the average F1 of DPO and IPO from 66.9 and 61.1 to 64.2 and 43.2, respectively. These results suggest that the effect of regularization depends on the underlying objective and support the joint control of chosen and rejected log-ratios within CM-PRM under our evaluated settings.

Furthermore, the Verdict variant, which outputs only correctness judgments, achieves an average F1 of 53.9, below the SFT baseline (63.9) and CM-PRM (71.3), both of which retain analysis trajectories. This gap persists across all four subsets and is consistent with the observations on PRMBench and guided search, supporting the role of explicit analysis under our experimental setup. Meanwhile, CM-PRM outperforms SFT, DPO, and IPO, which also retain analyses, indicating that analysis trajectories alone do not explain its full performance gains. Taken together, these ablations suggest that reliable error localization benefits from jointly designing analysis trajectories and training objectives to balance error detection with the recognition of valid reasoning.

[1]AIME24 and AMC23 Dataset Link: https://huggingface.co/datasets/AI-MO

Table 4: Detailed ProcessBench results. ★ marks our **CM-PRM**; † and ‡ denote results from She et al. (2025) and Pala et al. (2025), respectively. Bold/underline indicate the best/second-best PRM results. * and ** indicate significant differences against Vanilla DPO baseline with $p < 0.05$ and $p < 0.01$.

MODEL	# Samples	GSM8K			MATH			OLYMPIADBENCH			OMNIMATH			Avg. F1
		error	correct	F1	error	correct	F1	error	correct	F1	error	correct	F1	
LLM-as-judge, Proprietary language models														
GPT-4o†	-	70.0	91.2	79.2	54.4	76.6	63.6	45.8	58.4	51.4	45.2	65.6	53.5	61.9
o1-mini†	-	88.9	97.9	93.2	83.5	95.1	88.9	80.2	95.6	87.2	74.8	91.7	82.4	87.9
LLM-as-judge, Open-source language models														
Llama-3.3-70B-Instruct†	-	72.5	96.9	82.9	43.3	83.2	59.4	31.0	94.1	46.7	28.2	90.5	43.0	58.0
Qwen2.5-Math-72B-Instruct†	-	49.8	96.9	65.8	36.0	94.3	52.1	19.5	97.3	32.5	19.0	96.3	31.7	45.5
Qwen2.5-72B-Instruct†	-	62.8	96.9	76.2	46.3	93.1	61.8	38.7	92.6	54.6	36.6	90.9	52.2	61.2
Discriminative PRMs														
Math-Shepherd-7B†	445K	32.4	91.7	47.9	18.0	82.0	29.5	15.0	71.1	24.8	14.2	73.0	23.8	31.5
Math-PSA-7B†	860K	48.3	88.1	62.4	29.5	72.7	41.9	20.7	65.8	31.5	15.4	68.9	25.2	40.3
RLHFlow-Mistral-8B†	273K	33.8	**99.0**	50.4	21.7	72.2	33.4	8.2	43.1	13.8	9.6	45.2	15.8	28.4
RLHFlow-DeepSeek-8B†	250K	24.2	98.4	38.8	21.4	80.0	33.8	10.1	51.0	16.9	10.9	51.9	16.9	26.6
Llemma-PRM800K-7B†	350K	36.7	71.0	48.4	39.2	47.8	43.1	33.1	25.1	28.5	35.4	31.5	33.4	38.4
Skywork-PRM-7B†	unk	61.8	82.9	70.8	43.8	62.2	53.6	17.9	31.9	22.9	14.0	41.9	21.0	42.1
ReasonEval-7B†	350K	26.1	95.3	41.0	35.7	77.6	48.9	27.5	55.2	36.7	27.0	60.6	37.4	41.0
Qwen2.5-Math-7B-PRM800K†	264K	53.1	95.3	68.2	48.0	90.1	62.6	35.7	87.3	50.7	29.8	86.1	44.3	58.5
Qwen2.5-Math-PRM-7B†	∼1.8M	72.0	96.4	**82.4**	68.0	90.4	77.6	55.7	85.5	**67.5**	55.2	83.0	**66.3**	**73.5**
PathFinder-PRM-7B‡	400K	–	–	77.9	–	–	75.3	–	–	65.0	–	–	59.7	69.5
Generative PRMs														
GenPRM-7B	22.3K	69.8	83.2	75.9	66.8	59.2	62.7	53.4	32.8	40.6	**57.0**	45.4	50.5	57.4
R-PRM-7B-SFT†	289K	66.2	92.7	77.2	60.3	88.2	71.6	48.6	77.3	59.6	40.1	75.5	52.3	65.2
R-PRM-7B-DPO†	558K	72.0	91.7	80.7	71.2	83.5	76.9	**60.2**	67.8	63.8	55.5	65.6	60.1	70.4
Vanilla Ablation Baselines														
Backbone w/ SFT	289K	66.1	88.6	75.7	59.2	84.1	69.5	51.2	65.6	57.5	54.8	51.2	52.9	63.9
Backbone w/ SFT + Verdict	417K	62.8	79.8	70.3	58.6	61.8	60.2	45.4	38.6	41.7	44.8	42.2	43.5	53.9
Backbone w/ SFT + DPO	417K	69.5	96.0	80.6	55.4	92.8	69.4	56.8	66.4	61.2	46.8	70.4	56.2	66.9
Backbone w/ SFT + DPO + Reg.	417K	69.0	92.0	78.9	53.6	86.8	66.3	49.2	78.8	60.6	39.2	73.6	51.2	64.2
Backbone w/ SFT + IPO	417K	62.3	95.8	75.5	46.0	99.0	62.8	37.0	88.0	52.1	41.0	79.0	54.0	61.1
Backbone w/ SFT + IPO + Reg.	417K	28.5	99.0	44.3	36.0	97.8	52.6	28.8	94.0	44.1	19.2	96.0	32.0	43.2
Backbone w/ SFT + Pearson	417K	24.0	**99.0**	38.6	26.0	**100.0**	41.2	23.0	**98.0**	37.3	10.4	**96.6**	18.8	34.0
Backbone w/ SFT + Pearson + Reg^{+}	417K	67.5	93.2	78.3	53.0	91.2	67.0	46.8	77.4	58.3	40.6	74.0	52.4	64.0
Backbone w/ SFT + Pearson + Reg^{-}	417K	71.0	91.7	80.0	52.6	84.0	64.7	48.8	75.6	59.3	39.0	74.4	51.2	63.8
Our Model														
★ **CM-PRM**	417K	**75.7****	89.1*	81.9*	**73.2****	84.1*	**78.3****	56.4*	75.6*	64.3*	50.9*	75.0*	60.6*	71.3**

G.2 Effect of the Hyperparameter τ

To examine how τ affects reward dynamics and discriminative performance under controlled-margin optimization, we compare $\tau \in \{0.5, 1.0, 2.0\}$, tracking chosen and rejected rewards, reward margins, and accuracy while keeping β, γ, and the other training settings fixed.

As shown in the Figure 11, smaller τ values yield larger reward margins in later training stages. With $\tau = 0.5$, the chosen reward initially declines before increasing substantially, while the rejected reward decreases to a lower level; the final accuracy reaches approximately 0.60. With $\tau = 2.0$, the chosen reward begins increasing earlier, but subsequent reward changes and margin growth are more limited, resulting in a final accuracy of approximately 0.50. The results for $\tau = 1.0$ fall between these two settings. These observations indicate that an earlier increase in the chosen reward does not necessarily translate into better final discriminative performance and should be interpreted alongside subsequent reward dynamics.

This behavior can be understood through the dual role of τ in the objective: the linear terms scale with τ, whereas the quadratic regularization terms scale with τ^2. Increasing τ therefore not only strengthens the update signals but also increases the relative weight of the quadratic terms and brings the target log-ratios, $h_w^* = 1/(\beta\tau)$ and $h_l^* = -1/(\gamma\tau)$, closer to zero. This is consistent with the more constrained late-stage reward dynamics observed at larger τ. Conversely, smaller τ permits a larger target margin but may also be accompanied by a temporary decline in the chosen reward.

Overall, τ jointly controls update strength and the target margin, requiring a balance between reward constraints and effective learning rather than favoring larger values unconditionally. Within the tested

range, $\tau = 0.5$ achieves the highest final accuracy. However, its early decline in the chosen reward also highlights the importance of assessing final performance alongside the optimization trajectory.

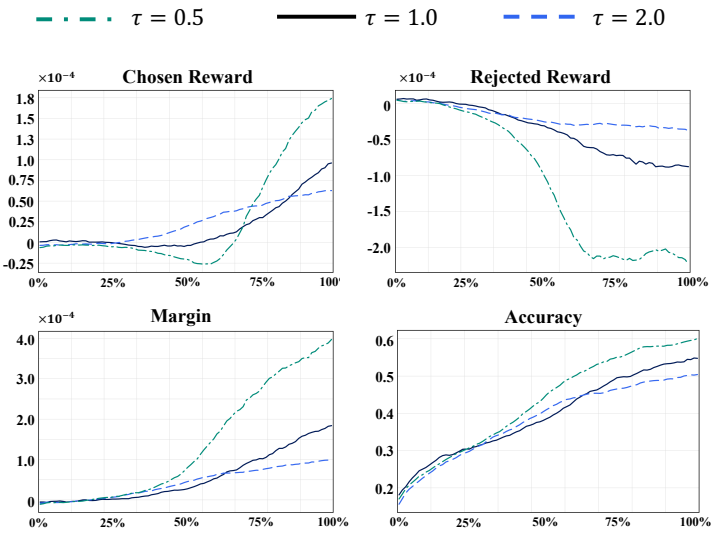


Figure 11: Training Dynamics under Different τ Values.

G.3 Data Efficiency and Training Progression

To examine whether CM-PRM effectively utilizes limited preference supervision, we compare the PRMBench F1 scores and reported training data sizes of different PRMs. We also track CM-PRM's performance from SFT initialization through subsequent training stages to assess supervision efficiency and the benefits of continued optimization.

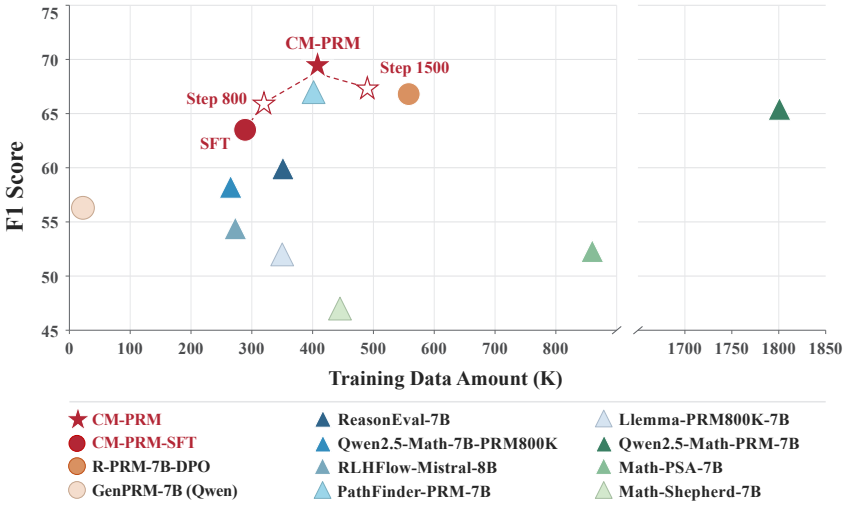


Figure 12: Data efficiency comparison on PRMBench. The x-axis denotes the reported training data scale, and the y-axis denotes the avg. F1.

As shown in the Figure 12, CM-PRM achieves an F1 score of 69.1 using 289K SFT samples and 128K preference samples, outperforming Qwen2.5-Math-PRM-7B, which uses approximately 1.8M step-level samples, by 3.6 points. It also outperforms R-PRM-7B-DPO, which uses more training samples. Across training stages, performance improves from 63.5 after SFT to approximately 66.0 at Step 800, subsequently reaches 69.1, and then declines to approximately 67.3 with continued training to Step 1500 (additionally extended training steps). These observations indicate that performance

gains do not simply follow from increasing the amount of training data or the number of optimization steps.

Together with the preceding ablation results, the early performance gains support the effectiveness of controlled-margin optimization in leveraging preference supervision to enhance process-level discrimination. The subsequent decline indicates that the benefits of continued optimization are not monotonic, motivating checkpoint selection based on validation performance. Overall, CM-PRM achieves stronger process-evaluation performance with fewer supervision samples, highlighting that effectively utilizing existing supervision is as important as increasing its quantity. The results across training stages further show that effectively learning from preference signals does not necessarily require prolonged training.