Reverse-KL Reinforcement Learning Can Sample From Multiple Diverse Modes

Anonymous Author(s)
Affiliation
Address
email

Abstract

It is commonly believed that optimizing the reverse KL divergence result in "mode seeking", while optimizing forward KL result in "mass covering", with the latter being preferred if the goal is to sample from multiple diverse modes. We show—mathematically and empirically—that this intuition does not necessarily transfer well to doing reinforcement learning with reverse/forward KL regularization (as used with verifiable rewards, human feedback, and reasoning tasks). Instead, the choice of reverse/forward KL determines the *family* of target distributions which maximizes the objective, while mode coverage depends primarily on other factors, such as regularization strength. Further, we show commonly used settings such as low regularization strength and equal verifiable rewards tend to specify uni-modal target distributions, meaning the optimization objective is *by construction* non-diverse. Finally, we leverage these insights to construct a simple, theoretically principled algorithm which explicitly optimizes for a multi-modal target distribution that puts high probability over *all* high quality samples. We show this works to post-train LLMs to have high solution diversity with both forward and reverse KL, when using either the forward or reverse KL naively fails.

1 Introduction

Reinforcement Learning (RL) is now the predominant way of post-training Large Language Models (LLMs) to be proficient at various tasks and to do reasoning. At its core, the problem involves solving a KL-regularized reward maximization problem, where the LLM is trained to maximize an external reward, while preserving "closeness" to a base policy as measured by KL divergence. However, it has been found that RL tends to collapse the policy distribution, leading to a lack of diversity in the trained model (Kirk et al., 2023). A number of works have sought out to address this, such as by explicitly incorporating diversity rewards (Li et al., 2025), changing the KL regularizer (Wang et al., 2023), or selecting data in a way that promotes diversity (Lanchantin et al., 2025).

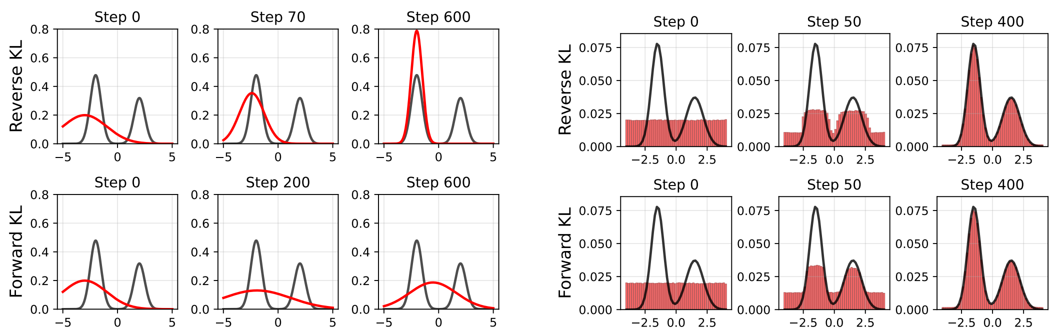
In this work, we take a step back and ask a more fundamental question: *does the objective we are optimizing have a solution that is diverse?* In other words, if we perfectly solve the RL problem in the limit of compute, will we get the solution that we want? We find that with current set-ups, the answer is often "no". Concretely, we theoretically show the properties of the solution distribution depend on an interplay between the reward function, reference / base model, and the regularization strength. Interestingly, the properties are *predictable*, and we can prove that under typical settings (such as weak KL regularization and using the same reward for all correct answers) the optimal solution is *by construction* non-multimodal. Using the same insights, we can derive conditions under which we *can* achieve diverse outcomes, by specifying a *different*, multi-modal target distribution to optimize towards. This is principled, requires minimal changes to the KL-regularized RL objective, and uses no additional information beyond the reward and reference model.

Our contributions are as follows,

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

1. We show RL with different KL-regularization have different *families* of solution distributions, with levels of mode coverage depending primarily on regularization strength and reward shapes, rather than the type of KL (potentially contrary to commonly beliefs).
2. We show that with typically used RL hyperparameters, the solution distribution RL optimizes towards is often *by definition* uni-modal, regardless of the type of regularizer, making diversity collapse a natural consequence of solving the RL problem.
3. We derive conditions required for multi-modal solution distributions, and use this insight to construct a simple and principled RL algorithm that directly optimizes for multi-modality, without the need for any external diversity signals.

2 The Kullback-Leibler (KL) Divergence



(a) Restrictive (Gaussian) approximation

(b) Flexible (categorical) approximation

Figure 1: Illustration of how the choice of approximate distribution family affects KL optimization. With a restrictive approximate distribution (e.g. two-parameter Gaussian), KL exhibit the typical "mode seeking" and "mass covering" characteristics. This intuition does not necessarily hold for flexible distributions (e.g. independent categoricals, language models).

The Kullback–Leibler (KL) divergence (Kullback and Leibler, 1951) measures the discrepancy between two probability distributions. In machine learning, it is commonly used in variational inference (VI), where minimizing the KL divergence enables a tractable variational distribution q to approximate an intractable posterior p (Jordan et al., 1999; Blei et al., 2017). Beyond VI, KL divergence plays an important role in RL (Chan et al., 2022), such as in regularizing an LLM policy from drifting too far from a pretrained base model (Ouyang et al., 2022).

Following Murphy (2012), we refer to $D_{KL}(q||p) = \mathbb{E}_q[\log q(y) - \log p(y)]$ as the *reverse KL divergence*, and $D_{KL}(p||q) = \mathbb{E}_p[\log p(y) - \log q(y)]$ as the *forward KL divergence*. Reverse KL is often described as "mode seeking", avoiding mass where p is small (Figure 1a, top), while forward KL is often described as "mass covering", putting mass anywhere p has mass (Figure 1a, bottom). These intuitions hold *if* the variational family is not sufficiently expressive and the objective cannot be fully optimized (Bishop and Nasrabadi, 2006; Murphy, 2012). With a flexible family, however, optimizing either KL to optimum can well-approximate a complex posterior (Figure 1b).

3 KL-Regularized Reward Maximization

KL-regularized reward maximization aims to (i) maximize a reward function $R : \mathcal{Y} \to \mathbb{R}$, mapping from samples to a scalar outcome (e.g. improve human preference), while (ii) keeping the policy π_θ close to a reference distribution π_{ref} (e.g. maintain grammatical coherence). The objective is,

$$J(\pi_\theta) = \mathbb{E}_{\pi_\theta(y)}[R(y)] - \beta\, D\big(\pi_\theta, \pi_{\text{ref}}\big)\,, \tag{1}$$

where $D(\cdot, \cdot)$ denotes a divergence between the policy and reference distributions. For brevity, we consider the unconditional generation problem where the policy models distribution $\pi_\theta(y)$. Note that the problem is the same in the case of conditional generation (e.g. question answering), where the objective is simply defined over the conditional distribution $\pi_\theta(y|x)$.

In this section, we consider the *solution / target distribution* of KL-regularization reward maximization—i.e. the distribution which maximizes the objective. The central question is:

If we perfectly solve the RL problem at the limit of compute, what does the solution policy distribution look like?

3.1 Solution of the Reverse KL Regularized Objective

The most common KL-regularized policy gradient objective uses the *reverse KL divergence*,

$$J_\beta(\pi_\theta) = \mathbb{E}_{\pi_\theta(y)}[R(y)] - \beta\, D_{KL}\big(\pi_\theta || \pi_{\text{ref}}\big)\,. \tag{2}$$

A number previous works have discussed the solution / optimal distribution of this optimization problem (Korbak et al., 2022; Go et al., 2023; Rafailov et al., 2023), which we note again below.

Remark 3.1. *The optimal solution to the reverse-KL regularized reward maximization problem,* $\arg\max_{\pi_\theta} J_\beta(\pi_\theta)$*, is given by the solution distribution* $\pi^* = G_\beta$*,*

$$G_\beta(y) = \frac{1}{\zeta}\, \pi_{ref}(y) \exp\left(\frac{R(y)}{\beta}\right), \tag{3}$$

where $\zeta = \int \pi_{ref}(y) \exp(R(y)/\beta)\, dy$ *is the normalizing constant.*

Proof. Appendix B.1. □

3.2 Gradient of the Reverse KL Regularized Objective

Remark 3.1 tells us the solution distribution maximizing Equation 2 is $\pi_\theta = G_\beta$. However, it may not be immediately obvious *how* the gradient of Equation 2, $\nabla_\theta J_\beta(\pi_\theta)$, moves π_θ toward G_β. We analyze this to understand the behaviour of optimizing Equation 2.

Remark 3.2. *The gradient of Equation 2 is a gradient of the reverse KL divergence between the current policy* π_θ *and the target distribution* G_β*,*

$$\nabla_\theta\, D_{KL}\big(\pi_\theta \,||\, G_\beta\big) \propto -\nabla_\theta\, J_\beta(\pi_\theta)\,. \tag{4}$$

Proof. Appendix B.2. □

Therefore, optimizing Equation 2 to optimum with a flexible policy distribution will give us G_β.

Main Takeaway

Maximizing the reverse-KL regularized RL objective J_β (Equation 2) is equivalent to doing distribution matching by minimizing a reverse KL toward the target distribution G_β (Equation 3).

3.3 Solution of the Forward KL Regularized Objective

Alternatively, we can regularized the reward maximizaztion with a forward KL penalty,

$$J_{\text{fwd}}(\pi_\theta) = \mathbb{E}_{\pi_\theta(y)}[R(y)] - \beta\, D_{KL}\big(\pi_{\text{ref}} || \pi_\theta\big)\,. \tag{5}$$

A number of recent works have used forward KL regularization. Some are motivated explicitly by the "mass covering" intuition of the forward KL (Wang et al., 2023), while others—such as GRPO (Shao et al., 2024)—may have incidentally estimated the forward KL, despite being motivated by using the reverse KL (Tang and Munos, 2025).

Remark 3.3. *The optimal solution to the forward-KL regularized reward maximization problem,* $\arg\max_{\pi_\theta} J_{fwd}$*, is given by the solution distribution:*

$$G_{fwd}(y) = \frac{\beta\, \pi_{ref}(y)}{\Lambda - R(y)}\,, \quad \Lambda > \max_y R(y)\,, \tag{6}$$

where Λ *is chosen such that* G_{fwd} *is a valid probability distribution.*

Proof. Appendix B.3. □

Notably, Equation 6 yields a *completely different* distribution family from the reverse KL case (Equation 3). Unlike the reverse case, it does not have a closed form solution and requires solving Λ for each value of β. Moreover, while the gradient of the reverse-KL regularized objective is itself a reverse KL gradient (Remark 3.2), the gradient of the forward-KL regularized objective (Equation 5) is *not* a forward KL gradient. Consequently, optimizing Equation 5 does not necessarily inherit the properties of a "forward KL gradient", such as common intuitions about "mass seeking". While it may still have desirable properties, a deeper analysis of this gradient is left for future work.

Main Takeaway

Maximizing the forward-KL regularized objective J_{fwd} (Equation 5) does not yield a forward-KL gradient, so its behaviour cannot be naively equated to forward-KL optimization.

3.4 Computing a Forward KL Gradient

If not Equation 5, what *is* the forward KL toward the target G_β, then?

Remark 3.4. *The gradient of the forward KL divergence between policy π_θ and target G_β is,*

$$\nabla_\theta D_{KL}\big(G_\beta || \pi_\theta\big) = -\mathbb{E}_{G_\beta}\big[\nabla_\theta \log \pi_\theta(y)\big]\,. \tag{7}$$

Proof. See Appendix B.4. □

We see that optimizing the forward KL gradient amounts to doing maximum likelihood / supervised fine-tuning on trajectories sampled from the target distribution G_β. This is generally intractable as it requires sampling from G_β, which we do not have. Nevertheless, this does give some insights into algorithms such as RAFT (Dong et al., 2023; Xiong et al., 2025) which filter high-reward trajectories to do maximum likelihood. One can interpret filtering as constructing an approximate target distribution (that puts high mass over high-reward regions) and optimizing a forward KL.

3.5 Both KL Regularization Have Multimodal Solution Distributions

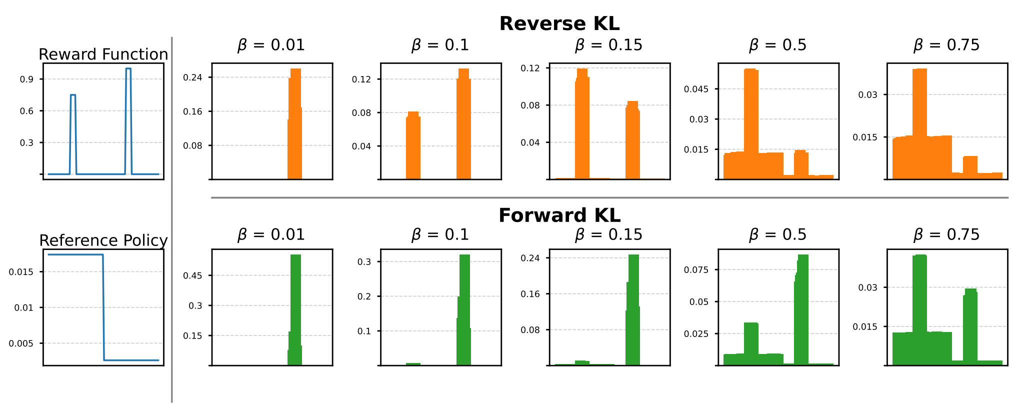


Figure 2: Final policy distribution from optimizing a reverse/forward KL regularized reward maximization objective, given the same reward function, reference policy, across a range of regularization strengths (β). Note that both KLs can lead to multi-modal solution distributions.

It is worth briefly noting that the solution distributions for both reverse (Equation 3) and forward (Equation 6) KL regularization *can* be multi-modal. We show this in a didactic example in Figure 2, where given the same reward function containing two high-reward modes, and a reference policy with support over the first half of the token space, optimizing the reverse and forward KL objectives lead to a wide variety of solutions that depend on the regularization coefficient β. Both KLs have settings of β that induce multi-modal solution distributions. We analyze the properties of the target distribution in the subsequent section, and return to the Figure 2 example in detail in Section 4.3.

4 Analysis of KL Regularized Optimal Distribution

We have seen in Section 3.5 that both *KL-regularized reward maximization objectives* can have multi-modal solutions, and that optimizing either the reverse or the forward *KL gradient* can lead to good approximations of the solution distribution, if done to optimum (Section 2). However, the shape of the solution distribution depend heavily on the reward, reference distribution, and regularization strength. This begs the central question of this section:

Is the solution we are optimizing for actually multi-modal?

Definition 4.1. (Informal) A solution distribution for KL-regularized reward maximization is "multi-modal" if all high-reward samples have high probability.

We will use Definition 4.1 as a loose working definition going forward. The central tools we will use in this section will be a *probability ratio* between two samples under a distribution. Intuitively, we want (i) high-reward samples to be much more probable than low-reward samples, and (ii) similarly high-reward samples to have similar high probabilities. We focus our analysis on the solution of the reverse-KL regularized objective (Equation 3), both for its clean form and because it is the most common way KL-regularized RL is formulated.

Proposition 4.2. *The (log) probability ratio between any two samples, y_1, y_2, under the optimal solution distribution for reverse-KL regularized RL, G_β, can be written in closed form,*

$$\log \frac{G_\beta(y_1)}{G_\beta(y_2)} = \log \frac{\pi_{ref}(y_1)}{\pi_{ref}(y_2)} + \frac{1}{\beta}\Big(R(y_1) - R(y_2)\Big)\,. \tag{8}$$

Proof. Because normalization constant ζ cancel out in ratios. See Appendix B.5. □

This means that we can exactly compute how likely one sample is relative to another in the *optimal final solution*, using *only* π_{ref} and the reward function R. We see there are a number of consequential insights about the objective we are optimizing for.

4.1 With equal supports, small reward differences lead to large probability differences

Remark 4.3. *For any two samples y_1 and y_2, if $\pi_{ref}(y_1) = \pi_{ref}(y_2)$, their probability ratio is:*

$$\log \frac{G_\beta(y_1)}{G_\beta(y_2)} = \frac{1}{\beta}\Big(R(y_1) - R(y_2)\Big)\,. \tag{9}$$

In words, for two samples that have the same probability under the reference distribution ("equal support"), the difference in their final log probabilities is simply the difference in their rewards, scaled by $1/\beta$. Smaller β exaggerates the difference between their log probability ratios. Note a *linear* difference in rewards result in an *exponential* difference in probabilities: for a 0.1 difference in rewards, and a commonly used $\beta = 1\text{e-}3$, the higher reward sample is 2.6×10^{43} times more likely in the solution distribution (Figure 3). This suggests for commonly used hyperparameter settings, the solution distribuiton is highly concentrated around its mode.

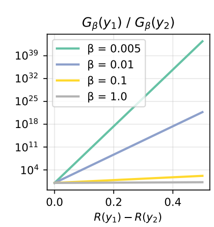


Figure 3

To build additional intuition and empirically validate the theory, we use a didactic example where we optimize a categorical distribution using KL regularized RL (details in Appendix C.1). We observe in Figure 4 that regularization strength β controls the difference in rewards, and below some threshold of regularization the solution policy becomes uni-modal.

4.2 With equal rewards, solution *never* prefers off-support samples

We now analyze the case where the correct solutions all have equal reward. This is a common set-up for the case of RL with verifiable reward (e.g. math), where a correct answer is usually given a reward of 1, and incorrect answers given reward of 0.

Remark 4.4. *For any two samples with the same reward, $R(y_1) = R(y_2)$, their probability ratio is:*

$$\log \frac{G_\beta(y_1)}{G_\beta(y_2)} = \log \frac{\pi_{ref}(y_1)}{\pi_{ref}(y_2)}\,. \tag{10}$$

In words, their relative probabilities in the solution is simply their relative probabilities in the reference distribution, and *do not depend on the KL-regularization strength β*. In other words, with identical rewards, RL only changes the relative probability between correct and incorrect answers, but not between on- and off-support correct answers. Setting a lower regularization strength β only

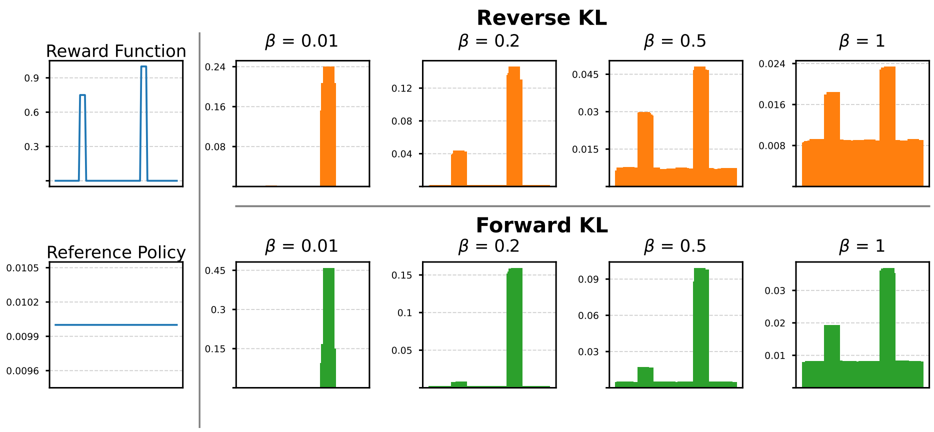


Figure 4: **(Left)** Reward function and reference distribution. **(Right)** Empirical distribution after optimizing the regularized RL for 1000 gradient steps, with reverse KL regularization (top) or forward KL regularization (bottom). Regularization strength β controls the difference in probability between differently rewarding regions, with a low β concentrating all mass on the highest reward mode.

encourages the correct answers to become relatively more likely, but *do not encourage more off-support answers*. Said another way, the **RL with equal verifiable reward objective by construction discourages off-support answers**.

We empirically verify this prediction in Figure 5. We see that the final policy distribution *never* favours the (equally correct) off-support mode. This is not an issue with exploration: we will see in the subsequent section and Figure 2 that with a small change in reward we can indeed optimize for a distribution that equally weights or even prefers the off-support solution. This also provides an explanation for methods that have demonstrated RL being able to discover abilities not present in the base model: they can only do so by changing the reference policy, for e.g. through periodic resets to the most recent online policy (Liu et al., 2025).

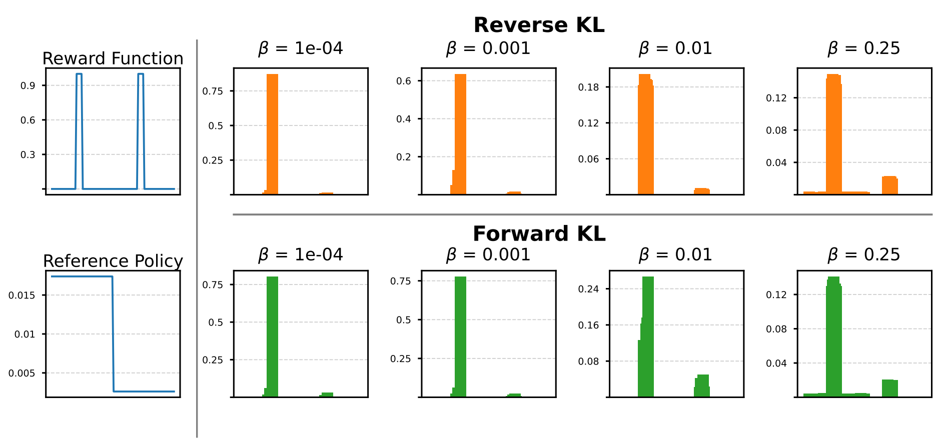


Figure 5: **(Left)** Reward function with identical reward for correct answers, and reference distribution with support over first half of token space. **(Right)** Empirical distribution after optimizing the regularized RL objective with reverse KL regularization (top) or forward KL regularization (bottom).

Main Takeaway

KL-regularized RL does not increase the probability of off-support samples relative to on-support ones as long as their rewards are the same. Lowering the KL regularization strength β has *no effect* on up-weighting off-support samples.

4.3 For unequal rewards *and* supports, regularization strength determines mode coverage

When two trajectories have different rewards and different probabilities under the reference policy, a unique setting of β will induce the two to have the same probability in the solution distribution.

Remark 4.5. *Two samples have the same probability in the target distribution if,*

$$R(y_2) - R(y_1) = \beta\big(\log \pi_{ref}(y_1) - \log \pi_{ref}(y_2)\big)\,. \tag{11}$$

This condition allow us to predict, given only the reward and reference policy, when two samples will have the same probabilities in the solution to the RL problem. As an example, we know in Figure 2 that the two high-reward modes have rewards 0.75 and 1.0, and reference policy probabilities of $\log \pi_{\text{ref}}(y_1) \approx -4.05$ and $\log \pi_{\text{ref}}(y_2) \approx -5.95$, respectively. This allows us to predict the setting of β which will "flip" the solution distribution's preference from the on-support mode to the off-support mode to be $(1-0.75)/(-4.05+5.95) \approx 0.132$. Indeed, we see in Figure 2 for the reverse KL case, the preference between the two modes switch as we move from $\beta = 0.15$ to $\beta = 0.10$. This is the true role of the regularization coefficient β: it is a knob that decides between picking higher rewarding, off-support solutions, vs. lower rewarding, on-support solutions.

5 Directly Optimizing for Multi-Modality

Having identified the various failure cases of the KL-regularized RL objective (Section 4), and the role of regularization in balancing reward differences (Section 4.3), we now turn to the question:

> *Can we construct an objective such that when optimized, naturally give rise to a multi-modal solution distribution?*

Indeed, Remark 4.5 already gives us the ingredients required to do this. Below, we derive a simple procedure which will ensure we are optimizing for a solution that puts *equal* probabilities on all high-quality samples (per Definition 4.1). Concretely, we construct the augmented reward function,

$$\bar{R}(y) = \begin{cases} R(y) & \text{if } R(y) < \tau, \\ R(z) + \beta\big(\log \pi_{\text{ref}}(z) - \log \pi_{\text{ref}}(y)\big) & \text{if } R(y) \geq \tau. \end{cases} \tag{12}$$

where $\tau \leq \max_y R(y)$ is some threshold for "goodness", and z is a fixed "anchor" sample chosen from the set of high-quality samples. We can pick it to be $z = \arg\max_y \pi_{\text{ref}}(y)$ where $R(y) \geq \tau$. Because we are choosing the "anchor" sample to be from a high-reward mode, we will colloquially refer to this approach as "*mode anchoring*".

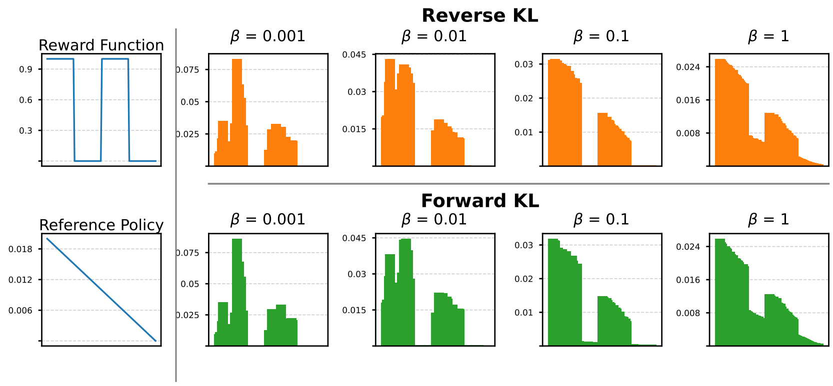
Intuitively, the augmented reward function induces a new *target distribution* with *uniform* high density over regions where the reward is above threshold τ, and stays close to the reference π_{ref} in regions where the reward is below the threshold. We see in the Figure 6a example that naive KL-regularized RL lead to solutions that heavily favour the left mode (which is more on-support), regardless of the choice of β or KL. On the other hand, using mode-anchored reward augmentation result in solutions that put *equal* high mass over *all* high quality samples (Figure 6b). Interestingly, while the theory is developed for the reverse-KL regularized case, we find that it also helps the forward-KL regularized optimization (Fig 6b, bottom row), albeit with some unexpected behaviour at higher β's.

Remark 5.1. *Optimizing the reverse-KL regularized RL objective with the augmented reward function $\bar{R}$ yields the following solution distribution,*

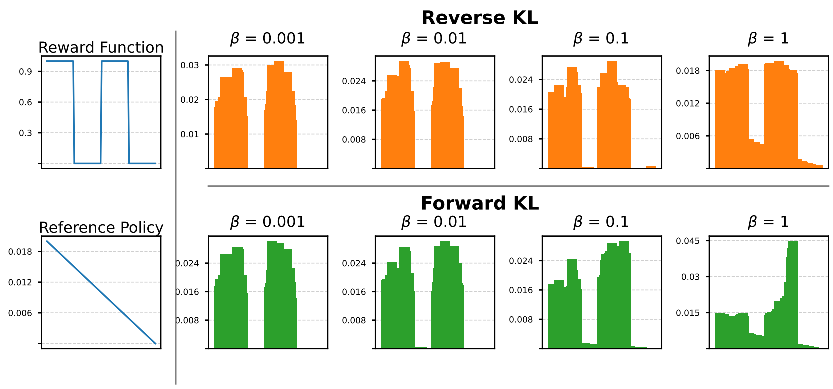
$$\bar{G}_\beta(y) \propto \begin{cases} \pi_{ref}(y) \exp\left(\frac{R(y)}{\beta}\right) & \textit{if } R(y) < \tau, \\ \pi_{ref}(z) \exp\left(\frac{R(z)}{\beta}\right) & \textit{if } R(y) \geq \tau. \end{cases} \tag{13}$$

Proof. Appendix B.6. □

This formally shows the target will have uniformly high density proportional to $\pi_{\text{ref}}(z)\exp(R(z)/\beta)$ for all samples if their original reward $R(y)$ is above threshold τ. If we pick z to be likely under π_{ref}, e.g. $z = \arg\max_y \pi_{\text{ref}}(y)$, we can also show these samples will have the highest probabilities in the solution distribution.



(a) Vanilla KL-regularized reward maximization



(b) Reward maximization with mode-anchoring augmented rewards (`MARA`)

Figure 6: Our approach vs naive reverse or forward KL

5.1 The 1-2 Task for LLM Diversity

We further demonstrate our method in a more realistic LLM post-training task. Specifically, we ask the LM to generate a uniform random integer that is either 1 or 2 (Hopkins et al., 2023), as illustrated in Figure 9. We train a `Qwen2.5 3B` model with KL-regularized RL, giving it a reward of 1 for getting the answer correct (if it produces “1“ or “2“ in XML format), and a reward of 0 otherwise.

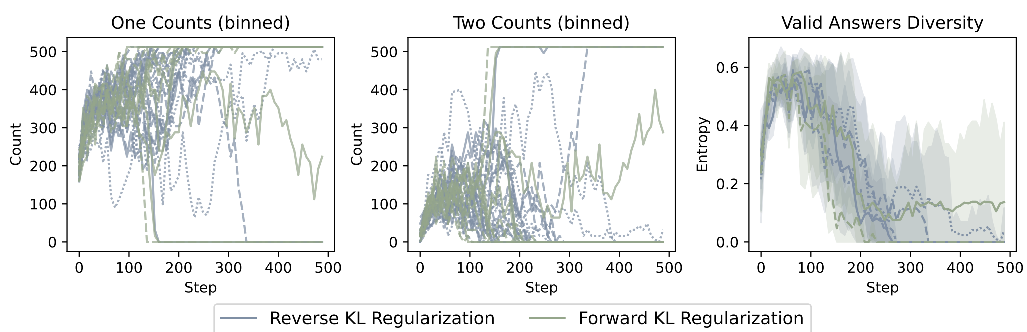
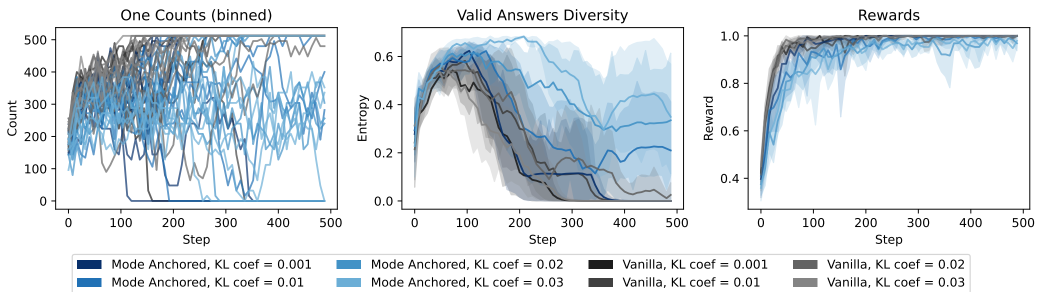


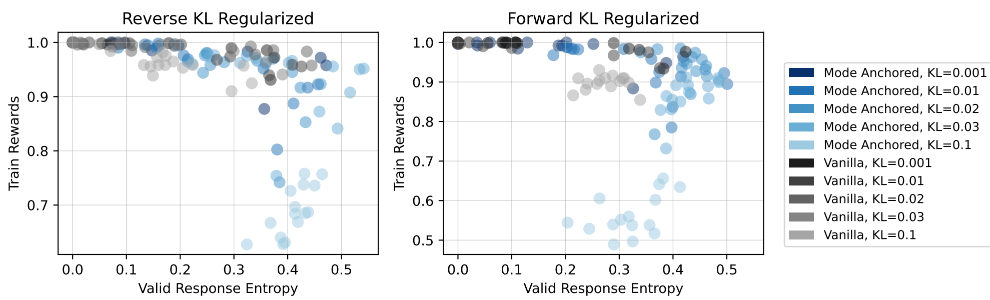
Figure 7: Training outcomes using vanilla RL. **(Left, Middle)** Policy's empirical distribution over valid answers for runs that reached high rewards (counts binned over 8 consecutive training batches), across a range of regularization coefficients (β). **Right** Diversity of the valid answers over the course of training, measured as the entropy of the Bernoulli distribution over answers of 1's and 2's.

Naive RL result in mode collapse We run KL-regularized RL for a range of KL coefficients (β) and multiple random seeds. Most runs are able to optimize the reward well and get a reward of ~ 1. Figure 7 shows the distribution of correctly formatted 1's and 2's the LM generates over the course of training. We observe that for the 34 runs shown, all but one collapsed into a generating only a single answer as a result of RL. This is true for both reverse and forward KL regularization, and most runs collapsed into generating 1's, which has higher likelihood under the base policy.

Online RL with Mode Anchored Reward Augmentation We now apply the mode anchoring idea into an efficient online algorithm for LLM fine-tuning. While we could first run an optimization for $\arg\max_y \pi_{\text{ref}}(y)$ (where $R(y) \geq \tau$), we opt to simply use the within-batch *most likely correct sample* under the reference policy as the anchor trajectory z. This does introduce bias as the anchor is different across batches, but we will see below that this nevertheless improves diversity. We refer to the algorithm as ***M**ode **A**nchored **R**eward **A**ugmentation* (**MARA**, Algorithm 1).



(a) Valid answer entropy and rewards



(b) Pareto front of reward (quality) and entrpy (diversity)

Figure 8: Our approach vs naive reverse or forward KL

MARA maximizes diversity while preserving quality We run KL-constrained RL with the same hyperparameters, only now with MARA. We see in Figure 8a that compared to vanilla RL (grey), MARA (blue) is able to preserve the diversity in the correct answers, with many runs learning to generate 1's and 2's with near uniform probability, while still correctly learning to generate with the correct format. Further, we can plot the pareto front of the different ways of training at various points of training, for different KL coefficients and averaged over seeds. We see in Figure 8b that for both reverse and forward KL regularization, MARA is able to match vanilla training in terms of format correctness, while exceeding vanilla training in terms of generation diversity.

6 Conclusion

The lesson of Artificial Intelligence over the past decade has been that with simple, sound, objectives, scaling compute and data will consistently out-perform ad-hoc, human-designed approaches. In this work, we provide an in-depth analysis of the properties of the KL-regularized RL objective, to provide understanding into whether this is the objective we are hoping to achieve. Using these insights, we also construct a simple alternative objective that directly optimizes for high multi-modal diversity, a feat that existing objectives are fundamentally unable to achieve.

References

Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.

David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877, 2017.

Alan Chan, Hugo Silva, Sungsu Lim, Tadashi Kozuno, A Rupam Mahmood, and Martha White. Greedification operators for policy optimization: Investigating forward and reverse kl divergences. *Journal of Machine Learning Research*, 23(253):1–79, 2022.

Daixuan Cheng, Shaohan Huang, Xuekai Zhu, Bo Dai, Wayne Xin Zhao, Zhenliang Zhang, and Furu Wei. Reasoning with exploration: An entropy perspective. *arXiv preprint arXiv:2506.14758*, 2025.

John Joon Young Chung, Vishakh Padmakumar, Melissa Roemmele, Yuqian Sun, and Max Kreminski. Modifying large language model post-training for diverse creative writing. *arXiv preprint arXiv:2503.17126*, 2025.

Ganqu Cui, Yuchen Zhang, Jiacheng Chen, Lifan Yuan, Zhi Wang, Yuxin Zuo, Haozhan Li, Yuchen Fan, Huayu Chen, Weize Chen, et al. The entropy mechanism of reinforcement learning for reasoning language models. *arXiv preprint arXiv:2505.22617*, 2025.

Xingyu Dang, Christina Baek, Kaiyue Wen, Zico Kolter, and Aditi Raghunathan. Weight ensembling improves reasoning in language models. *arXiv preprint arXiv:2504.10478*, 2025.

Hanze Dong, Wei Xiong, Deepanshu Goyal, Yihan Zhang, Winnie Chow, Rui Pan, Shizhe Diao, Jipeng Zhang, Kashun Shum, and Tong Zhang. Raft: Reward ranked finetuning for generative foundation model alignment. *arXiv preprint arXiv:2304.06767*, 2023.

Dongyoung Go, Tomasz Korbak, Germán Kruszewski, Jos Rozen, Nahyeon Ryu, and Marc Dymetman. Aligning language models with preferences through f-divergence minimization. *arXiv preprint arXiv:2302.08215*, 2023.

Jean-Bastien Grill, Florent Altché, Yunhao Tang, Thomas Hubert, Michal Valko, Ioannis Antonoglou, and Rémi Munos. Monte-carlo tree search as regularized policy optimization. In *International Conference on Machine Learning*, pages 3769–3778. PMLR, 2020.

Andre He, Daniel Fried, and Sean Welleck. Rewarding the unlikely: Lifting grpo beyond distribution sharpening. *arXiv preprint arXiv:2506.02355*, 2025.

Aspen K Hopkins, Alex Renda, and Michael Carbin. Can LLMs generate random numbers? evaluating LLM sampling in controlled domains. In *ICML 2023 Workshop: Sampling and Optimization in Discrete Space*, 2023. URL https://openreview.net/forum?id=Vhh1K9LjVI.

Edward J Hu, Moksh Jain, Eric Elmoznino, Younesse Kaddar, Guillaume Lajoie, Yoshua Bengio, and Nikolay Malkin. Amortizing intractable inference in large language models. *arXiv preprint arXiv:2310.04363*, 2023.

Mete Ismayilzada, Antonio Laverghetta Jr, Simone A Luchini, Reet Patel, Antoine Bosselut, Lonneke van der Plas, and Roger Beaty. Creative preference optimization. *arXiv preprint arXiv:2505.14442*, 2025.

Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233, 1999.

Robert Kirk, Ishita Mediratta, Christoforos Nalmpantis, Jelena Luketina, Eric Hambro, Edward Grefenstette, and Roberta Raileanu. Understanding the effects of rlhf on llm generalisation and diversity. *arXiv preprint arXiv:2310.06452*, 2023.

Tomasz Korbak, Hady Elsahar, Germán Kruszewski, and Marc Dymetman. On reinforcement learning and distribution matching for fine-tuning language models with no catastrophic forgetting. *Advances in Neural Information Processing Systems*, 35:16203–16220, 2022.

Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

Jack Lanchantin, Angelica Chen, Shehzaad Dhuliawala, Ping Yu, Jason Weston, Sainbayar Sukhbaatar, and Ilia Kulikov. Diverse preference optimization. *arXiv preprint arXiv:2501.18101*, 2025.

Tianjian Li, Yiming Zhang, Ping Yu, Swarnadeep Saha, Daniel Khashabi, Jason Weston, Jack Lanchantin, and Tianlu Wang. Jointly reinforcing diversity and quality in language model generations. *arXiv preprint arXiv:2509.02534*, 2025.

Mingjie Liu, Shizhe Diao, Ximing Lu, Jian Hu, Xin Dong, Yejin Choi, Jan Kautz, and Yi Dong. Prorl: Prolonged reinforcement learning expands reasoning boundaries in large language models. *arXiv preprint arXiv:2505.24864*, 2025.

Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. *Advances in Neural Information Processing Systems*, 36:53728–53741, December 2023. URL https://papers.nips.cc/paper_files/paper/2023/hash/a85b405ed65c6477a4fe8302b5e06ce7-Abstract-Conference.html.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Yunhao Tang and Rémi Munos. On a few pitfalls in kl divergence gradient estimation for rl. *arXiv preprint arXiv:2506.09477*, 2025.

Daniil Tiapkin, Nikita Morozov, Alexey Naumov, and Dmitry P Vetrov. Generative flow networks as entropy-regularized rl. In *International Conference on Artificial Intelligence and Statistics*, pages 4213–4221. PMLR, 2024.

Chaoqi Wang, Yibo Jiang, Chenghao Yang, Han Liu, and Yuxin Chen. Beyond reverse kl: Generalizing direct preference optimization with diverse divergence constraints. *arXiv preprint arXiv:2309.16240*, 2023.

Shenzhi Wang, Le Yu, Chang Gao, Chujie Zheng, Shixuan Liu, Rui Lu, Kai Dang, Xionghui Chen, Jianxin Yang, Zhenru Zhang, et al. Beyond the 80/20 rule: High-entropy minority tokens drive effective reinforcement learning for llm reasoning. *arXiv preprint arXiv:2506.01939*, 2025.

Wei Xiong, Jiarui Yao, Yuhui Xu, Bo Pang, Lei Wang, Doyen Sahoo, Junnan Li, Nan Jiang, Tong Zhang, Caiming Xiong, and Hanze Dong. A minimalist approach to llm reasoning: from rejection sampling to reinforce. *arXiv preprint arXiv:2504.11343*, 2025.

A Related Work

Training for diversity Wang et al. (2023) generalizes the DPO objective (Rafailov et al., 2023) from reverse-KL regularized to a more general class of f-divergence regularizers, with the key motivation being that reverse-KL can be mode-seeking, therefore reduce diversity. They do not explore the effect of reward function or regularization coefficient β, which our work examines. Diverse DPO Lanchantin et al. (2025) and variants (Chung et al., 2025; Ismayilzada et al., 2025) encourage diversity in preference learning by selecting diverse positives/negatives. Most closely related to our reward augmentation approach is He et al. (2025), which uses a rank based "unlikeliness reward" by ranking the in-batch samples based on their likelihood under the current policy, and penalize the most likely samples. Similarly related is Li et al. (2025), which use an external model to evaluate diversity (via a semantic classifier) and use the diversity metric to modify the reward. We do not require an external model to evaluate diversity.

More distantly, Dang et al. (2025) found that combining weights of earlier and later checkpoints can improve pass@k performance—a loose measure of diversity (albeit over both correct and incorrect answers). GFlowNets also provide diversity-seeking policies that sample proportionally to reward, albeit they use different algorithms than the KL-regularized policy gradient which is the most commonly used algorithm for LM post-training (Hu et al., 2023; Tiapkin et al., 2024).

Entropy and reasoning in RL We can view mode collapse in solutions as a collapse in the entropy of the *trajectory* distribution. This is related (but not identical) to token entropy. A growing line of empirical work do tie together entropy, exploration, and reasoning in LLMs. Cui et al. (2025) notes entropy collapses during RL. Cheng et al. (2025) incorporates an entropy term in the advantage to encourage better reasoning. Wang et al. (2025) show that focusing gradient updates on a minority of high-entropy tokens ("forking tokens") can improve reasoning.

B Mathematical Derivations

B.1 Target Distribution of Reverse-KL Reward Maximization

Proof of Remark 3.1 We want to find the distribution which maximizes the objective from equation 2,

$$\arg\max_{\pi_\theta} J_\beta(\pi_\theta) = \arg\max_{\pi_\theta} \mathbb{E}_{\pi_\theta(y)}[R(y)] - \beta\, D_{KL}\big(\pi_\theta || \pi_{\text{ref}}\big) \tag{14}$$

We can re-write Equation 2 by re-arranging terms, note for notation brevity we denote $g_\beta(y) = \pi_{\text{ref}}(y)\exp\big(\frac{R(y)}{\beta}\big)$,

$$\begin{aligned}
J_\beta(\pi_\theta) &= \mathbb{E}_{\pi_\theta(y)}[R(y)] - \beta\, D_{KL}\big(\pi_\theta || \pi_{\text{ref}}\big)\,, && (15)\\
&= \mathbb{E}_{\pi_\theta(y)}\Big[R(y) - \beta\big(\log \pi_\theta(y) - \log \pi_{\text{ref}}(y)\big)\Big]\,, && (16)\\
&= -\beta\, \mathbb{E}_{\pi_\theta(y)}\Big[\log \pi_\theta(y) - \big(\frac{R(y)}{\beta} + \log \pi_{\text{ref}}(y)\big)\Big]\,, && (17)\\
&= -\beta\, \mathbb{E}_{\pi_\theta(y)}\Big[\log \pi_\theta(y) - \log \pi_{\text{ref}}(y)\exp\big(\frac{R(y)}{\beta}\big)\Big]\,, && (18)\\
&= -\beta\, \mathbb{E}_{\pi_\theta(y)}\Big[\log \pi_\theta(y) - \log g_\beta(y) + \log\zeta - \log\zeta\Big]\,, && (19)\\
&= -\beta\, \mathbb{E}_{\pi_\theta(y)}\Big[\log \pi_\theta(y) - \log G_\beta(y)\Big] + \beta\log\zeta\,, && (20)\\
&= -\beta D_{KL}\Big(\pi_\theta || G_\beta\Big) + \beta\log\zeta\,. && (21)
\end{aligned}$$

It is easy to see that the above is maximized when $D_{KL}\big(\pi_\theta || G_\beta\big) = 0$, which is when the policy is the target distribution, $\pi_\theta = G_\beta$.

B.2 Gradient of Reverse-KL Reward Maximization

Proof of Remark 3.2 From Appendix B.1, we have the identity,

$$-\frac{1}{\beta}J_\beta(\pi_\theta) = D_{KL}\big(\pi_\theta||G_\beta\big) - \log\zeta\,. \tag{22}$$

We can easily show that the gradient is,

$$\nabla_\theta\Big(-\frac{1}{\beta}J_\beta(\pi_\theta)\Big) = \nabla_\theta\, D_{KL}\big(\pi_\theta||G_\beta\big) - \nabla_\theta \log\zeta\,, \tag{23}$$

$$= \nabla_\theta\, D_{KL}\big(\pi_\theta||G_\beta\big)\,. \tag{24}$$

In other words, they are the same up to constant $-\beta$,

$$\nabla_\theta J_\beta(\pi_\theta) = -\beta\,\nabla_\theta D_{KL}\big(\pi_\theta||G_\beta\big)\,. \tag{25}$$

B.3 Target Distribution of Forward-KL Reward Maximization

Via calculus of variations. See Grill et al. (2020); Tang and Munos (2025) for the same result.

B.4 Gradient of the forward KL

The gradient of the forward KL between the policy π_θ and the target distribution G_β is,

$$\nabla_\theta\, D_{KL}\big(G_\beta||\pi_\theta\big) = \nabla_\theta \mathbb{E}_{G_\beta}\big[\log G_\beta(y) - \log\pi_\theta(y)\big]\,, \tag{26}$$

$$= \mathbb{E}_{G_\beta}\big[\nabla_\theta\big(\log G_\beta(y) - \log\pi_\theta(y)\big)\big]\,, \tag{27}$$

$$= -\mathbb{E}_{G_\beta}\big[\nabla_\theta \log\pi_\theta(y)\big]\,. \tag{28}$$

B.5 Probability Ratio Under Optimal Target Distribution

Proof of Proposition 4.2 For any two samples, y_1 and y_2, their probability ratio under the target distribution is given by,

$$\frac{G_\beta(y_1)}{G_\beta(y_2)} = \frac{g_\beta(y_1)}{\zeta}\frac{\zeta}{g_\beta(y_2)} = \frac{g_\beta(y_1)}{g_\beta(y_2)}\,, \tag{29}$$

which only require the unnormalized likelihood as the normalization constant ζ cancel out. Expanding the terms, we can write the log likeilhood ratio in closed form,

$$\log\frac{G_\beta(y_1)}{G_\beta(y_2)} = \log\pi_{\text{ref}}(y_1)\exp\big(\frac{R(y_1)}{\beta}\big) - \log\pi_{\text{ref}}(y_2)\exp\big(\frac{R(y_2)}{\beta}\big)\,, \tag{30}$$

$$= \log\frac{\pi_{\text{ref}}(y_1)}{\pi_{\text{ref}}(y_2)} + \frac{1}{\beta}\Big(R(y_1) - R(y_2)\Big)\,. \tag{31}$$

B.6 Solution distribution after reward augmentation

Proof of Remark 5.1 We have established already in Appendix B.1 that the solution distribution of reward maximization with reverse KL regularization is,

$$G_\beta(y) \propto \pi_{\text{ref}}(y)\exp\big(\frac{R(y)}{\beta}\big)\,. \tag{32}$$

We now plug in the augmented reward function,

$$\bar{R}(y) = \begin{cases} R(y) & \text{if } R(y) < \tau, \\ R(z) + \beta\big(\log\pi_{\text{ref}}(z) - \log\pi_{\text{ref}}(y)\big) & \text{if } R(y) \geq \tau, \end{cases} \tag{33}$$

which gives us the augmented solution distribution,

$$\bar{G}_\beta(y) \propto \pi_{\text{ref}}(y)\exp\big(\frac{\bar{R}(y)}{\beta}\big)\,. \tag{34}$$

In the $R(y) < \tau$ case, $\bar{R}(y) = R(y)$, and there is no change to the (unnormalized) likelihood. In the $R(y) \geq \tau$ case,

$$\log \pi_{\text{ref}}(y) \exp\big(\frac{\bar{R}(y)}{\beta}\big) = \log \pi_{\text{ref}}(y) + \frac{1}{\beta}\bar{R}(y)\,, \tag{35}$$

$$= \log \pi_{\text{ref}}(y) + \frac{1}{\beta}\Big(R(z) + \beta\big(\log \pi_{\text{ref}}(z) - \log \pi_{\text{ref}}(y)\big)\Big)\,, \tag{36}$$

$$= \frac{R(z)}{\beta} + \log \pi_{\text{ref}}(y) + \log \pi_{\text{ref}}(z) - \log \pi_{\text{ref}}(y) \tag{37}$$

$$= \frac{R(z)}{\beta} + \log \pi_{\text{ref}}(z)\,. \tag{38}$$

Therefore we see in the $R(y) \geq \tau$ case we have,

$$\pi_{\text{ref}}(y) \exp\big(\frac{\bar{R}(y)}{\beta}\big) = \pi_{\text{ref}}(z) \exp\big(\frac{R(z)}{\beta}\big)\,. \tag{39}$$

C Additional Experimental Details

C.1 Didactic Experiments

We construct our didactic experiment as a vector of size 100 (akin to a "token space" with 100 tokens). We initialize a categorical distribution over this token space whose logits are all 0's (i.e. uniform distribution over all tokens). Given some reward function and reference distribution defined over this space, we optimize this categorical distribution with the KL-regularized policy gradient for 1000 gradient steps in PyTorch with Adam optimizer, with learning rate 5e-3 and batch size 32.

C.2 The 1-2 Task

Prompt

`Uniformly randomly generate an integer that is either 1 or 2. Respond strictly in this format: <think>Your internal reasoning</think><answer>1 or 2</answer>`

Example Generation

`Let me decide randomly. <think></think><answer>1 </answer><|endoftext|>`

Figure 9: The 1-2 task to test output distribution of LMs.

D More Method Details

Algorithm 1 Mode Anchored Reward Augmentation (MARA)

Given initial policy π_θ, reference distribution π_{ref}, reward function R, and regularization coefficient β
Set threshold for good answers: $\tau \in \mathbb{R}$, $\tau \leq \max_y R(y)$
for each iteration **do**
  Sample batch of trajectories $\{y_i\}_{i=1}^N \sim \pi_\theta$
  Pick anchor trajectory: $y_{\text{anch}} = \arg\max_{y_i} \pi_{\text{ref}}(y_i)$, s.t. $R(y_i) \geq \tau$
  for each y_i in batch **do**
    if $R(y_i) \geq \tau$ **then**
      $\bar{r}_i = R(y_{\text{anch}}) + \beta\big(\log \pi_{\text{ref}}(y_{\text{anch}}) - \log \pi_{\text{ref}}(y_i)\big)$
    else
      $\bar{r}_i = R(y_i)$
    end if
  end for
  Estimate KL-regularized policy gradient:

$$\tilde{J} \;=\; \frac{1}{N}\sum_{i=1}^{N} \hat{r}_i \nabla_\theta \log \pi_\theta(y_i) - \beta\Big(\log \pi_\theta(y_i) - \log \pi_{\text{ref}}(y_i)\Big)\nabla_\theta \log \pi_\theta(y_i)$$

  Update policy parameters θ with gradient estimate $\tilde{J}$
end for