

BPO: REVISITING PREFERENCE MODELING IN DIRECT PREFERENCE OPTIMIZATION

Anonymous authors

Paper under double-blind review

ABSTRACT

Direct Preference Optimization (DPO) have emerged as a popular method for aligning LLMs with human preferences. While DPO effectively preserves the relative ordering between chosen and rejected responses through pairwise ranking losses, it often neglects absolute reward magnitudes. This oversight can decrease the likelihood of chosen responses and increase the risk of generating out-of-distribution responses, leading to poor performance. We term this issue *Degraded Chosen Responses (DCR)*. To address this issue, we propose **Balanced Preference Optimization (BPO)**, a novel framework that dynamically balances the optimization of chosen and rejected responses through two key components: *balanced reward margin* and *gap adaptor*. Unlike previous methods, BPO can fundamentally resolve DPO’s DCR issue, without introducing additional constraints to the loss function. Experimental results on multiple mathematical reasoning tasks show that BPO significantly outperforms DPO, improving accuracy by **+10.1%** with Llama-3.1-8B-Instruct (18.8% \rightarrow 28.9%) and **+11.7%** with Qwen2.5-Math-7B (35.0% \rightarrow 46.7%). It also surpasses DPO variants by **+3.6%** over IPO (43.1%), **+5.0%** over SLiC (41.7%), and **+3.1%** over Cal-DPO (43.6%) on the same model. Remarkably, our algorithm requires only *a single line of code modification*, making it simple to implement and fully compatible with existing DPO-based frameworks.

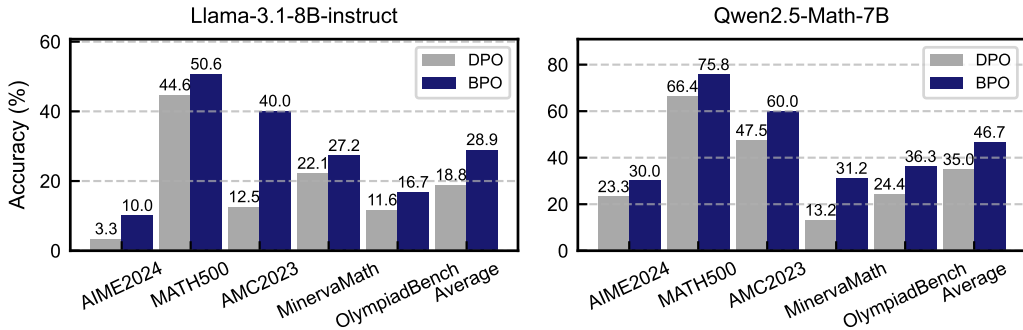


Figure 1: Overall performance across five competition-level benchmarks (AIME2024, MATH500, AMC2023, MinervaMath, and OlympiadBench). **BPO** achieves an average score of **28.9%** using Llama-3.1-8B-Instruct policy generator, and **46.7%** with Qwen2.5-Math-7B. This represents a substantial improvement over DPO, yielding average gains of **+10.1%** and **+11.7%**, respectively.

1 INTRODUCTION

Aligning LLMs with human preferences is essential to ensure their responses are safe, helpful, and aligned with user intent [Bai et al. \(2022\)](#); [Ouyang et al. \(2022\)](#); [Stiennon et al. \(2020\)](#). While Reinforcement Learning from Human Feedback (RLHF) [Ouyang et al. \(2022\)](#); [Christiano et al. \(2017\)](#) has become a standard approach for aligning models with preferences, it suffers from training instability and complexity. Recent direct preference optimization methods [Xu et al. \(2024a\)](#); [Ethayarajh](#)

et al. (2024); Azar et al. (2024); Liu et al. (2024) offer a simpler alternative by replacing RLHF with supervised learning on preference data. These methods avoid explicit reward modeling by using policy likelihood to define an implicit reward, achieving both high efficiency and competitive performance.

However, direct preference optimization methods suffer from a critical flaw: *the likelihood of chosen responses often decreases alongside that of rejected responses*, as is shown in Figure 2. We refer to this issue as Degraded Chosen Responses (**DCR**). An undesirable consequence of DCR issue is that the learned policy tends to increase the probability of unknown out-of-distribution (OOD) responses, leading to degraded performance Xiao et al. (2024).

We analyze the main reason for DCR issue is the mismatch in task difficulty: The shared preference modeling of direct preference optimization methods is to maximize the expected relative difference between the implicit rewards of chosen and rejected responses. This objective can be broken down into two tasks, increasing the probability of chosen responses and decreasing the probability of rejected ones. However, reducing the probability of rejected responses is far easier than increasing that of chosen responses, as lowering the likelihood of a rejected response only requires boosting arbitrary alternative tokens, whereas increasing the probability of a specific response, the model must deeply understand the problem and identify relevant patterns to boost the likelihood of very specific tokens, which is an inherently harder task.

To mitigate the DCR issue, several approaches such as DPOP Pal et al. (2024) and Cal-DPO Xiao et al. (2024) have been proposed. These methods aim to preserve the likelihood of chosen responses by introducing additional constraints into the loss function, such as enforcing a minimum probability threshold or regularizing the reward magnitude. While these modifications can help prevent the degradation of chosen response probabilities, they often come at a cost. The added constraints may inadvertently bias the optimization process, leading to suboptimal generalization and reduced model robustness. This lead us to the following question:

How can we fundamentally address the DCR issue?

Our answer to this question is **BPO**, a simple yet effective framework that addresses the DCR issue by explicitly balancing the optimization of chosen and rejected responses. The key intuition behind BPO is quite simple: instead of only maximizing the relative reward gap between chosen and rejected responses, we should also ensure that the absolute chosen reward is preserved. This can be achieved through a simple modification to existing methods. For instance, BPO can be implemented on top of DPO with just a single line of code, by replacing the relative reward margin term with a balanced reward margin. Moreover, BPO can easily generalize to other preference optimization functions (see Section 3.3).

We summarize our contributions as follows:

- We propose BPO, a novel framework which effectively mitigates the DCR issue by explicitly preserving the absolute likelihood of chosen responses while still maximizing the reward gap between chosen and rejected responses.
- Our method is simple, general, and easy to implement, requiring only a minor modification to existing preference optimization algorithms.
- Through extensive experiments and ablation studies, we demonstrate that BPO consistently outperforms DPO and its variants. We present a new perspective on reparametrizing the reward margin, which significantly enhances training stability and overall performance.

2 METHODOLOGY

2.1 BALANCED PREFERENCE OPTIMIZATION

Problem Setup. Let $\mathbf{x} = [x_1, x_2, \dots]$ denote an input sequence, and let $\mathbf{y}_w = [y_1, y_2, \dots]$ and $\mathbf{y}_l = [y_1, y_2, \dots]$ be two responses sampled from a reference language model $\pi_{\text{ref}}(\mathbf{y}|\mathbf{x})$. These response pairs are then presented to human or model-based annotators, which provide preference labels of the form $\mathbf{y}_w \succ \mathbf{y}_l \mid \mathbf{x}$, indicating that \mathbf{y}_w is preferred over \mathbf{y}_l given the input \mathbf{x} . The preference distribution is typically modeled using a latent reward function, as follows:

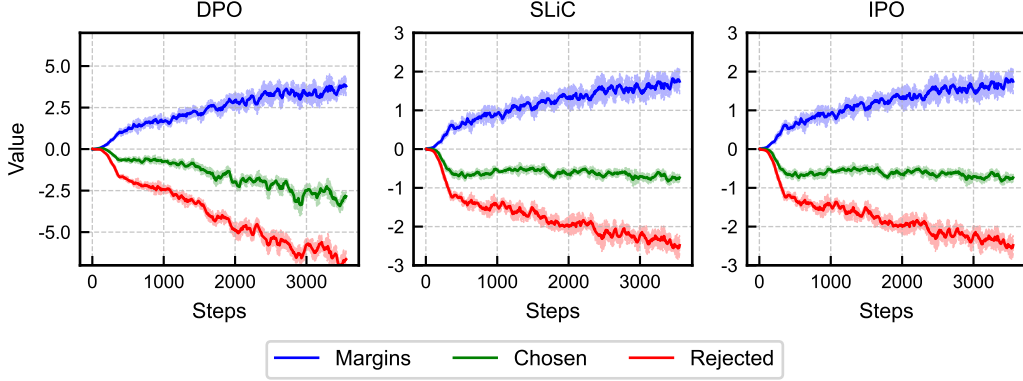


Figure 2: In DPO, the rewards for chosen responses can drop below zero, whereas in our BPO, they remain positive and continue to increase. A smaller gap adaptor α reduces the penalty on rejected responses, while a larger α shifts the focus toward improving chosen responses, resulting in more balanced and effective updates.

$$p(\mathbf{y}_w \succ \mathbf{y}_l | \mathbf{x}) = f \left(\beta \log \frac{\pi_\theta(\mathbf{y}_w | \mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_w | \mathbf{x})} - \beta \log \frac{\pi_\theta(\mathbf{y}_l | \mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_l | \mathbf{x})} \right). \quad (1)$$

where $f : \mathbb{R} \rightarrow [0, 1]$ is preference optimization function, which converts reward differences into winning probabilities. When f is the logistic log function, we get the Bradley-Terry (BT) preference model [41]. Given a dataset $\mathcal{D} = \{(\mathbf{x}^{(i)}, \mathbf{y}_w^{(i)}, \mathbf{y}_l^{(i)})\}_{i=1}^N$ of human preferences, the learning objective is to optimize the policy π_θ such that it aligns with the preference distribution while maintaining a controlled divergence from π_{ref} .

Balanced Reward Margin. Standard direct preference optimization methods optimize the relative reward margin defined as:

$$\rho_\theta^d = r_w - r_l. \quad (2)$$

Where, $r_w = \beta \log \pi_\theta(\mathbf{y}_w | \mathbf{x}) - \beta \log \pi_{\text{ref}}(\mathbf{y}_w | \mathbf{x})$, $r_l = \beta \log \pi_\theta(\mathbf{y}_l | \mathbf{x}) - \beta \log \pi_{\text{ref}}(\mathbf{y}_l | \mathbf{x})$. While maximizing ρ_θ^d encourages $\pi_\theta(\mathbf{y}_w | \mathbf{x}) \gg \pi_\theta(\mathbf{y}_l | \mathbf{x})$, it disregards the absolute magnitudes of r_w and r_l . This permits two failure modes: (1) Degraded Chosen Responses: $\pi_\theta(\mathbf{y}_w | \mathbf{x})$ can diminish as long as $\pi_\theta(\mathbf{y}_l | \mathbf{x})$ degrades faster, since ρ_θ^d depends only on their relative difference. (2) Overestimated Rejected Responses: Insufficient suppression of $\pi_\theta(\mathbf{y}_l | \mathbf{x})$ (small $|r_l|$) may lead to low-quality or out-of-distribution \mathbf{y}_l . To address these issues, we propose a balanced reward margin:

$$\rho_\theta^b = \min(r_w, -r_l). \quad (3)$$

It dynamically prioritizes the weaker component of the reward pair: (1) when $r_w \leq -r_l$, focuses on improving r_w , ensuring $\pi_\theta(\mathbf{y}_w | \mathbf{x})$ aligns with high-quality responses. (2) when $-r_l \leq r_w$, prioritizes suppressing $\pi_\theta(\mathbf{y}_l | \mathbf{x})$, mitigating overconfidence in undesired outputs. By optimizing ρ_θ^b , the policy maintains a balance between reinforcing chosen responses and penalizing rejected ones, preventing pathological optimization trajectories inherent to standard DPO.

Gap Adaptor. To provide finer control over the balance, we introduce a gap adaptor $\alpha \in (0, 1]$ that controls the relative gap between r_w and r_l . The balanced reward margin becomes:

$$\rho_\theta^b = \min(r_w, -\alpha r_l). \quad (4)$$

The gap adaptor α allows us to adjust the relative importance of suppressing rejected responses compared to improving chosen responses. A smaller α decreases the penalty on rejected responses,

while a larger α places more emphasis on improving chosen responses, resulting in more balanced updates, as illustrated in Figure 5.

Final Objective. Incorporating the balanced reward margin and the gap adaptor, the final loss function of BPO is defined as:

$$\mathcal{L}(\theta) = -\mathbb{E}_{(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l) \sim \mathcal{D}} \left[f \left(\min \left(\beta \log \frac{\pi_{\theta}(\mathbf{y}_w | \mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_w | \mathbf{x})}, -\alpha \beta \log \frac{\pi_{\theta}(\mathbf{y}_l | \mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_l | \mathbf{x})} \right) \right) \right]. \quad (5)$$

By optimizing this loss, we ensure that the policy does not overfit to either the chosen or rejected responses, thereby maintaining a balanced and robust alignment with human preferences.

2.2 THEORETICAL ANALYSIS

Gradient analysis. When employing logistic log as preference optimization function, which is same with standard DPO, the gradient of the BDO loss is given by (derived from the Appendix A.1):

$$\nabla_{\theta} \mathcal{L}(\theta) = -\mathbb{E}_{(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l) \sim \mathcal{D}} \begin{cases} \sigma(-\beta r_w) \cdot \beta \cdot \nabla_{\theta} \log \pi_{\theta}(\mathbf{y}_w | \mathbf{x}), & \text{if } r_w < -\alpha r_l, \\ \sigma(\alpha \beta r_l) \cdot (-\alpha \beta) \cdot \nabla_{\theta} \log \pi_{\theta}(\mathbf{y}_l | \mathbf{x}), & \text{otherwise.} \end{cases} \quad (6)$$

In contrast, the gradient for DPO is:

$$\nabla_{\theta} \mathcal{L}_{\text{DPO}} = -\beta \mathbb{E}_{(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l) \sim \mathcal{D}} [\sigma(-\beta(r_w - r_l)) \cdot (\nabla_{\theta} \log \pi_{\theta}(\mathbf{y}_w | \mathbf{x}) - \nabla_{\theta} \log \pi_{\theta}(\mathbf{y}_l | \mathbf{x}))]. \quad (7)$$

From this comparison, we can highlight several key advantages of BPO:

(1) **Dynamic Balancing of Chosen and Rejected Response Updates.** In DPO, gradient updates are weighted based on the relative log-probabilities of chosen and rejected responses. This approach can lead to an overemphasis on rejected responses due to mismatch in task difficulty, potentially neglecting updates from chosen ones. In contrast, BPO introduces a threshold parameter α to dynamically adjust the contribution of each response type. When $r_w < -\alpha r_l$, only the chosen response influences the gradient. Otherwise, the model prioritizes updating the rejected response via $-\alpha \beta \nabla_{\theta} \log \pi_{\theta}(\mathbf{y}_l | \mathbf{x})$, it ensures balanced learning.

(2) **Accelerated Convergence.** We visualize the gradient distributions of the loss for both the relative reward margin $r_w - r_l$ and the balanced reward margin $\min(r_w, -\alpha r_l)$. The visualization reveals that when using the relative reward margin, more than half of the gradients in the lower-right region have very low values. This indicates that when the model parameters converge to this area, the updates will be exceedingly slow. In contrast, when using the balanced reward margin, the region with low gradient values is significantly reduced. As a result, the model can benefit from more substantial updates, thereby accelerating the convergence.

(3) **Reduced Computational Overhead.** DPO requires computing gradients for both chosen and rejected responses, along with their probability ratios, which can be computationally intensive, especially on large datasets. BPO reduces computational overhead by computing gradients only for the response under strong preference conditions ($\beta r_w < -\alpha \beta r_l$). This effectively halves the computational cost.

Theorem 1. Let γ be the maximized margin in the balanced reward margin constraint $\rho_{\theta}^b \geq \gamma$. Under BPO, the likelihood of the chosen response satisfies:

$$\pi_{\theta}(\mathbf{y}_w | \mathbf{x}) \geq \exp \left(\frac{\gamma}{\beta} \right) \pi_{\text{ref}}(\mathbf{y}_w | \mathbf{x}). \quad (8)$$

Proof provided in Appendix A.2, this inequality ensures that the learned policy π_{θ} assigns a probability to the chosen response \mathbf{y}_w that is at least exponentially greater, scaled by γ/β than that assigned by the reference policy π_{ref} . Importantly, this constraint prevents the degradation of $\pi_{\theta}(\mathbf{y}_w | \mathbf{x})$, as the likelihood ratio cannot fall below $\exp(\gamma/\beta)$, even if $\pi_{\theta}(\mathbf{y}_l | \mathbf{x})$ is further suppressed.

In contrast, DPO optimizes the relative margin $r_w - r_l$, which allows $\pi_\theta(\mathbf{y}_w | \mathbf{x})$ to decrease as long as $\pi_\theta(\mathbf{y}_l | \mathbf{x})$ degrades faster. This can lead to a collapse in the probability of high-quality chosen responses.

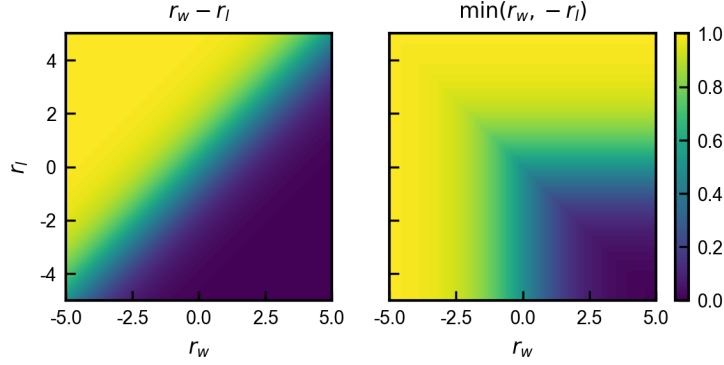


Figure 3: The gradient distributions under the logistic log preference optimization function are compared for both the relative reward margin $r_w - r_l$ and the balanced reward margin $\min(r_w, -r_l)$. Notably, the region with low gradient values is significantly reduced when using the balanced reward margin.

3 EXPERIMENT

3.1 EXPERIMENT SETUP

Training Dataset. Our experiments were performed using the publicly available Open Reasoner Zero 57k dataset [Hu et al. \(2025\)](#), which consists of mathematical problems sourced from a variety of collections, such as AIME, MATH, Numina-Math Collection, Tulu3 MATH, OpenR1-Math-220k, and other open-source datasets.

Evaluation To comprehensively evaluate our model’s reasoning capabilities, we conduct experiments on a diverse range of mathematical reasoning benchmarks, including AIME2024, MATH500 [Hendrycks et al. \(2021\)](#), AMC2013 [Lewkowycz et al. \(2022\)](#), MinervaMath [Lewkowycz et al. \(2022\)](#) and OlympiadBench [He et al. \(2024\)](#). We report the average accuracy across these five datasets as our primary evaluation metric. For each dataset, we compute the pass@1 score. To further assess the generalizability of our approach, we test BPO using two model families: the Llama series [Dubey et al. \(2024a\)](#) and the Qwen2.5 series [Yang et al. \(2024\)](#). Specifically, we use Llama-3.1-8B-Instruct, along with three models from the Qwen2.5 series: Qwen2.5-Math-1.5B, Qwen2.5-Math-7B, and Qwen2.5-Math-7B-Instruct.

Baselines. We compare BPO with several strong baselines. Specifically, we contrast BPO against GPT-4o [Hurst et al. \(2024\)](#), as well as open-source instruction-tuned models such as Llama-3.1-70B-Instruct and Qwen2.5-Math-7B-Instruct. Additionally, we compare BPO with fine-tuned models including Qwen2.5-Math-7B-Base-SFT [Dubey et al. \(2024b\)](#), which uses Supervised Fine-Tuning (SFT), and Qwen2.5-7B-RAFT-Zero [Dubey et al. \(2024b\)](#), which employs Reward-Ranked Fine-Tuning. Furthermore, we benchmark BPO against several preference optimization methods: DPO [Rafailov et al. \(2023\)](#), IPO [Azar et al. \(2024\)](#), SLiC [Liu et al. \(2024\)](#), Cal-DPO [Xiao et al. \(2024\)](#), and DPOP [Pal et al. \(2024\)](#).

Implementation Details. Our base models is Qwen2.5-Math-7B-Base, following prior work [Dubey et al. \(2024b\)](#), we sample 8 responses per prompt and rank them based on the correctness of the final outcome. We then apply the max-min strategy to select preference pairs, specifically the response with the highest reward is paired with the one with the lowest reward. If all responses receive the same reward, we discard the prompt and proceed to the next one. We train the model with a learning rate of 5e-6, a batch size of 8, for 2 epochs, using a maximum sequence length of 2048. The experiments are run on 2 Nvidia A40 GPUs with BF16 precision.

3.2 OVERALL PERFORMANCE

BPO achieves superior performance, outperforming DPO and its variants. As shown in Table 1, the proposed method, BPO, demonstrates superior performance across five competition-level mathematical reasoning benchmarks, achieving an average accuracy of 46.7%, significantly outperforming DPO and its variants. Notably, BPO excels on the challenging AIME 2024 benchmark with 30.0% accuracy, showcasing its effectiveness in complex reasoning tasks. Compared to base models and those fine-tuned via Supervised Fine-Tuning, BPO shows clear advantages, highlighting the benefits of preference-based training. Importantly, BPO achieves these improvements through minimal modifications to existing frameworks, making it a practical and effective enhancement for large language models.

3.3 ABLATION STUDY

BPO consistently outperforms DPO across all model architectures and scales. The results in Table 2 demonstrate that BPO improves performance across different LLM architectures and scales. Specifically, when applied to diverse model architectures, BPO significantly boosts average accuracy on Llama-3.1-8B-Instruct (from 18.8% to 28.9%) and Qwen2.5-Math-7B-Instruct (from 42.8% to 48.8%). In terms of model scale, BPO consistently enhances performance across parameter sizes: from 28.2% to 38.3% on the smaller Qwen2.5-Math-1.5B-Base, and from 41.0% to 46.7% on the larger Qwen2.5-Math-7B-Base. Furthermore, BPO not only surpasses the closed-source model GPT-4o, but also outperforms instruction-tuned models such as Qwen2.5-Math-7B-Instruct and fine-tuned variants including Qwen2.5-Math-7B-Base-SFT and Qwen2.5-7B-RAFT-Zero. These results demonstrate that BPO achieves both effective alignment and strong generalization, confirming its robust performance across different LLM architectures and model scales.

Table 1: Overall performance across five competition-level math reasoning benchmarks. Results for BPO are shaded. **Avg.** indicates the mean accuracy across all datasets. The top results are shown in **bold**. The table demonstrates that BPO outperforms both standard DPO and its variants, achieving the highest average accuracy.

| Method (\downarrow) / Dataset (\rightarrow) | AIME2024 | MATH500 | AMC2023 | Minerva Math | Olympiad Bench | Avg. |
|---|----------|---------|---------|--------------|----------------|-------------|
| GPT-4o | 9.3 | 76.4 | 45.8 | 36.8 | 43.3 | 43.3 |
| Llama-3.1-70B-Instruct | 16.7 | 64.6 | 30.1 | 35.3 | 31.9 | 35.7 |
| Qwen2.5-Math-7B-Base | 23.3 | 66.4 | 47.5 | 13.2 | 24.4 | 35.0 |
| Qwen2.5-Math-7B-Base-SFT | 20.0 | 73.2 | 62.5 | 30.5 | 35.6 | 44.4 |
| Qwen2.5-Math-7B-Instruct | 13.3 | 79.8 | 50.6 | 34.6 | 40.7 | 43.8 |
| Qwen2.5-7B-RAFT-Zero | 20.0 | 77.6 | 55.0 | 30.5 | 38.7 | 44.4 |
| DPO | 6.7 | 71.2 | 55.0 | 39.3 | 32.9 | 41.0 |
| IPO | 10.0 | 75.6 | 52.5 | 39.7 | 37.6 | 43.1 |
| SLiC | 10.0 | 73.2 | 55.0 | 37.5 | 33.0 | 41.7 |
| Cal-DPO | 20.0 | 75.4 | 62.5 | 24.3 | 35.9 | 43.6 |
| DPOP | 23.3 | 77.0 | 57.5 | 30.9 | 35.9 | 44.9 |
| BPO (ours) | 30.0 | 75.8 | 60.0 | 31.2 | 36.3 | 46.7 |

Balanced reward margin is applicable to various preference optimization functions. We investigate the applicability of the balanced reward margin across various preference optimization functions introduced in Tang et al. (2024). As shown in Table 3, the balanced reward margin consistently outperforms the relative margin across all tested optimization objectives. Specifically, under the logistic log loss used in DPO, the balanced margin improves performance from 41.0% to 44.5% (+3.5%). Similarly, when applied with the hinge loss employed in SLiC, it achieves an improvement of +5.0%, reaching 46.7%. The balanced formulation also yields consistent gains of +1.9% and +1.0% under less commonly used losses such as truncated quadratic loss and Savage loss, respectively. Notably, as shown in Figure 4, the balanced reward margin demonstrates significant advantages on challenging datasets like AIME2024 and AMC2023. These results indicate that the balanced reward margin provides more effective alignment with human preferences compared to the

standard relative margin, highlighting its robustness and broad applicability across diverse preference learning settings.

Table 2: Performance comparison across different model architectures and scales, it shows that BPO consistently outperforms DPO across all configurations and datasets.

| Base Model | Method | AIME2024 | MATH500 | AMC2023 | Minerva Math | Olympiad Bench | Avg. |
|--------------------------|--------|-------------|-------------|-------------|--------------|----------------|-------------|
| Llama-3.1-8B-Instruct | DPO | 3.3 | 44.6 | 12.5 | 22.1 | 11.6 | 18.8 |
| | BPO | 10.0 | 50.6 | 40.0 | 27.2 | 16.7 | 28.9 |
| Qwen2.5-Math-1.5B-Base | DPO | 3.3 | 58.8 | 27.5 | 27.6 | 23.6 | 28.2 |
| | BPO | 16.7 | 64.8 | 52.5 | 26.8 | 30.5 | 38.3 |
| Qwen2.5-Math-7B-Base | DPO | 6.7 | 71.2 | 55.0 | 39.3 | 32.9 | 41.0 |
| | BPO | 30.0 | 75.8 | 60.0 | 31.2 | 36.3 | 46.7 |
| Qwen2.5-Math-7B-Instruct | DPO | 10.0 | 77.0 | 60.0 | 28.7 | 38.1 | 42.8 |
| | BPO | 20.0 | 82.4 | 60.0 | 40.8 | 40.6 | 48.8 |

Table 3: Performance comparison between the relative reward margin $x_1 - x_2$ and the balanced reward margin $\min(x_1, -x_2)$ under different loss functions. The proposed balanced reward margin shows consistent gains across various preference optimization objectives. Gap adaptor is set to 0.3 in this experiment.

| Loss Type | Algorithm | $f(\beta\rho_\theta)$ | $x_1 - x_2$ | $\min(x_1, -x_2)$ | Δ |
|--------------------------|-----------|-------------------------------------|-------------|-------------------|----------|
| logistic log loss | DPO | $\log(1 + \exp(-\beta\rho_\theta))$ | 41.0 | 44.5 | + 3.5 |
| hinge loss | SLiC | $\max(0, 1 - \beta\rho_\theta)$ | 41.7 | 46.7 | + 5.0 |
| squared loss | IPO | $(\beta\rho_\theta - 1)^2$ | 43.1 | 43.9 | + 0.8 |
| exponential loss | N/A | $\exp(-\beta\rho_\theta)$ | 43.5 | 43.9 | + 0.4 |
| truncated quadratic loss | N/A | $(\max(0, 1 - \beta\rho_\theta))^2$ | 42.4 | 44.3 | + 1.9 |
| savage loss | N/A | $1/(1 + \exp(\beta\rho_\theta))^2$ | 42.7 | 43.7 | + 1.0 |

Moderately relaxing the constraints on rejected responses can enhance model performance.

We analyzed the impact of the gap adaptor (α) parameter on model performance. As shown in Figure 5, BPO performance initially increases and then decreases as α increases. For the logistic log loss, the best performance is achieved when $\alpha = 0.5$. In contrast, for the hinge loss, the optimal performance is observed at $\alpha = 0.3$. These findings suggest that it is not necessary to strictly balance rewards of chosen responses and rejected responses. Instead, moderately relaxing the constraints on rejected responses can enhance model performance.

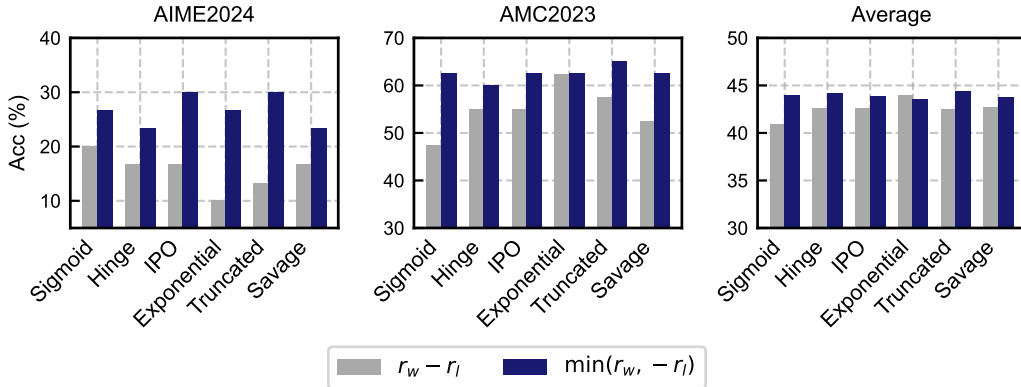


Figure 4: Relative Reward Margin vs. Balanced Reward Margin under different preference optimization functions. Balanced Reward Margin shows significant advantages on AIME2024 and AMC2023.

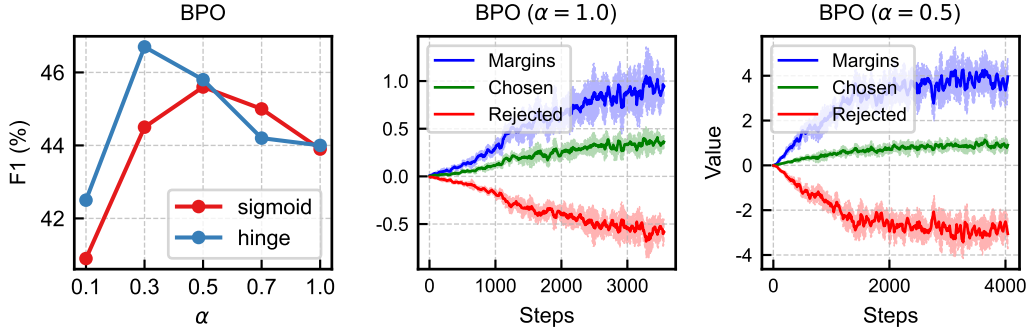


Figure 5: In DPO, the rewards for chosen responses can drop below zero, whereas in our BPO, they remain positive and continue to increase. A smaller gap adaptor α reduces the penalty on rejected responses, while a larger α shifts the focus toward improving chosen responses, resulting in more balanced and effective updates.

BPO performs better when addressing more difficult challenges. As shown in Figure 6, BPO demonstrates increasingly significant performance gains over both Base and DPO as problem difficulty escalates. While all methods perform comparably on easier tasks, BPO’s advantage becomes markedly pronounced starting from Level 3. Notably, at Level 5, which represents the most challenging tier, DPO’s performance even drops below that of the Base model. This suggests potential instability or overfitting when handling highly complex problems. In contrast, BPO maintains consistent and robust improvement, achieving a score 17.2% higher than Base and a substantial 26.1% advantage over DPO. This trend highlights BPO’s superior capacity to navigate complex reasoning landscapes, making it particularly well suited for tasks where difficulty and nuance demand more sophisticated optimization.

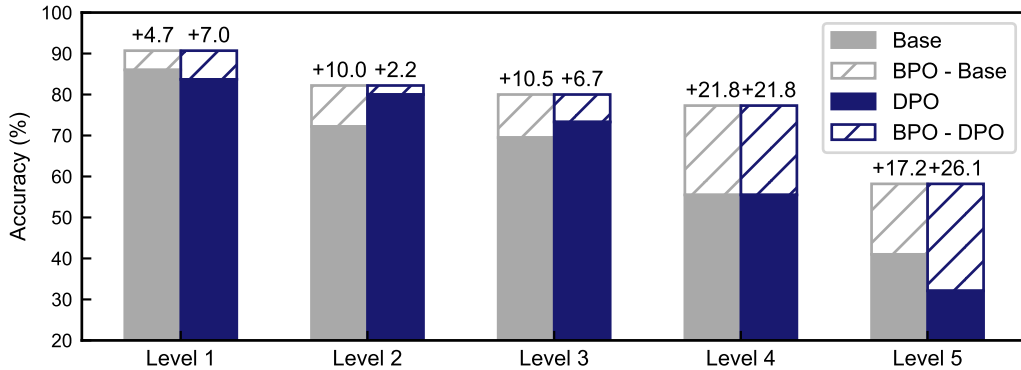


Figure 6: Performance comparison between Base, DPO and BPO on Math500 with varying difficulty levels. It shows that BPO achieves the best performance across all difficulty levels. The enhancement in performance becomes more evident when tackling more challenging problems. The BPO result is based on Qwen2.5-Math-7B with hinge loss function and α set to 0.3.

4 RELATED WORK

Direct Preference Optimization. Efficient preference optimization algorithms, such as DPO have emerged as a popular approach for aligning LLMs with human preferences. Compared to traditional methods like RLHF, DPO offers several advantages, including greater stability, strong performance, and improved computational efficiency. Despite these benefits, recent studies have identified several challenges associated with DPO. For instance, its implicit reward modeling can lead to biased policies that favor out-of-distribution responses [Xu et al. \(2024b\)](#); [Saeidi et al. \(2024\)](#). Additionally, offline DPO has been found to be empirically less effective than online alignment methods [Iverson](#)

et al. (2024), and aligned models may suffer from degrading performance after alignment Lin et al. (2024); Lu et al. (2024). In response to these limitations, various enhanced versions of DPO have been proposed, including CPO Xu et al. (2024a), KTO Ethayarajh et al. (2024), IPO Azar et al. (2024), SLiC Liu et al. (2024), Cal-DPO Xiao et al. (2024) and DPOP Pal et al. (2024). However, none of these approaches fundamentally address the DCR problem, which limits the potential performance improvements of DPO.

Bradley-Terry Model. The Bradley-Terry (BT) model Bradley & Terry (1952) was originally proposed to convert pairwise comparisons into numerical scores. With the development of reinforcement learning from human feedback (RLHF) Stiennon et al. (2020); Ouyang et al. (2022); Bai et al. (2022) the BT model has been widely used to optimize reward models, achieving significant success in improving the performance of LLMs across various tasks. Later, with the introduction of DPO, the BT model was further applied to model preferences directly for LLM preference tuning. However, several studies have discussed the limitations and challenges of using the BT model within the RLHF framework from different perspectives Azar et al. (2024); Tang et al. (2024); Zhao et al. (2023). For instance, it has been noted that the Bradley-Terry (BT) model cannot capture non-transitive preferences, and maximizing the corresponding Elo score may not align with the true objective of preference optimization Munos et al. (2024). Additionally, applying the BT model within DPO can result in problematic overfitting when the observed preferences are deterministic Azar et al. (2024). In our work, we revisit the foundational mechanisms of the BT model and propose improvements to address these limitations.

5 CONCLUSION AND LIMITATIONS

In this work, we identify a critical issue in direct preference optimization methods: the Degraded Chosen Responses (DCR) problem. To address this limitation, we propose BPO, a simple yet effective framework that explicitly preserves the absolute likelihood of chosen responses while still maximizing the reward gap between chosen and rejected responses. BPO can be seamlessly integrated into existing preference optimization algorithms with minimal modifications, offering both theoretical guarantees and practical benefits. We hope it inspires further research into more balanced and effective preference learning objectives. A limitation of BPO is that it is currently restricted to offline methods and does not incorporate on-policy learning, where the policy can interact with the reward model during training. It would be interesting to explore how the balanced reward margin approach used in BPO performs in an on-policy learning scenario. We consider this an exciting direction for future research.

REFERENCES

- Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Rémi Munos, Mark Rowland, Michal Valko, and Daniele Calandriello. A general theoretical paradigm to understand learning from human preferences. In Sanjoy Dasgupta, Stephan Mandt, and Yingzhen Li (eds.), *International Conference on Artificial Intelligence and Statistics, 2-4 May 2024, Palau de Congressos, Valencia, Spain*, volume 238 of *Proceedings of Machine Learning Research*, pp. 4447–4455. PMLR, 2024.
- Yuntao Bai, Andy Jones, Kamal Ndousse, and et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.
- Ralph A. Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952. doi: 10.1093/biomet/39.3-4.324.
- Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 4299–4307, 2017.
- Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, and et al. The llama 3 herd of models. *CoRR*, abs/2407.21783, 2024a. doi: 10.48550/ARXIV.2407.21783. URL <https://doi.org/10.48550/arXiv.2407.21783>.

- Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, and et al. The llama 3 herd of models. *CoRR*, abs/2407.21783, 2024b. doi: 10.48550/ARXIV.2407.21783. URL <https://doi.org/10.48550/arXiv.2407.21783>.
- Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Model alignment as prospect theoretic optimization. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024.
- Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. Olympiadbench: A challenging benchmark for promoting AGI with olympiad-level bilingual multimodal scientific problems. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pp. 3828–3850. Association for Computational Linguistics, 2024.
- Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In Joaquin Vanschoren and Sai-Kit Yeung (eds.), *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*, 2021.
- Jingcheng Hu, Yinmin Zhang, Qi Han, Daxin Jiang, Xiangyu Zhang, and Heung-Yeung Shum. Open-reasoner-zero: An open source approach to scaling up reinforcement learning on the base model. *arXiv preprint arXiv:2503.24290*, 2025.
- Aaron Hurst, Adam Lerer, Adam P. Goucher, and et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.
- Hamish Ivison, Yizhong Wang, Jiacheng Liu, Zeqiu Wu, Valentina Pyatkin, Nathan Lambert, Noah A. Smith, Yejin Choi, and Hanna Hajishirzi. Unpacking DPO and PPO: disentangling best practices for learning from preference feedback. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024.
- Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay V. Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022.
- Yong Lin, Hangyu Lin, Wei Xiong, Shizhe Diao, Jianmeng Liu, Jipeng Zhang, Rui Pan, Haoxiang Wang, Wenbin Hu, Hanning Zhang, Hanze Dong, Renjie Pi, Han Zhao, Nan Jiang, Heng Ji, Yuan Yao, and Tong Zhang. Mitigating the alignment tax of RLHF. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pp. 580–606. Association for Computational Linguistics, 2024.
- Tianqi Liu, Yao Zhao, Rishabh Joshi, Misha Khalman, Mohammad Saleh, Peter J. Liu, and Jialu Liu. Statistical rejection sampling improves preference optimization. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024.
- Keming Lu, Bowen Yu, Fei Huang, Yang Fan, Runji Lin, and Chang Zhou. Online merging optimizers for boosting rewards and mitigating tax in alignment. *CoRR*, arXiv preprint arXiv:2405.17931, 2024.
- Rémi Munos, Michal Valko, Daniele Calandriello, Mohammad Gheshlaghi Azar, Mark Rowland, Zhaohan Daniel Guo, Yunhao Tang, Matthieu Geist, Thomas Mesnard, Côme Fiegl, Andrea

- 540 Michi, Marco Selvi, Sertan Girgin, Nikola Momchev, Olivier Bachem, Daniel J. Mankowitz,
541 Doina Precup, and Bilal Piot. Nash learning from human feedback. In *Forty-first Interna-*
542 *tional Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. Open-
543 Review.net, 2024.
- 544
- 545 Long Ouyang, Jeffrey Wu, Xu Jiang, and et al. Training language models to follow instructions
546 with human feedback. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho,
547 and A. Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on*
548 *Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November*
549 *28 - December 9, 2022*, 2022.
- 550 Arka Pal, Deep Karkhanis, Samuel Dooley, Manley Roberts, Siddartha Naidu, and Colin White.
551 Smaug: Fixing failure modes of preference optimisation with dpo-positive. *arXiv preprint*
552 *arXiv:2402.13228*, 2024.
- 553
- 554 Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and
555 Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model.
556 In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine
557 (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural*
558 *Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16,*
559 *2023*, 2023.
- 560 Amir Saeidi, Shivanshu Verma, and Chitta Baral. Insights into alignment: Evaluating DPO and its
561 variants across multiple tasks. *arXiv preprint arXiv:2404.14723*, 2024.
- 562
- 563 Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec
564 Radford, Dario Amodei, and Paul F. Christiano. Learning to summarize with human feedback.
565 In Hugo Larochelle, Marc’Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien
566 Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural*
567 *Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.
- 568
- 569 Yunhao Tang, Zhaohan Daniel Guo, Zeyu Zheng, Daniele Calandriello, Rémi Munos, Mark Row-
570 land, Pierre Harvey Richemond, Michal Valko, Bernardo Ávila Pires, and Bilal Piot. Generalized
571 preference optimization: A unified approach to offline alignment. In *Forty-first International Con-*
572 *ference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net,
573 2024.
- 574
- 575 Teng Xiao, Yige Yuan, Huaisheng Zhu, Mingxiao Li, and Vasant G. Honavar. Cal-dpo: Cali-
576 brated direct preference optimization for language model alignment. In Amir Globersons, Lester
577 Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang
578 (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural*
579 *Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 -*
580 *15, 2024*, 2024.
- 581
- 582 Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton
583 Murray, and Young Jin Kim. Contrastive preference optimization: Pushing the boundaries of
584 LLM performance in machine translation. In *Forty-first International Conference on Machine*
585 *Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024a.
- 586
- 587 Shusheng Xu, Wei Fu, Jiakuan Gao, Wenjie Ye, Weilin Liu, Zhiyu Mei, Guangju Wang, Chao Yu,
588 and Yi Wu. Is DPO superior to PPO for LLM alignment? A comprehensive study. In *Forty-first*
589 *International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*.
590 OpenReview.net, 2024b.
- 591
- 592 An Yang, Baosong Yang, Beichen Zhang, and et al. Qwen2.5 technical report. *arXiv preprint*
593 *arXiv:2412.15115*, 2024.
- 594
- 595 Yao Zhao, Rishabh Joshi, Tianqi Liu, Misha Khalman, Mohammad Saleh, and Peter J. Liu. Slic-hf:
Sequence likelihood calibration with human feedback. *arXiv preprint arXiv:2305.10425*, 2023.

A DERIVATIONS AND PROOFS

A.1 GRADIENT OF BPO LOSS

We consider the following BPO loss function:

$$\mathcal{L}(\theta) = -\mathbb{E}_{(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l) \sim D} \left[f \left(\min \left(\beta \log \frac{\pi_\theta(\mathbf{y}_w | \mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_w | \mathbf{x})}, -\alpha \beta \log \frac{\pi_\theta(\mathbf{y}_l | \mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_l | \mathbf{x})} \right) \right) \right]. \quad (9)$$

The loss function can be simplified as follows:

$$\mathcal{L}(\theta) = -\mathbb{E}_{(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l)} [f(\min(\beta r_w, -\alpha \beta r_l))]. \quad (10)$$

Where,

$$r_w = \log \frac{\pi_\theta(\mathbf{y}_w | \mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_w | \mathbf{x})}, \quad r_l = \log \frac{\pi_\theta(\mathbf{y}_l | \mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_l | \mathbf{x})}. \quad (11)$$

Let: $z = \min(\beta r_w, -\alpha \beta r_l)$, we compute the gradient:

$$\nabla_\theta \mathcal{L}(\theta) = -\nabla_\theta \mathbb{E}[f(z)] = -\mathbb{E}[\nabla_\theta f(z)]. \quad (12)$$

By the chain rule: $\nabla_\theta f(z) = f'(z) \cdot \nabla_\theta z$, thus:

$$\nabla_\theta \mathcal{L}(\theta) = -\mathbb{E}[f'(z) \cdot \nabla_\theta z]. \quad (13)$$

To proceed, we analyze $\nabla_\theta z$, which depends on which argument achieves the minimum in the definition of z .

Case 1: $\beta r_w < -\alpha \beta r_l$, then $z = \beta r_w$, so:

$$\nabla_\theta f(z) = f'(\beta r_w) \cdot \beta \cdot \nabla_\theta \log \pi_\theta(\mathbf{y}_w | \mathbf{x}). \quad (14)$$

Case 2: $\beta r_w \geq -\alpha \beta r_l$, then $z = -\alpha \beta r_l$, so:

$$\nabla_\theta f(z) = f'(-\alpha \beta r_l) \cdot (-\alpha \beta) \cdot \nabla_\theta \log \pi_\theta(\mathbf{y}_l | \mathbf{x}). \quad (15)$$

Combining both cases, the gradient of BPO loss function is:

$$\nabla_\theta \mathcal{L}(\theta) = -\mathbb{E}_{(\mathbf{x}, \mathbf{y}_w, \mathbf{y}_l)} \begin{cases} f'(\beta r_w) \cdot \beta \cdot \nabla_\theta \log \pi_\theta(\mathbf{y}_w | \mathbf{x}), & \text{if } \beta r_w < -\alpha \beta r_l, \\ f'(-\alpha \beta r_l) \cdot (-\alpha \beta) \cdot \nabla_\theta \log \pi_\theta(\mathbf{y}_l | \mathbf{x}), & \text{otherwise.} \end{cases} \quad (16)$$

A.2 PROOFS OF THEOREM 1

BPO maximizes ρ_θ^b , where γ denotes the margin achieved during optimization such that: $\rho_\theta^b \geq \gamma$.

Recall that:

$$\rho_\theta^b = \min(r_w, -\alpha r_l). \quad (17)$$

For this minimum to be at least γ , both components must individually satisfy:

$$r_w \geq \gamma \quad \text{and} \quad -\alpha r_l \geq \gamma. \quad (18)$$

Focusing on the first inequality, $r_w \geq \gamma$, and expanding r_w gives:

$$\beta \log \frac{\pi_\theta(\mathbf{y}_w | \mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_w | \mathbf{x})} \geq \gamma. \quad (19)$$

Dividing both sides by β (noting that $\beta > 0$):

$$\log \frac{\pi_{\theta}(\mathbf{y}_w|\mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_w|\mathbf{x})} \geq \frac{\gamma}{\beta}. \quad (20)$$

Applying the exponential function to both sides to eliminate the logarithm:

$$\frac{\pi_{\theta}(\mathbf{y}_w|\mathbf{x})}{\pi_{\text{ref}}(\mathbf{y}_w|\mathbf{x})} \geq \exp\left(\frac{\gamma}{\beta}\right). \quad (21)$$

Finally, rearranging yields the desired result:

$$\pi_{\theta}(\mathbf{y}_w|\mathbf{x}) \geq \exp\left(\frac{\gamma}{\beta}\right) \pi_{\text{ref}}(\mathbf{y}_w|\mathbf{x}). \quad (22)$$

B ADDITIONAL RESULTS

We present the reward dynamics of chosen responses (r_w) and rejected responses (r_l), as well as the evolution of their margins ($r_w - r_l$), for both DPO and BPO using different loss functions during training. As shown in Figure 7, despite various efforts to improve DPO, none of the existing variants effectively address the DCR problem. In these methods, r_w consistently declines throughout training, which contradicts the intended training objective. In contrast, our proposed method, BPO, successfully mitigates the DCR issue. When applied with different preference optimization functions, BPO ensures that all r_w values remain positive and continue to increase during training. This demonstrates the clear advantage of using a balanced reward margin over the traditional relative reward margin in preference optimization.

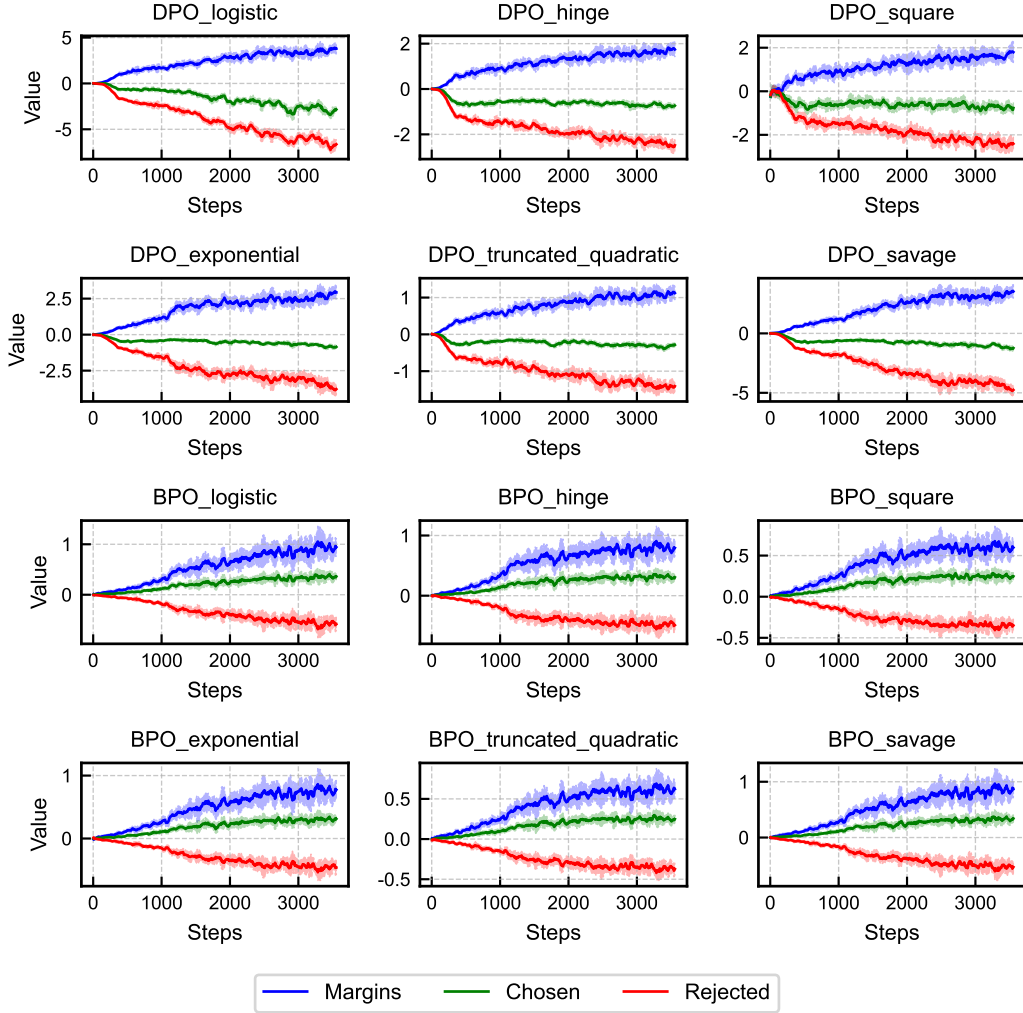


Figure 7: The reward dynamics of both chosen and rejected responses, along with the evolution of their margins. These results demonstrate that BPO consistently maintains positive r_w values, which continue to increase during training. Note that the gap adaptor is set to 1.0 for these comparisons.

C CASE STUDY

This section presents comparative examples between BPO and the baseline DPO algorithm, along with its variant IPO, the results are shown below in Tables 4 and 5.

Table 4: Examples comparing BPO with the baseline DPO show that BPO arrives at the correct solution more concisely.





| | |
|---------------|---|
| Prompt | Human: Find the roots of $(x - 3)^3 + (x - 7)^3 = (2x - 10)^3$. Assistant: |
| DPO | <p>First, let's simplify the equation. We can make a substitution to make the equation easier to handle. Let $y = x - 5$. This substitution is chosen because 5 is the midpoint. When $x = y + 5$, the equation becomes: $(y + 5 - 3)^3 + (y + 5 - 7)^3 = (2(y + 5) - 10)^3$ $(y + 2)^3 + (y - 2)^3 = (2y)^3$ This simplifies. Let $(y + 2)^3 + (y - 2)^3 = (2y)^3$ Expanding the cubes: $(y + 2)^3 = y^3 + 12y^2 + 8y + 8$ $(y - 2)^3 = y^3 - 12y^2 - 8y + 8$ So, $(y + 2)^3 + (y - 2)^3 = y^3 + 12y^2 + 8y + 8 + y^3 - 12y^2 - 8y + 8 = 8y^3$ We can simplify this by combining the terms: $y^3 + 12y^2 + 8y + 8 + y^3 - 12y^2 - 8y + 8 = 8y^3$ $2y^3 + 16 = 8y^3$ Rearranging terms, we $8y^3 - 2y^3 = 16 \implies 6y^3 = 16$ Simplify further: $y^3 = \frac{16}{6} = \frac{8}{3}$ or there is a simpler transformation we, look at the structure. Instead, let's look at the equation in a different way. Notice that the equation $(y + 2)^3 + (y - 2)^3 = (2y)^3$ can. Notice that if we let $a = y + 2$, $b = y - 2$, and $c = 2y$, we. One obvious solution to $a^3 + b^3 = c^3$ we recall that equations like $(y + 2)^3 + (y - 2)^3 = (2y)^3$ often resolves to simple substitutions or specific values. Not directly, we see that: If $y = 0$, then $(2)^3 + (-2)^3 = 0^3$, which is $8 - 8 = 0$, which is true. Another way to see Let $y = x - 5$. If $y = 0$, then $x = 5$. We can substitute $x - 5 = 0$ If $x = 5$: $x - 3 = 2$, $x - 7 = -2$, $2x - 10 = 0$ Clearly, $(2)^3 + (-2)^3 = 0^3$, which is $8 - 8 = 0$, and it So, $x = 5$ is a root. We can use. Thus, Let's box, 5 is the solution. </p> |
| BPO | <p>To solve the equation $(x - 3)^3 + (x - 7)^3 = (2x - 10)^3$, we can use a substitution to simplify the equation. Let's set $y = x - 5$. This substitution is chosen because it centers the terms around a common midpoint, which can simplify the algebra. With this substitution, we have: $x - 3 = y + 2$, $x - 7 = y - 2$, $2x - 10 = 2y$. Substituting these into the original equation, we get: $(y + 2)^3 + (y - 2)^3 = (2y)^3$. Expanding both sides, we have: $y^3 + 6y^2 + 12y + 8 + y^3 - 6y^2 + 12y - 8 = 8y^3$, $2y^3 + 24y = 8y^3$, $24y = 6y^3$, $4y = y^3$, $y^3 - 4y = 0$, $y(y^2 - 4) = 0$, $y(y - 2)(y + 2) = 0$. So, the solutions for y are $y = 0$, $y = 2$, and $y = -2$. Converting back to x, we have: $x = 5 + 0 = 5$, $x = 5 + 2 = 7$, $x = 5 - 2 = 3$. So, the final answer is: 3, 5, 7 </p> |

Table 5: Example comparing BPO with the the DPO variant IPO.

| | |
|---------------|---|
| Prompt | Human: For some real numbers a and b , the equation $8x^3 + 4ax^2 + 2bx + a = 0$ has three distinct positive roots. If the sum of the base-2 logarithms of the roots is 5, what is the value of a ? Assistant: |
| IPO | Given the equation $8x^3 + 4ax^2 + 2bx + a = 0$ with roots r_1, r_2, r_3 , we are also given that the roots are positive and distinct. By the properties of logarithms and the problem statement, we, know that: $\log_2(r_1) + \log_2(r_2) + \log_2(r_3) = 5$ Using the logarithm property that $\log_2(r_1 r_2 r_3) = \log_2(r_1) + \log_2(r_2) + \log_2(r_3)$, we. can deduce that: $\log_2(r_1 r_2 r_3) = 5 \implies r_1 r_2 r_3 = 2^5 = 32$, which means the product of the roots $r_1 r_2 r_3 = 32$. By Vieta's formulas for the polynomial $8x^3 + 4ax^2 + 2bx + a = 0$, the product of the roots for a polynomial $Ax^3 + Bx^2 + Cx + D = 0$ is given by $-\frac{D}{A}$. Here, the leading coefficient $A = 8$ and the constant term is a , so: $r_1 r_2 r_3 = -\frac{a}{8} = \frac{a}{8} \implies \frac{a}{8} = 32 \implies a = 8 \cdot 32 = 256$. Thus, the value of a is 256. The final answer is: 256  |
| BPO | To solve the problem, we need to use the properties of polynomials and logarithms. Here's the step-by-step reasoning: 1. Sum of the Base-2 Logarithms of the Roots: If the roots of the polynomial $8x^3 + 4ax^2 + 2bx + a = 0$ are r_1, r_2, r_3 , then the sum of the base-2 logarithms of the roots is given by: $\log_2(r_1) + \log_2(r_2) + \log_2(r_3) = 5$ Using the properties of logarithms, this can be rewritten as: $\log_2(r_1 r_2 r_3) = 5$ This implies: $r_1 r_2 r_3 = 2^5 = 32$ 2. Product of the Roots: By Vieta's formulas for the polynomial $8x^3 + 4ax^2 + 2bx + a = 0$, the product of the roots r_1, r_2, r_3 is given by: $r_1 r_2 r_3 = -\frac{a}{8}$ From the previous step, we know $r_1 r_2 r_3 = 32$, so: $-\frac{a}{8} = 32$ Solving for a : $a = -256$ The value of a is -256  |

D LLM USAGE STATEMENT

LLMs were used in the preparation of this paper solely for grammatical correction and language polishing, and they help improve the clarity and fluency of the writing. All intellectual contributions, including concept development, algorithm design, experimentation, and interpretation of results, are entirely the work of the authors.