A General Theoretical Paradigm to Understand Learning from Human Preferences

Mohammad Gheshlaghi Azar **Mark Rowland** **Bilal Piot**
Daniel Guo **Daniele Calandriello** **Michal Valko** **Rémi Munos**
Google DeepMind

Abstract

The prevalent deployment of learning from human preferences through reinforcement learning (RLHF) relies on two important approximations: the first assumes that pairwise preferences can be substituted with pointwise rewards. The second assumes that a reward model trained on these pointwise rewards can generalize from collected data to out-of-distribution data sampled by the policy. Recently, Direct Preference Optimisation (DPO) has been proposed as an approach that bypasses the second approximation and learn directly a policy from collected data without the reward modelling stage. However, this method still heavily relies on the first approximation.

In this paper we try to gain a deeper theoretical understanding of these practical algorithms. In particular we derive a new general objective called ΨPO for learning from human preferences that is expressed in terms of pairwise preferences and therefore bypasses both approximations. This new general objective allows us to perform an in-depth analysis of the behavior of RLHF and DPO (as special cases of ΨPO) and to identify their potential pitfalls. We then consider another special case for ΨPO by setting Ψ simply to Identity, for which we can derive an efficient optimisation procedure, prove performance guarantees and demonstrate its empirical superiority to DPO on some illustrative examples.

Proceedings of the 27$^{\text{th}}$ International Conference on Artificial Intelligence and Statistics (AISTATS) 2024, Valencia, Spain. PMLR: Volume 238.

1 INTRODUCTION

Learning from human preferences (Christiano et al., 2017) is a paradigm adopted in the natural language processing literature to better align pretrained (Radford et al., 2018; Ramachandran et al., 2016) and instruction-tuned (Wei et al., 2022) generative language models to human desiderata. It consists in first collecting large amounts of data where each datum is composed of a context, pairs of continuations of the context, also called generations, and a pairwise human preference that indicates which generation is the best. Then, a policy generating *good* generations given a context is learnt from the collected data. We frame the problem of learning from human preferences as an offline contextual bandit problem (Lu et al., 2010). The goal of this bandit problem is that given a context to choose an action (playing the role of the generation) which is most preferred by a human rater under the constraint that the resulting bandit policy should be close to some known reference policy. The constraint of staying close to a known reference policy can be satisfied e.g., by using KL regularisation (Geist et al., 2019) and its role is to avoid model drift (Lazaridou et al., 2020; Lu et al., 2020).

A prominent approach to tackle the problem of learning from human preferences is through reinforcement learning from human feedback (RLHF, Ouyang et al., 2022; Stiennon et al., 2020) in which first a reward model is trained in the form of a classifier of preferred and dispreferred actions. Then the bandit policy is trained through RL to maximize this learned reward model while minimizing the distance with the reference policy. Recently RLHF has been used successfully in solving the problem of aligning generative language models with human preferences (Ouyang et al., 2022). Furthermore recent works such as direct preference optimisation (DPO, Rafailov et al., 2023) and (SLiC-HF, Zhao et al., 2023) have shown that it is possible to optimize the bandit policy directly from human preferences without learning a reward model. They also have shown that on a selection of standard language tasks they are competitive with the state of the art RLHF while they are simpler to implement and require less resources.

Despite this practical success, little is known regarding theoretical foundations of these practical methods. Notable exceptions, that consider specific special cases, are (Wang et al., 2023; Chen et al., 2022) and prior work on preference-based (Busa-Fekete et al., 2014, 2013) and dueling bandits and RL (Novoseller et al., 2020; Pacchiano et al., 2023). However, these theoretical works focus on providing theoretical guarantees in terms of regret bounds in the standard bandit setting and they do not deal with the practical setting of RLHF, DPO and SLiC-HF.

In this work, our focus is on bridging the gap between theory and practice by introducing a simple and general theoretical representation of the practical algorithms for learning from human preferences. In particular, we show that it is possible to characterise the objective functions of RLHF and DPO as special cases of a more general objective exclusively expressed in terms of pairwise preferences. We call this objective Ψ-preference optimisation (ΨPO) objective, where Ψ is an arbitrary non-deceasing mapping. We then analyze this objective function in the special cases of RLHF and DPO and investigate its potential pitfalls. Our theoretical investigation of RLHF and DPO reveals that in principle they can be both vulnerable to overfitting. This is due to the fact that those methods rely on the strong assumption that pairwise preferences can be substituted with ELo-score (pointwise rewards) via a Bradley-Terry (BT) modelisation (Bradley and Terry, 1952). In particular, this assumption could be problematic when the (sampled) preferences are deterministic or nearly deterministic as it leads to over-fitting to the preference dataset at the expense of ignoring the KL-regularisation term (see Sec. 4.2). We then present a simple solution to avoid the problem of overfitting, namely by setting Ψ to identity in the ΨPO. This method is called Identity-PO (IPO) and by construction bypasses the BT modelisation assumption for preferences (see Sec. 5). Finally, we propose a practical solution, via a sampled loss function (see Sec. 5.2), to optimize this simplified version of ΨPO empirically and, we compare its performance with DPO on simple bandit examples, providing empirical support for our theoretical findings (see Sec. 5.3 and Sec. 5.4).

2 NOTATIONS

In the remaining, we build on the notations of DPO (Rafailov et al., 2023). Given a context $x \in \mathcal{X}$, where $\mathcal{X}$ is the finite space of contexts, we assume a finite action space $\mathcal{Y}$. A policy $\pi \in \Delta_{\mathcal{Y}}^{\mathcal{X}}$ associates to each context $x \in \mathcal{X}$ a discrete probability distribution $\pi(.|x) \in \Delta_{\mathcal{Y}}$ where $\Delta_{\mathcal{Y}}$ is the set of discrete distributions over $\mathcal{Y}$. We denote $\mu \in \Delta_{\mathcal{Y}}^{\mathcal{X}}$ the behavior policy. From a given context x, let $y, y' \sim \mu(x)$ be two actions generated independently by the reference policy. These are then presented to human raters who express preferences for one of the generations, denoted as $y_w \succ y_l$ where y_w and y_l denote the preferred and dispreferred actions amongst $\{y, y'\}$ respectively. We then write true human preference $p^*(y \succ y'|x)$ the probability of y being preferred to y' knowing the context x. The probability comes from the randomness of the choice of the human we ask for their preference. So $p^*(y \succ y'|x) = \mathbb{E}_h[\mathbb{I}\{h \text{ prefers } y \text{ to } y' \text{ given } x\}]$, where the expectation is over humans h. We also introduce the expected preference of a generation y over a distribution μ knowing x, noted $p^*(y \succ \mu|x)$, via the following equation:

$$p^*(y \succ \mu|x) = \mathop{\mathbb{E}}_{y' \sim \mu(.|x)}[p^*(y \succ y'|x)] .$$

For any two policy $\pi, \mu \in \Delta_{\mathcal{Y}}^{\mathcal{X}}$ and a context distribution ρ we denote the total preference of policy π to μ as

$$p_\rho^*(\pi \succ \mu|x) = \mathop{\mathbb{E}}_{\substack{x \sim \rho \\ y \sim \pi(.|x)}} [p^*(y \succ \mu|x)] .$$

In practice, we do not observe p^* directly, but samples $I(y, y'|x)$ from a Bernoulli distribution with mean $p^*(y \succ y'|x)$ (i.e., $I(y, y'|x)$ is 1 with probability $p^*(y \succ y'|x)$ and 0 otherwise). In particular, we assume we have access to the preferences through a dataset of rated generations $\mathcal{D} = (x_i, y_i, y_i')_{i=1}^N = (x_i, y_{w,i} \succ y_{l,i})_{i=i}^N$, where N is the dataset size. In addition, for a general finite set $\mathcal{S}$, a discrete probability distribution $\eta \in \Delta_{\mathcal{S}}$ and a real function $f \in \mathbb{R}^{\mathcal{S}}$, we note the expectation of f under η as $\mathbb{E}_{s\sim\eta}[f(s)] = \sum_{s \in \mathcal{S}} f(s)\eta(s)$. For a finite dataset $\mathcal{D} = (s_i)_{i=1}^N$, with $s_i \in \mathcal{S}$ for each i, and a real function $f \in \mathbb{R}^{\mathcal{S}}$, we denote the *empirical expectation* of f under $\mathcal{D}$ as $\mathbb{E}_{s\sim D}[f(s)] = \frac{1}{N}\sum_{i=1}^N f(s_i)$.

3 BACKGROUND

3.1 Reinforcement Learning from Human Feedback (RLHF)

The standard RLHF paradigm (Christiano et al., 2017; Stiennon et al., 2020) consists of two main stages: (i) learning the reward model; (ii) policy optimisation using the learned reward. Here we provide a recap of these stages.

3.1.1 Learning the Reward Model

Learning a reward model consists in training a binary classifier to discriminate between the preferred and dispreferred actions using a logistic regression loss. For the classifier, a popular choice is Bradley-Terry model: for a given context x and action y, we denote the pointwise reward, which can also be interpreted as an Elo score, of y given x by $r(x, y)$. The Bradley-Terry model represents the preference function $p(y \succ y'|x)$ (classifier) as a sigmoid of the difference of rewards:

$$p(y \succ y'|x) = \sigma\big(r(x,y) - r(x,y')\big)\,, \tag{1}$$

where $\sigma(\cdot)$ denotes the sigmoid function and plays the role of normalisation. Given the dataset $\mathcal{D} = (x_i, y_{w,i} \succ y_{l,i})_{i=1}^N$ one can learn the reward function by optimizing the following logistic regression loss

$$\mathcal{L}(r) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\Big[\log\left(p(y_w \succ y_l|x)\right)\Big]\,. \tag{2}$$

Assuming that $p^*(y \succ y'|x)$ conforms to the Bradley-Terry model, one can show that as the size of the dataset $\mathcal{D}$ grows, $p(y \succ y'|x)$ becomes a more and more accurate estimate of true $p^*(y \succ y'|x)$ and in the limit converges to $p^*(y \succ y'|x)$.

3.1.2 Policy Optimisation with the Learned Reward

Using the reward (Elo-score) $r(x, y)$ the RLHF objective is simply to optimize for the policy $\pi \in \Delta_{\mathcal{Y}}^{\mathcal{X}}$ that maximizes the expected reward while minimizing the distance between π and some reference policy $\pi_{\text{ref}} \in \Delta_{\mathcal{Y}}^{\mathcal{X}}$ through the following KL-regularized objective function:

$$J(\pi) = \mathbb{E}_\pi[r(x,y)] - \tau D_{\text{KL}}(\pi \;||\; \pi_{\text{ref}})\,, \tag{3}$$

in which the context x is drawn from ρ and the action y is drawn from $\pi(.|x)$. The divergence $D_{\text{KL}}(\pi||\pi_{\text{ref}})$ is defined as follows:

$$D_{\text{KL}}(\pi \;||\; \pi_{\text{ref}}) = \mathbb{E}_{x\sim\rho}[\text{KL}(\pi(.|x) \;||\; \pi_{\text{ref}}(.|x))]\,.$$

where:

$$\text{KL}(\pi(.|x) \;||\; \pi_{\text{ref}}(.|x)) = \mathbb{E}_{y\sim\pi(.|x)}\left[\log\left(\frac{\pi(y|x)}{\pi_{\text{ref}}(y|x)}\right)\right]\,.$$

The objective in Equation (3) is essentially optimized by PPO (Schulman et al., 2017) or similar approaches.

The combination of RLHF +PPO has been used with great success in practice (e.g., InsturctGPT and GPT-4 Ouyang et al., 2022; OpenAI, 2023).

3.2 Direct Preference Optimisation

An alternative approach to the RL paradigm described above is direct preference optimisation (DPO; Rafailov et al., 2023), which avoids the training of a reward model altogether. The loss that DPO optimises, given an empirical dataset $\mathcal{D}$, as a function of π, is given by

$$\min_\pi \mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\Bigg[-\log\sigma\Bigg(\tau\log\left(\frac{\pi(y_w|x)}{\pi(y_l|x)}\right) - \tau\log\left(\frac{\pi_{\text{ref}}(y_w|x)}{\pi_{\text{ref}}(y_l|x)}\right)\Bigg)\Bigg]\,. \tag{4}$$

In its population form, the loss takes on the form

$$\min_\pi \mathop{\mathbb{E}}_{\substack{x\sim\rho\\ y,y'\sim\mu}}\Bigg[-p^*(y\succ y'|x)\log\sigma\Bigg(\tau\log\left(\frac{\pi(y|x)}{\pi(y'|x)}\right) - \tau\log\left(\frac{\pi_{\text{ref}}(y|x)}{\pi_{\text{ref}}(y'|x)}\right)\Bigg)\Bigg]\,. \tag{5}$$

Rafailov et al. (2023) show that when (i) the Bradley-Terry model in Equation (1) perfectly fits the preference data and (ii) the optimal reward function r is obtained from the loss in Equation (2), then the global optimisers of the RLHF objective in Equation (3) and the DPO objective in Equation (5) perfectly coincide. In fact, this correspondence is true more generally; see Proposition 4 in Appendix B.

4 A GENERAL OBJECTIVE FOR PREFERENCE OPTIMISATION

A central conceptual contribution of the paper is to propose a general objective for RLHF, based on maximizing a non-linear function of preferences. To this end, we consider a general non-decreasing function $\Psi : [0,1] \to \mathbb{R}$, a reference policy $\pi_{\text{ref}} \in \Delta_{\mathcal{Y}}^{\mathcal{X}}$, and a real positive regularisation parameter $\tau \in \mathbb{R}_+^*$, and define the *Ψ-preference optimisation objective* (ΨPO) as

$$\max_\pi \mathop{\mathbb{E}}_{\substack{x\sim\rho\\ y\sim\pi(.|x)\\ y'\sim\mu(.|x)}}[\Psi(p^*(y\succ y'|x))] - \tau D_{\text{KL}}(\pi \;||\; \pi_{\text{ref}})\,. \tag{6}$$

This objective balances the maximisation of a potentially non-linear function of preference probabilities with the KL regularisation term which encourages policies to be close to the reference π_{ref}. This is motivated by the form of Equation (3), and we will see in the next subsection that it strictly generalises both RLHF and DPO, when the BT model holds.

4.1 A Deeper Analysis of DPO and RLHF

In the remaining, we omit the dependency on x for the ease of notations. This is without losing generality and all the following results are true for all $x \in \texttt{Supp}(\rho)$.

We first connect DPO and RLHF with the Ψ-preference objective in Equation (6), under the special choice of $\Psi(q) = \log(q/(1-q))$. More precisely, the following proposition establishes this connection.

Proposition 1. Suppose $\Psi(q) = \log(q/(1-q))$. When the Bradley-Terry model holds for p^*, that is, there exists $r : \mathcal{Y} \to \mathbb{R}$ such that

$$p^*(y \succ y') = \sigma(r(y) - r(y'))\,,$$

then the optimal policy for Equation (6), for the RLHF objective in Equation (3), and for the standard DPO objective in Equation (5) are identical.

Proof. Note that under the assumption that the Bradley-Terry model holds, we have

$$\begin{aligned}
\mathop{\mathbb{E}}_{y'\sim\mu}[\Psi(p^*(y \succ y'))] &= \mathop{\mathbb{E}}_{y'\sim\mu}\left[\Psi\left(\frac{e^{r(y)}}{e^{r(y)}+e^{r(y')}}\right)\right] \\
&= \mathop{\mathbb{E}}_{y'\sim\mu}[\log(e^{r(y)}/e^{r(y')})] \\
&= \mathop{\mathbb{E}}_{y'\sim\mu}[r(y) - r(y')] \\
&= r(y) - \mathop{\mathbb{E}}_{y'\sim\mu}[r(y')]\,.
\end{aligned}$$

This is equal to the reward in Equation (3), up to an additive constant, and so it therefore follows that the optimal policy for Equation (6) and for optimizing the objective in Equation (3) are identical. Further, as shown by Rafailov et al. (2023), the optimal policy for the DPO objective in Equation (5) and the objective in Equation (3) are identical, which gives the statement of the proposition. □

Applying this proposition to the objective function of Equation (6), for which there exists an analytical solution, reveals that under the BT assumption the closed-form solution to DPO and RLHF can be written as

$$\pi^*(y) \propto \pi_{\text{ref}}(y)\exp\left(\tau^{-1}\mathbb{E}_{y'\sim\mu}[\Psi(p^*(y \succ y'))]\right). \quad (7)$$

The derivations leading to Equation 7 is a well known result and is provided in App. A.1 for completeness.

4.2 Weak Regularisation and Overfitting

It is worth taking a step back, and asking what kinds of policies the above objective leads us to discover. This highly non-linear transformation of the preference probabilities means that small increases in preference probabilities already close to 1 are just as incentivized as larger increases in preference probabilities around 50%, which may be undesirable. The maximisation of logit-preferences, or Elo score in game-theoretic terminology, can also have counter-intuitive effects, even in transitive settings (Bertrand et al., 2023).

Consider the simple example where we have two actions y and y' such that $p^*(y \succ y') = 1$, i.e., y is always preferred to y'. Then the Bradley-Terry model would require that $(r(y) - r(y')) \to +\infty$ to satisfy (1). If we plug this into the optimal policy (7) then we would get that $\frac{\pi^*(y')}{\pi^*(y)} = 0$ (i.e., $\pi^*(y') = 0$) irrespective of what constant τ is used for the KL-regularisation. Thus the strength of the KL-regularisation becomes weaker and weaker the more deterministic the preferences.

The weakness of the KL-regularisation becomes even more pronounced in the finite data regime, where we only have access to a sample estimate of the preference $\hat{p}(y \succ y')$. Even if the true preference is, e.g., $p^*(y \succ y') = 0.8$, empirically it can be very possible when we only have a few data points to estimate $\hat{p}(y \succ y') = 1$, in which case the empirical optimal policy would make $\pi(y') = 0$ for any τ. This means that overfitting can be a substantial empirical issue, especially when the context and action spaces are extremely large as it is for large language models.

Why may standard RLHF be more robust to this problem in practice? While a purported advantage of DPO is that it avoids the need to fit a reward function, we observe that in practice when empirical preference probabilities are in the set $\{0, 1\}$, the reward function ends up being *underfit*. The optimal rewards in the presence of $\{0, 1\}$ preference probabilities are infinite, but these values are avoided, and indeed regularisation of the reward function has been observed to be an important aspect of RLHF training in practice (Christiano et al., 2017). This underfitting of the reward function is thus crucial in obtaining a final policy that is sufficiently regularised towards the reference policy π_{ref}, and DPO, in avoiding the training of the reward function, loses the regularisation of the policy that the underfitted reward function affords.

While standard empirical practices such as early-stopping can still be used as an additional form of regularisation to curtail this kind of overfitting, in the next section, we will introduce a modification of the ΨPO objective such that the optimal empirical policy can be close to π_{ref} even when preferences are deterministic.

5 IPO: ΨPO WITH IDENTITY MAPPING

We have observed in the previous section that DPO is prone to overfitting, and this stems from a combination of the unboundedness of Ψ, together with not training an explicit reward function. Not training a reward function directly is a clear advantage of DPO, but we would like to avoid the problems of overfitting as well.

This analysis of DPO motivates choices of Ψ which are bounded, ensuring that the KL regularisation in Equation 6 remains effective even in the regime of $\{0,1\}$-valued preferences, as it is often the case when working with empirical datasets. A particularly natural form of objective to consider is given by taking Ψ to be the identity mapping in Equation (6), leading to direct regularized optimisation of *total preferences*:

$$\max_{\pi} p^*_{\rho}(\pi \succ \mu) - \tau D_{\mathrm{KL}}(\pi \,||\, \pi_{\mathrm{ref}})\,. \tag{8}$$

The standard approach to optimize the objective function of Equation (8) is through RLHF with the choice of reward $r(y) = p^*(y \succ \mu)$. However both using RL and estimating the reward model $r(y)$ can be costly. Inspired by DPO one would like to devise an empirical solution for the optimisation problem of Equation (8) which can directly learn from the preference dataset. Thus it would be able to avoid RL and reward modeling altogether.

5.1 Derivations and Computationally Efficient Algorithm

As with DPO, it will be beneficial to re-express Equation (8) as an offline learning objective. To derive such an expression, we begin by following the derivation of Rafailov et al. (2023), manipulating the analytic expression for the optimal policy into a system of root-finding problems. As in the previous section, we drop dependence on the context x from our notation, as all arguments can be applied on a per-context basis.

Root-finding problems. Let $g(y) = \mathbb{E}_{y'\sim\mu}[\Psi(p^*(y \succ y'))]$. Then we have

$$\pi^*(y) \propto \pi_{\mathrm{ref}}(y)\exp(\tau^{-1}g(y))\,. \tag{9}$$

For any $y, y' \in \mathtt{Supp}(\pi_{\mathrm{ref}})$, we therefore have

$$\frac{\pi^*(y)}{\pi^*(y')} = \frac{\pi_{\mathrm{ref}}(y)}{\pi_{\mathrm{ref}}(y')}\exp(\tau^{-1}(g(y) - g(y')))\,. \tag{10}$$

By letting

$$h^*(y,y') = \log\left(\frac{\pi^*(y)\pi_{\mathrm{ref}}(y')}{\pi^*(y')\pi_{\mathrm{ref}}(y)}\right)$$

and rearranging Equation (10), we obtain

$$h^*(y,y') = \tau^{-1}\big(g(y) - g(y')\big)\,. \tag{11}$$

The core idea now is to consider a policy π, define

$$h_\pi(y,y') = \log\left(\frac{\pi(y)\pi_{\mathrm{ref}}(y'}{\pi(y')\pi_{\mathrm{ref}}(y)}\right),$$

and aim to solve the equations:

$$h_\pi(y,y') = \tau^{-1}\big(g(y) - g(y')\big)\,. \tag{12}$$

Loss for IPO. We now depart from the approach to the analysis employed by Rafailov et al. (2023), to obtain a novel offline formulation of Equation (6), in the specific case of Ψ as the identity function. In this case, Equation (12) reduces to

$$h_\pi(y,y') = \tau^{-1}\big(p^*(y \succ \mu) - p^*(y' \succ \mu)\big)\,.$$

We begin by re-expressing these root-finding problems as a single optimisation problem $L(\pi)$:

$$L(\pi) = \mathop{\mathbb{E}}_{y,y'\sim\mu}\left[\left(h_\pi(y,y') - \frac{p^*(y \succ \mu) - p^*(y' \succ \mu)}{\tau}\right)^2\right]. \tag{13}$$

One can easily show that for the choice of π^* we have $L(\pi^*) = 0$. Thus π^* is a global minimizer of $L(\pi)$. The following theorem establishes the uniqueness of this solution.

Theorem 2 (Uniqueness of Global/Local Optima)**.** Assume that $\mathtt{Supp}(\mu) = \mathtt{Supp}(\pi_{\mathrm{ref}})$ and define Π to be the set of policies π such that $\mathtt{Supp}(\pi) = \mathtt{Supp}(\mu)$. Then $\pi \mapsto L(\pi)$ has a unique local/global minimum in Π, which is π^*.

Proof. By assumption, $\pi^* \in \Pi$, and by definition $\forall \pi \in \Pi, L(\pi) \geq 0$ as $L(\pi)$ is an expectation of squared terms. Further, from Equation (11), it follows immediately that $L(\pi^*) = 0$, and so we deduce that π^* is a global optimum for L. We now show that there are no other local/global minima for L in Π.

We write $J = \mathtt{Supp}(\mu)$. We parametrise the set Π via vectors of logits $s \in \mathbb{R}^J$, setting $\pi_s(y) = \exp(s(y))/\sum_{y'\in J}\exp(s(y'))$ for $y \in J$, and $\pi_s(y) = 0$ otherwise. Let us write $\mathcal{L}(s) = L(\pi_s)$ for the objective as a function of the logits s.

$$\mathcal{L}(s) = \mathbb{E}_{y,y'\sim\mu}\left[\left[\frac{p^*(y \succ \mu) - p^*(y' \succ \mu)}{\tau} - (s(y) - s(y')) - \log\left(\frac{\pi_{\mathrm{ref}}(y')}{\pi_{\mathrm{ref}}(y)}\right)\right]^2\right]. \tag{14}$$

The objective is quadratic as a function of the logits s. Further, by expanding the quadratic above, we see that the loss can be expressed as a sum of squares

$$\sum_{y,y' \in J} \mu(y)\mu(y')(s(y) - s(y'))^2 \tag{15}$$

plus linear and constant terms. This is therefore a positive-semidefinite quadratic, and hence is convex. We thus deduce that all local minimisers of the loss $\mathcal{L}(s)$ are global minimisers as well (Boyd and Vandenberghe, 2004, Chap. 4). We now notice since π_s is a surjective continuous mapping from s to π one can easily show from the definition of local minimum that every local minimiser π of L corresponds to a set of local minimisers $\mathcal{S}_\pi$ of $\mathcal{L}$. Thus all local minimums of L are also global minimums as well.

Finally, the only direction s in which the quadratic in Equation (15) does not increase away from 0 is when all bracketed terms remain 0; that is, in the direction $(1, \ldots, 1) \in \mathbb{R}^J$. Thus, $\mathcal{L}(s)$ is strictly convex, except in the direction $(1, \ldots, 1)$. (Boyd and Vandenberghe, 2004, Chap. 3). However, modifying logits in the direction $e = (1, \ldots, 1)$ does not modify the resulting policy π_s, since, for $y \in J$,

$$\pi_{s+\lambda e}(y) = \frac{e^{s(y)+\lambda}}{\sum_{y' \in J} e^{s(y')+\lambda}} = \frac{e^{s(y)}}{\sum_{y' \in J} e^{s(y')}} = \pi_s(y)\,.$$

The strict convexity combined with the fact that π^* is a global minima proves that π^* is the unique global/local minima in Π (Boyd and Vandenberghe, 2004, Chap. 4). □

5.2 Sampled Loss for IPO

In order to obtain the sampled loss for IPO we need to show that we can build an unbiased estimate of the right-hand side of the equation (13). To this end, we consider the **Population IPO Loss**:

$$\mathop{\mathbb{E}}_{y,y' \sim \mu} \left[\left(h_\pi(y, y') - \tau^{-1} I(y, y') \right)^2 \right], \tag{16}$$

where $I(y, y')$ is drawn from a Bernoulli distribution with mean $p^*(y \succ y')$, i.e., $I(y, y')$ is 1 if y is preferred to y' (which happens with probability $p^*(y \succ y')$), and 0 otherwise. This straightforwardly yields a sample-based loss that can be used, by sampling a pair (y, y') from the preference dataset, and consulting the recorded preference to obtain a sample from $I(y, y')$. The following proposition justifies the switch from Equation (13) to Equation (16), by demonstrating their equality.

Proposition 3. The expressions in Equation (13) and Equation (16) are equal, up to an additive constant independent of π.

Proof. This equivalence is not completely trivial, since in general the conditional expectation

$$\mathbb{E}[h_\pi(Y, Y') - \tau^{-1} I(Y, Y') \mid Y = y, Y' = y']$$

is not equal to the corresponding quantity appearing in Equation (13), namely

$$h_\pi(y, y') - \tau^{-1}\big(p^*(y \succ \mu) - p^*(y' \succ \mu)\big)\,.$$

We instead need to exploit some symmetry between the distributions of y and y', and use the fact that $h_\pi(y, y')$ decomposes as an additive function of y and y'. To show this equality of losses, it is enough to focus on the "cross-terms" obtained when expanding the quadratics in Equations (13) and (16); that is, to show

$$\begin{aligned} &\mathop{\mathbb{E}}_{y,y' \sim \mu} \Big[h_\pi(y, y') I(y, y') \Big] \\ &= \mathop{\mathbb{E}}_{y,y' \sim \mu} \Big[h_\pi(y, y')(p^*(y \succ \mu) - p^*(y' \succ \mu)) \Big]\,. \end{aligned}$$

Now, starting with the right-hand side, and using the shorthand $\pi_y = \log(\pi(y))$, $\pi_y^{\mathrm{R}} = \log(\pi_{\mathrm{ref}}(y))$, $p_y = p^*(y \succ \mu)$, and similarly for y', we have

$$\begin{aligned} &\mathop{\mathbb{E}}_{y,y' \sim \mu} \Big[h_\pi(y, y')(p^*(y \succ \mu) - p^*(y' \succ \mu)) \Big] \\ &= \mathop{\mathbb{E}}_{y,y' \sim \mu} \Big[(\pi_y - \pi_{y'} + \pi_{y'}^{\mathrm{R}} - \pi_y^{\mathrm{R}})(p_y - p_{y'}) \Big] \\ &= \mathop{\mathbb{E}}_{y,y' \sim \mu} \Big[\pi_y p_y - \pi_y p_{y'} - \pi_{y'} p_y + \pi_{y'} p_{y'} \\ &\qquad + \pi_{y'}^{\mathrm{R}} p_y - \pi_{y'}^{\mathrm{R}} p_{y'} - \pi_y^{\mathrm{R}} p_y + \pi_y^{\mathrm{R}} p_{y'} \Big] \\ &= \mathop{\mathbb{E}}_{y,y' \sim \mu} \Big[(2p_y - 1)\pi_y - (2p_y - 1)\pi_y^{\mathrm{R}} \Big]\,, \end{aligned}$$

where we have used iid-ness of y and y', and $\mathbb{E}_{y \sim \mu}[p_y] = 1/2$. Turning to the left-hand side, we have

$$\begin{aligned} &\mathop{\mathbb{E}}_{y,y' \sim \mu} \Big[h_\pi(y, y') I(y, y') \Big] \\ &= \mathop{\mathbb{E}}_{y,y' \sim \mu} \Big[\big(\pi_y - \pi_{y'} + \pi_{y'}^{\mathrm{R}} - \pi_y^{\mathrm{R}}\big) I(y, y') \Big] \\ &= \mathop{\mathbb{E}}_{y \sim \mu} \Big[\big(\pi_y - \pi_y^{\mathrm{R}}\big) \mathop{\mathbb{E}}_{y' \sim \mu} [I(y, y') \mid y] \Big] \\ &\quad + \mathop{\mathbb{E}}_{y' \sim \mu} \Big[\big(-\pi_{y'} + \pi_{y'}^{\mathrm{R}}\big) \mathop{\mathbb{E}}_{y \sim \mu} [I(y, y') \mid y'] \Big] \\ &= \mathop{\mathbb{E}}_{y,y' \sim \mu} \Big[\pi_y p_y - \pi_{y'}(1 - p_{y'}) + \pi_{y'}^{\mathrm{R}}(1 - p_{y'}) - \pi_y^{\mathrm{R}} p_y \Big] \\ &= \mathop{\mathbb{E}}_{y,y' \sim \mu} \Big[(2p_y - 1)\pi_y - (2p_y - 1)\pi_y^{R} \Big]\,, \end{aligned}$$

where we use the fact that $\mathbb{E}_{y' \sim \mu} I(y, y') = p_y$ and $\mathbb{E}_{y \sim \mu} I(y, y') = 1 - p_{y'}$. This demonstrates equality of the losses, as required. □

We now discuss how to approximate the loss in Equation (16) with an empirical dataset. As in our earlier discussion, the empirical dataset $\mathcal{D}$ takes the form $(y_{w,i}, y_{l,i})_{i=i}^N$. Note that each datapoint $(y_{w,i}, y_{l,i})$ contributes two terms to an empirical approximation of Equation (16), with $(y, y', I(y, y')) = (y_{w,i}, y_{l,i}, 1)$, and also $(y, y', I(y, y')) = (y_{l,i}, y_{w,i}, 0)$. This symmetry is important to exploit, and leads to a reduction in the variance of the loss. The overall empirical loss is therefore given by

$$\frac{1}{2}\mathop{\mathbb{E}}_{(y_w,y_l)\sim D}\Big[(h_\pi(y_w,y_l)-\tau^{-1})^2+h_\pi(y_l,y_w)^2\Big] =$$
$$\frac{1}{2}\mathop{\mathbb{E}}_{(y_w,y_l)\sim D}\Big[(h_\pi(y_w,y_l)-\tau^{-1})^2+h_\pi(y_w,y_l)^2\Big],$$

which up to a constant equals:

$$\mathop{\mathbb{E}}_{(y_w,y_l)\sim D}\left[\tau\left(h_\pi(y_w,y_l)-\frac{\tau^{-1}}{2}\right)^2\right]. \qquad (17)$$

This simplified form of the loss provides some valuable insights on the way in which `IPO` optimizes the policy π: `IPO` learns from preferences dataset simply by regressing the gap between log-likelihood ratios $\log(\pi(y_w)/\pi(y_l))$ and $\log(\pi_{\text{ref}}(y_w)/\pi_{\text{ref}}(y_l))$ to $\frac{\tau^{-1}}{2}$. So the weaker the regularisation becomes, the higher would be the log-likelihood ratio of y_w to y_l. In other words `IPO`, unlike `DPO`, always regularizes its solution towards π_{ref} by controlling the gap between the log-likelihood ratios $\log(\pi(y_w)/\pi(y_l))$ and $\log(\pi_{\text{ref}}(y_w)/\pi_{\text{ref}}(y_l))$, thus avoiding the over-fitting to the preference dataset. We summarize the sampled `IPO` in Algorithm 1:

Algorithm 1 Sampled `IPO`

Require: Dataset D of prompts, preferred and dispreferred generations x, y_w and y_l, respectively. A reference policy π_{ref}

1: Define

$$h_\pi(y,y',x)=\log\left(\frac{\pi(y|x)\pi_{\text{ref}}(y'|x)}{\pi(y'|x)\pi_{\text{ref}}(y|x)}\right)$$

2: Starting from $\pi=\pi_{\text{ref}}$ minimize

$$\mathop{\mathbb{E}}_{(y_w,y_l,x)\sim D}\left[\tau\left(h_\pi(y_w,y_l,x)-\frac{\tau^{-1}}{2}\right)^2\right].$$

5.3 Illustrative Examples

To illustrate the qualitative difference between our algorithm and `DPO` we will consider a few simple cases. For simplicity we assume there is no context x, i.e., we are in the bandit setting.

5.3.1 Asymptotic Setting

We first consider the simple case where we have 2 actions only, y_1 and y_2, and a deterministic preference between them: $p^*(y_1 \succ y_2) = 1$. Suppose we start with a uniform π_{ref} and μ. We know from Section 4.2 that `DPO` will converge to the deterministic policy $\pi^*(y_1) = 1$, $\pi^*(y_2) = 0$ regardless of the value of τ. Thus even when the regularisation coefficient τ is very large, this is very different from the uniform π_{ref}.

Now, let us derive the optimal policy for `IPO`. We have $p^*(y_1 \succ \mu) = 3/4$ and $p^*(y_2 \succ \mu) = 1/4$. Plugging this into equation (9) with $\Psi = I$ we get that $\pi^*(y_1) = \frac{\exp(0.75\tau^{-1})}{\exp(0.75\tau^{-1})+\exp(0.25\tau^{-1})} = \sigma(0.5\tau^{-1})$, and $\pi^*(y_2) = \sigma(-0.5\tau^{-1})$, where σ is the sigmoid function. Hence we see that if we have large regularisation as $\tau \to +\infty$, then π^* converges to the uniform policy π_{ref}, and on the flip side as $\tau \to +0$, then $\pi^*(y_1) \to 1$ and $\pi^*(y_2) \to 0$, which is the deterministic optimal policy. The regularisation parameter τ can now actually be used to control how close to π_{ref} we are.

5.4 Sampled Preferences

So far we relied on the closed-form optimal policy from Eq. (9) to study `DPO` and `IPO`'s stability, but this equation is not applicable to more complex settings where we only have access to sampled preference instead of $p^\star$. We can still however find accurate approximations of the optimal policy by choosing a parametrisation π_θ and optimize θ with an empirical loss over a dataset and iterative gradient-based updates. We will use this approach to show two non-asymptotic examples where `DPO` over-fits the dataset of preferences and ignore π_{ref}: when one action y wins against all others `DPO` pushes $\pi_\theta(y)$ to 1 regardless of τ, and conversely when one action y never wins against the others `DPO` pushes $\pi_\theta(y)$ to 0 again regardless of τ. In the same scenarios, `IPO` does not converge to these degenerate solutions but instead remains close to π_{ref} based on the strength of the regularisation τ.

For both scenarios we consider a discrete space $\mathcal{Y} = \{y_a, y_b, y_c\}$ with 3 actions, and select a dataset of pairs $\mathcal{D} = \{(y_{w,i}, y_{l,j})\}$. Given $\mathcal{D}$, we leverage the empirical losses from Eq. 4 and Eq. 13 to find `DPO`'s and `IPO`'s optimal policy. We encode policies as $\pi_\theta(y_i) = \text{softmax}(\theta)_i$ using a vector $\theta \in \mathbb{R}^3$, and optimize them for 18000 steps using Adam (Kingma and Ba, 2014) with learning rate 0.01 and mini-batch size 9. Mini-batches are constructed using uniform sampling with replacement from $\mathcal{D}$. Both policies and losses are im-

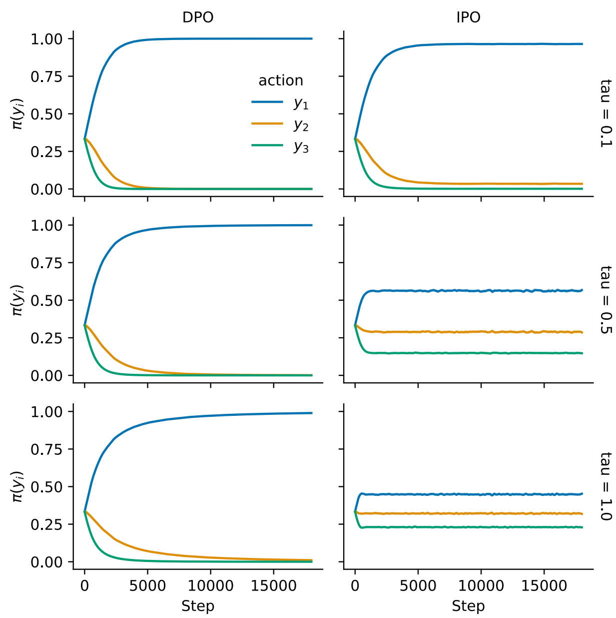


Figure 1: Comparison Between the Learning Curves of Action Probabilities of IPO and DPO for $\mathcal{D}_1$

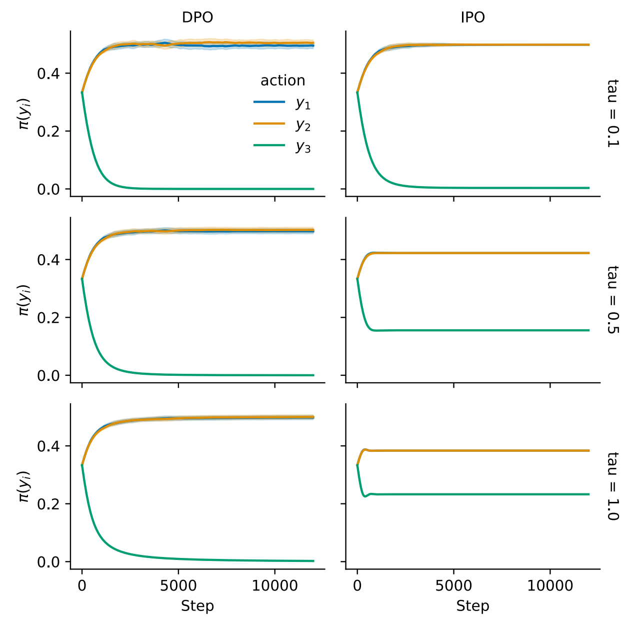


Figure 2: Comparison Between the Learning Curves of Action Probabilities of IPO and DPO for $\mathcal{D}_3$

plemented using the `flax` Python framework (Bradbury et al., 2018; Heek et al., 2023), and the Adam implementation is from `optax` (Babuschkin et al., 2020). For each set of hyper-parameters we repeat the experiment 10 times with different seeds, and report mean and 95% confidence intervals. Note that the confidence intervals are hard to see because they are very narrow. All experiments are executed on a modern cloud virtual machine with 4 cores and 32GB of ram.

IPO Avoids Greedy Policies For the first example we sample each unique action pair once to collect a dataset $\mathcal{D}$ containing 3 observed preferences. Due to symmetries of pairwise preferences sampling only 3 preferences can results in only two outcomes (up to permutations of the actions):

$$\mathcal{D}_1 = \{(y_a, y_b), (y_b, y_c), (y_a, y_c)\},$$
$$\mathcal{D}_2 = \{(y_a, y_b), (y_b, y_c), (y_c, y_a)\},$$

where we focus on $\mathcal{D}_1$, which represent a total ordering, rather than $\mathcal{D}_2$, which represent a cycle. The outcome of the experiment is reported in Fig. 1 in which, we report the learning curves for varying values of τ. We observe that DPO always converges to the deterministic policy for all values of τ. In other word DPO completely ignores the reference policy, no matter how strong is the regularisation term, and converges to the action which is preferred in the dataset. On the other hand, IPO prevent the policy from becoming greedy when the regularisation is strong.

IPO Does not Exclude Actions In the first example DPO converges to a deterministic policy because one action strictly dominates all others and the loss continues to push up its likelihood until it saturates. The opposite effect happens for the logical opposite condition, i.e., when one action does not have at least a victory in the dataset DPO will sets its probability to 0 regardless of τ. While this is less disruptive than the first example (a single probability is perturbed whereas previously the whole policy was warped by an over-achieving action) it is also much more common in real-world data. In particular, whenever the action space is large but the dataset small, some actions will necessarily be sampled rarely or only once, making it likely to never observe a victory. Especially because we do not have data on their performance π should stick close to π_{ref} for safety, but DPO's objective does not promote this.

In the final example the dataset consists of two observed preferences $\mathcal{D}_3 = \{(y_a, y_b), (y_b, y_a)\}$ and leave the pair (y_a, y_c) completely unobserved. We compute solutions using Adam once again, and report the results in Fig. 2 for varying values of τ. We observe again here that DPO ignores the prior π_{ref} completely, no matter how strong we regularize the objective, whereas IPO gradually decreases the probability of unobserved action with τ.

Performance of IPO/DPO on more complex preferences. Here we test out IPO and DPO using the more complex preference model over 4 actions

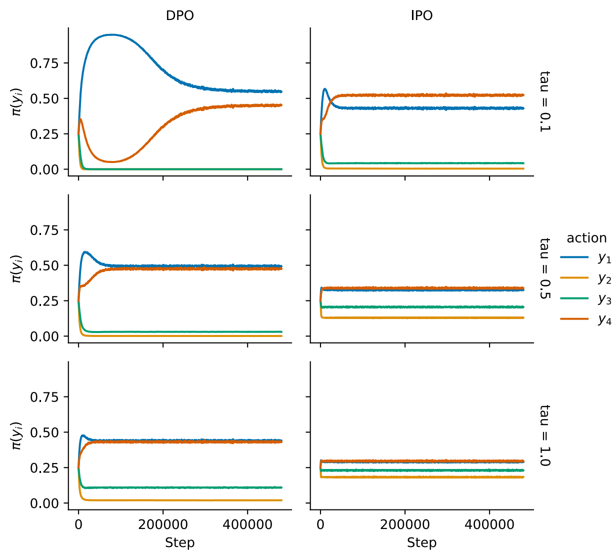


Figure 3: DPO and IPO on more complex preferences.

given in the table below:

$$\begin{pmatrix} .5 & .99 & .25 & .25 \\ .01 & .5 & .75 & .25 \\ .75 & .25 & .5 & .25 \\ .75 & .75 & .75 & .5 \end{pmatrix}.$$

These payoffs were selected to include non-transitive aspects (between the first 3 actions), and to illustrate the distinct behaviour of DPO and IPO in which IPO controls the distance between the learned policy and the reference policy (uniform) effectively as regularisation become stronger whereas DPO fails to do so (Figure 3). Furthermore, for $\tau = 0.1$, we can actually see a difference where DPO and IPO pick different actions as the best action. IPO picks action y_4, which can be considered as the best action in terms of not losing to any other action. DPO, on the other hand, picks action y_1, because y_1 beats y_2 with probability 0.99, which heavily skews the preference in y_1's favour.

6 CONCLUSION AND FUTURE WORK

We presented a unified objective, called ΨPO, for learning from preferences. It unifies RLHF and DPO methods. In addition, we introduced a particular case of ΨPO, called IPO, that allows to learn directly from preferences without a reward modelling stage and without relying on the Bradley-Terry modelisation assumption that assumes that pairwise preferences can be substituted with pointwise rewards. This is important because it allows to avoid the overfitting problem. This theoretical contribution is only useful in practice if an empirical sampled loss function can be derived. This is what we have done in Sec 5 where we show that IPO can be formulated as a root-finding problem from which an empirical sampled loss function can be derived. The IPO loss function is simple, easy to implement and theoretically justified. Finally, in Sec. 5.3 and Sec. 5.4, we provide illustrative examples where we highlight the instabilities of DPO when the preferences are fully-known as well as when they sampled. Those minimal experiments are sufficient to prove that IPO is better suited to learn from sampled preferences than DPO. Future works should scale those experiments to more complex settings such as training language models on human preferences data.

Acknowledgments

We thank Yasin Abbasi Yadkori and Csaba Szepesvari for feedback on a preliminary draft of the paper. We thank Ivo Danihelka for very insightful comments. We also thank the anonymous reviewers and area chair for their comments and feedback. Finally, we thank Doina Precup and Will Dabney for sponsoring this work and the entire Google DeepMind team.

References

Igor Babuschkin, Kate Baumli, Alison Bell, Surya Bhupatiraju, Jake Bruce, Peter Buchlovsky, David Budden, Trevor Cai, Aidan Clark, Ivo Danihelka, et al. The DeepMind JAX ecosystem, 2020. URL http://github.com/deepmind.

Quentin Bertrand, Wojciech Marian Czarnecki, and Gauthier Gidel. On the limitations of the Elo: Real-world games are transitive, not additive. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 2023.

Stephen P. Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge University Press, 2004.

James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018. URL http://github.com/google/jax.

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

Róbert Busa-Fekete, Balázs Szörényi, Paul Weng, Weiwei Cheng, and Eyke Hüllermeier. Preference-based reinforcement learning: Evolutionary direct policy search using a preference-based racing algorithm. *Machine Learning*, (3):327–351, 2014.

Róbert Busa-Fekete, Balázs Szörenyi, Paul Weng, Weiwei Cheng, and Eyke Hüllermeier. Preference-based evolutionary direct policy search. In *Autonomous Learning Workshop @ ICRA*, 2013.

Xiaoyu Chen, Han Zhong, Zhuoran Yang, Zhaoran Wang, and Liwei Wang. Human-in-the-loop: Provably efficient preference-based reinforcement learning with general function approximation. In *Proceedings of the International Conference on Machine Learning*, 2022.

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

Matthieu Geist, Bruno Scherrer, and Olivier Pietquin. A theory of regularized Markov decision processes. In *Proceedings of the International Conference on Machine Learning*, 2019.

Jonathan Heek, Anselm Levskaya, Avital Oliver, Marvin Ritter, Bertrand Rondepierre, Andreas Steiner, and Marc van Zee. Flax: A neural network library and ecosystem for JAX, 2023. URL http://github.com/google/flax.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations*, 2014.

Angeliki Lazaridou, Anna Potapenko, and Olivier Tieleman. Multi-agent communication meets natural language: Synergies between functional and structural language learning. In *Proceedings of the Annual Meeting of Association for Computational Linguistics*, 2020.

Tyler Lu, Dávid Pál, and Martin Pál. Contextual multi-armed bandits. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 2010.

Yuchen Lu, Soumye Singhal, Florian Strub, Aaron Courville, and Olivier Pietquin. Countering language drift with seeded iterated learning. In *Proceedings of the International Conference on Machine Learning*, 2020.

Ellen Novoseller, Yibing Wei, Yanan Sui, Yisong Yue, and Joel Burdick. Dueling posterior sampling for preference-based reinforcement learning. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, 2020.

OpenAI. Gpt-4 technical report, 2023.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller amd Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, 2022.

Aldo Pacchiano, Aadirupa Saha, and Jonathan Lee. Dueling RL: Reinforcement learning with trajectory preferences. *arXiv*, 2023.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. 2018.

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

Prajit Ramachandran, Peter J. Liu, and Quoc V. Le. Unsupervised pretraining for sequence to sequence learning. In *Proceedings of the Conference on Empirical Methods in Natural Language Processings*, 2016.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv*, 2017.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F. Christiano. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems*, 2020.

Yuanhao Wang, Qinghua Liu, and Chi Jin. Is RLHF more difficult than standard RL? *arXiv*, 2023.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. Finetuned language models are zero-shot learners. In *Proceedings of the International Conference on Learning Representations*, 2022.

Yao Zhao, Rishabh Joshi, Tianqi Liu, Misha Khalman, Mohammad Saleh, and Peter J Liu. SLiC-HF: Sequence likelihood calibration with human feedback. *arXiv*, 2023.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes] Analysed convergence and gradients.
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [No]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Yes]
   (c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [No] Source code not provided.
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Not Applicable]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

APPENDICES

A PROOFS

A.1 Existence and uniqueness of the regularized argmaximum

For completeness, we briefly recall the proof of existence and uniqueness of the argmaximum of the following regularized criterion that can also be found in the work of Rafailov et al. (2023):

$$\mathcal{L}_\tau(\delta) = \mathbb{E}_{s\in\delta}[f(s)] - \tau \mathrm{KL}(\delta \mid\mid \eta),$$
$$= \sum_{s\in\mathcal{S}} \delta(s) f(s) - \tau \mathrm{KL}(\delta \mid\mid \eta),$$

where $\mathcal{S}$ is a finite set, $f \in \mathbb{R}^{\mathcal{S}}$ a function mapping elements of $\mathcal{S}$ to real numbers, $\tau \in \mathbb{R}^*_+$ a strictly positive real number, $\delta \in \Delta_{\mathcal{S}}$ and $\eta \in \Delta_{\mathcal{S}}$ are discrete probability distributions over $\mathcal{S}$. In particular, we recall that a discrete probability distribution $\delta \in \Delta_{\mathcal{S}}$ can be identified as a positive real function $\delta \in \mathbb{R}^{\mathcal{S}}_+$ verifying:

$$\sum_{s\in\mathcal{S}} \delta(s) = 1.$$

Now, if we define the softmax probability $\delta^* \in \Delta_{\mathcal{S}}$ as:

$$\forall s \in \mathcal{S}, \delta^*(s) = \frac{\eta(s)\exp(\tau^{-1}f(s))}{\sum_{s'\in\mathcal{S}} \eta(s')\exp(\tau^{-1}f(s'))},$$

then, under the previous definitions, we have the following result:

$$\delta^* = \arg\max_{\delta\in\Delta_{\mathcal{S}}} \mathcal{L}_\tau(\delta)$$

Proof.

$$\begin{aligned}
\frac{\mathcal{L}_\tau(\delta)}{\tau} &= \sum_{s\in\mathcal{S}} \delta(s)\frac{f(s)}{\tau} - \mathrm{KL}(\delta \mid\mid \eta),\\
&= \sum_{s\in\mathcal{S}} \delta(s)\frac{f(s)}{\tau} - \sum_{s\in\mathcal{S}} \delta(s)\log\big(\frac{\delta(s)}{\eta(s)}\big),\\
&= \sum_{s\in\mathcal{S}} \delta(s)\big(\frac{f(s)}{\tau} - \log\big(\frac{\delta(s)}{\eta(s)}\big)\big),\\
&= \sum_{s\in\mathcal{S}} \delta(s)\big(\log\big(\exp(\tau^{-1}f(s))\big) - \log\big(\frac{\delta(s)}{\eta(s)}\big)\big),\\
&= \sum_{s\in\mathcal{S}} \delta(s)\big(\log\big(\frac{\eta(s)\exp(\tau^{-1}f(s))}{\delta(s)}\big)\big),\\
&= \sum_{s\in\mathcal{S}} \delta(s)\big(\log\big(\frac{\eta(s)\exp(\tau^{-1}f(s))\frac{\sum_{s'\in\mathcal{S}}\eta(s')\exp(\tau^{-1}f(s'))}{\sum_{s'\in\mathcal{S}}\eta(s')\exp(\tau^{-1}f(s'))}}{\delta(s)}\big)\big),\\
&= \sum_{s\in\mathcal{S}} \delta(s)\big(\log\big(\frac{\frac{\eta(s)\exp(\tau^{-1}f(s))}{\sum_{s'\in\mathcal{S}}\eta(s')\exp(\tau^{-1}f(s'))}}{\delta(s)}\big)\big) + \sum_{s\in\mathcal{S}} \delta(s)\log\big(\sum_{s'\in\mathcal{S}}\eta(s')\exp(\tau^{-1}f(s'))\big),\\
&= \sum_{s\in\mathcal{S}} \delta(s)\big(\log\big(\frac{\delta^*(s)}{\delta(s)}\big)\big) + \log\big(\sum_{s'\in\mathcal{S}}\eta(s')\exp(\tau^{-1}f(s'))\big),\\
&= -\mathrm{KL}(\delta \mid\mid \delta^*) + \log\big(\sum_{s'\in\mathcal{S}}\eta(s')\exp(\tau^{-1}f(s'))\big).
\end{aligned}$$

By definition of the KL, we now that $\delta^* = \arg\max_{\delta\in\Delta_{\mathcal{S}}} \Big[-\mathrm{KL}(\delta \mid\mid \delta^*) \Big]$ and as:

$$-\mathrm{KL}(\delta \mid\mid \delta^*) = \frac{\mathcal{L}_\tau(\delta)}{\tau} - \log\big(\sum_{s'\in\mathcal{S}} \eta(s')\exp(\tau^{-1}f(s'))\big)$$

where $\log\big(\sum_{s'\in\mathcal{S}} \eta(s')\exp(\tau^{-1}f(s'))\big)$ is a constant (does not depend on δ) and τ a positive multiplicative term, then $-\mathrm{KL}(\delta \mid\mid \delta^*)$ and $\mathcal{L}_\tau(\delta)$ share the same argmaximum. This concludes the proof. □

A.2 Non-uniqueness when $\texttt{Supp}(\pi) \neq \texttt{Supp}(\mu)$:

Notice that if we search for a solution where the support of π is strictly larger than that of μ then there could be multiple solutions. Let us illustrate this case with a simple example. Consider a single state x and 3 actions y_1, y_2, y_3. The reference policy π_{ref} is uniform over $\{y_1, y_2, y_3\}$ and the policy μ assigns a probability 1/2 to both y_1 and y_2 and 0 probability to y_3.

Thus the loss is $L(\pi) = 2\Big(\tau^{-1}\big(p^*(y_1 \succ \mu) - p^*(y_2 \succ \mu)\big) - \log\frac{\pi(y_1)}{\pi(y_2)}\Big)^2$. We deduce that any policy $\pi = (p, q, 1-p-q)$ such that $\frac{p}{q} = e^{\tau^{-1}(p^*(y_1\succ\mu)-p^*(y_2\succ\mu))}$ is a global minimum of $L(\pi)$.

In particular there are an infinity of solutions different from the optimal solution π^*. The problem comes from the fact that when the support of μ does not cover the whole action space there are not enough constraints to uniquely characterize π^*. Assuming that the supports of π_{ref} and μ coincide enables us to recover uniqueness of the solution, as proven in Theorem 2.

B ADDITIONAL RESULTS

In this section, we show the equivalence of DPO and RLHF, regardless of whether the preference model p^* corresponds to a Bradley-Terry model. Note that the assumption of the existence of a minimizer is to exclude cases where the loss is minimized by taking the rewards of certain actions to $+/-\infty$.

Proposition 4. Consider a preference model p^* such that there exists a minimizer to the Bradley-Terry loss

$$\arg\min_r \quad -\mathop{\mathbb{E}}_{\substack{x\sim\rho\\ y\sim\mu(\cdot|x)\\ y'\sim\mu(\cdot|x)}} \left[p^*(y\succ y'|x)\log\sigma(r(x,y)-r(x,y'))\right].$$

Then, the optimal policy for the DPO objective in Equation (4) and for the RLHF objective in Equation (3) with reward model given as the minimizer to the Bradley-Terry loss above are identical, regardless of whether or not p^* corresponds to a Bradley-Terry preference model.

Proof. Recall that the optimal policy π^*_r for a given reward function r for the objective in Equation (3) is given by $\pi^*_r(y|x) \propto \pi_{\text{ref}}(y|x)\exp(\tau^{-1}r(x,y))$. It therefore follows that

$$\begin{aligned}
&-\mathop{\mathbb{E}}_{\substack{x\sim\rho\\ y,y'\sim\mu(\cdot|x)}} \left[p(y\succ y'|x)\log\sigma(r(x,y)-r(x,y'))\right]\\
=&-\mathop{\mathbb{E}}_{\substack{x\sim\rho\\ y,y'\sim\mu(\cdot|x)}} \left[p(y\succ y'|x)\log\sigma\Big(\tau\log\left(\frac{\pi^*_r(y|x)}{\pi^*_r(y'|x)}\right) - \tau\log\left(\frac{\pi_{\text{ref}}(y|x)}{\pi_{\text{ref}}(y'|x)}\right)\Big)\right].
\end{aligned}$$

In words, the value of the Bradley-Terry reward objective for r is the value of the DPO objective for π^*_r. We recall also that the map $r \mapsto \pi^*_r$ is surjective.

Now, suppose r is optimal for the Bradley-Terry reward objective, meaning that π^*_r is optimal for the RLHF objective. If π^*_r is not optimal for the DPO objective, then there exists another policy π' that obtains a strictly lower value for the DPO loss. But then there exists a reward function r' such that $\pi' = \pi^*_{r'}$, such as $r'(x,y) = \tau\log(\pi'(y|x)/\pi_{\text{ref}}(y|x))$, and this r' therefore obtains a lower Bradley-Terry loss than r, a contradiction.

Similarly, if π^* is optimal for the DPO objective, the corresponding reward function $r(x,y) = \tau\log(\pi^*(y|x)/\pi_{\text{ref}}(y|x))$ must be optimal for the Bradley-Terry reward loss. The corresponding optimizer for the RLHF objective is then given by $\pi(y|x) \propto \pi_{\text{ref}}(y|x)\exp(\tau^{-1}\tau\log(\pi^*(y|x)/\pi_{\text{ref}}(y|x))) = \pi^*(y|x)$, as required. □

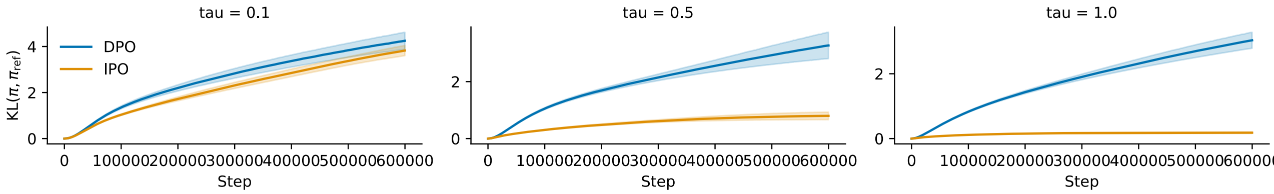


Figure 4: Comparison between the KL distance of IPO and DPO from the uniform policy for $\mathcal{D}_4$.

C ADDITIONAL EXPERIMENTS

In this section we describe experimental results, and further details regarding the experiments of Section 5.4.

C.1 Example on larger action spaces

In Section 5.3 we showed illustrative examples on small domains to support that:

- when we have a "winning" action that empirically strictly dominates all others (i.e., $\exists y$ s.t. $\forall y', \hat{p}(y, y') = 1$) then DPO will converge to the deterministic policy $\pi(y) = 1$;
- conversely, when we have a "losing" action that is empirically dominated by all others (i.e., $\exists y$ s.t. $\forall y', \hat{p}(y, y') = 0$ then DPO will converge to a policy with $\pi(y) = 0$.

In this section we will consider a more realistic, larger scale scenario where this can happen. In particular, we consider a discrete space $\mathcal{Y}$ with 1000 actions, and a preference function that is indifferent to all actions (i.e., $p^\star(y, y') = 1/2$ for all $y, y' \in \mathcal{Y}$). Combining this with a uniform π_{ref} and the KL regularization it is easy to see that $\pi^\star$ is also the uniform policy, and that both DPO and IPO will asymptotically converge to it. However if we consider a low-sample regime where we only observe a dataset $\mathcal{D}_4$ of 1000 preferences sampled uniformly at random we have that:

- a few of the actions will approximately fall in the "winning" category, where the dataset only records victories for them but not against all other actions;
- many actions will approximately fall in the "losing" category, where once again we only record losses but not against all other actions.

In this more empirical situation, the implied preference function $\hat{p}$ is substantially different from $p^\star$, and we expect DPO to converge to a near-deterministic policy around a few of the "winning" action, and completely ignore the "losing" actions, with τ inducing only a very weak regularizing effect. Conversely, IPO will be much more sensitive to τ and will be much closer to the optimal uniform policy (i.e., closer to π_{ref}). This is exactly the behaviour we observe in Figure 4, where we again compute the solution using Adam for 60 epochs with learning rate 0.01 and batch size 90, and plot the distance $\text{KL}(\pi, \pi_{\text{ref}})$ of the policy during training. While for low τ both IPO and DPO fail to reconstruct the true $p^\star$ and corresponding $\pi^\star = \pi_{\text{ref}}$ (since the sample size is too small), as we increase τ DPO continues to ignore the constraint while IPO correctly incorporates it to converge to a near-uniform policy. To give a better picture, we also report log-probabilities $\log(\pi(y_i))$ and empirical counts for the final policies generated by each algorithm in Figure 5. On the left we see log-probabilities plotted for each action (for ease of visualization we order the actions without loss of generality). It is evident that for lower τ DPO pushes the approximately "winning" actions very close to one, and the approximately "losing" actions all the way to numerical zero (i.e., e^{-57}), resulting in a very uneven distribution, and angled plot. IPO conversely has a much more stable behaviour and generate more uniform log-probabilities, resulting in an almost horizontal plot. This is reflected by the histogram plots on the right where we see that the distribution of probabilities is much more concentrated around the same value (i.e., uniform) for IPO than for DPO, which has long tails of both low and high probability actions.

We conclude this expanded experimental section with an experiment to investigate how being more sensitive to regularization can help with statistical efficiency, in particular we consider the same setting as before (i.e., $p^\star(y, y') = 1/2$ and uniform π_{ref}) but this time we will set the number of actions to 100, and draw a varying number of observations between $[100, 1000]$, and report once again $\text{KL}(\pi, \pi_{\text{ref}})$ of the final policy computed by DPO and IPO. The result is reported in Figure 6. As we can see, both IPO and DPO converge to 0 KL distance

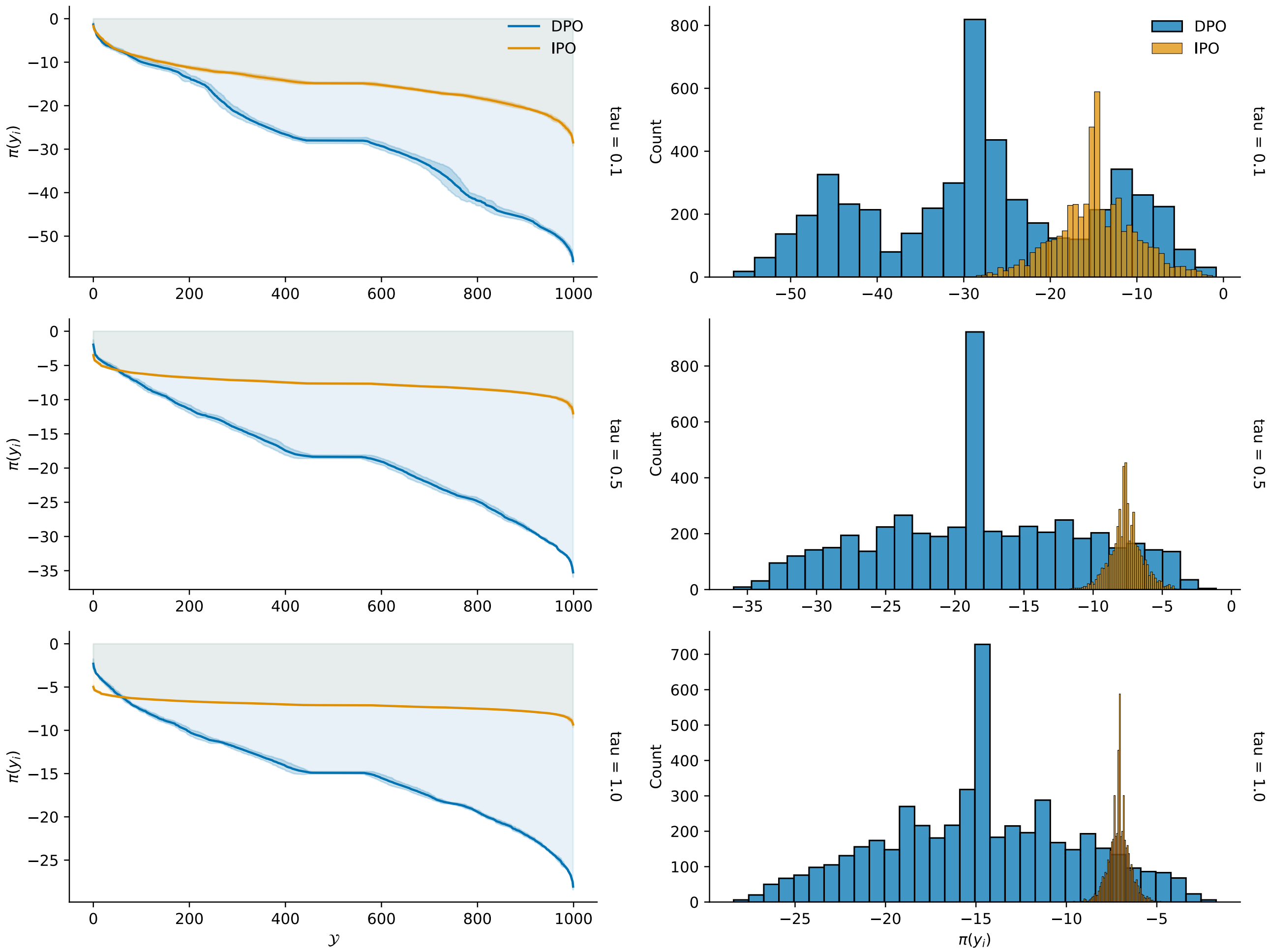
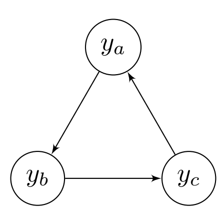


Figure 5: Comparison between the final policies of `IPO` and `DPO` in terms of log-probabilities (left) and empirical counts (right) for $\mathcal{D}_4$.

(i.e., converge to $\pi^\star = \pi_{\text{ref}}$) as the number of i.i.d. samples increases. However `IPO` makes a much better use of the regularization, and for all values of τ converges in a fraction of the samples of `DPO`.

C.2 Uniqueness of $\mathcal{D}_1$ and $\mathcal{D}_2$ in Section 5.4

Recall that in Section 5.4, the action space is defined as $\mathcal{Y} = \{y_a, y_b, y_c\}$ with 3 actions, and that to construct the dataset we sample each unique action pair $\{(y_a, y_b, I(y_a, y_b)), (y_b, y_c, I(y_b, y_c)), (y_c, y_a, I(y_c, y_a))\}$. The output of this process (i.e., the dataset) can be represented as a directed graph with three nodes and three edges, where an edge from y to y' represent having sampled a preference of y over y'. It is simple to see that $\mathcal{D}_2$ (the cycle) is one of the possible outcomes



.

Given this representation, we can also easily see that changing the outcome of one of the random preferences $I(y, y')$ is equivalent to inverting the direction of an edge. For example, one such outcome (and the one we use in the main paper) would be $\mathcal{D}_1$

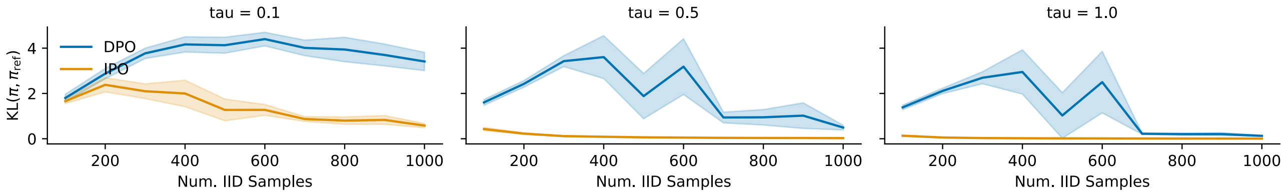
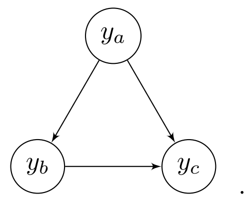


Figure 6: Comparison between the KL distance of `IPO` and `DPO` from the uniform policy for $\mathcal{D}_4$.



.

However, we can easily see that any other modifications applied to either $\mathcal{D}_1$ or $\mathcal{D}_2$ can be removed by reordering. For example if we invert the edge between y_a and y_b and then re-label the nodes we have