RL with KL penalties is better viewed as Bayesian inference

Tomasz Korbak
University of Sussex
New York University
tomasz.korbak@gmail.com

Ethan Perez
New York University
perez@nyu.edu

Christopher L. Buckley
University of Sussex
c.l.buckley@sussex.ac.uk

Abstract

Reinforcement learning (RL) is frequently employed in fine-tuning large language models (LMs) to penalize them for undesirable features of generated sequences, such as offensiveness or harmfulness. In this paper, we analyze challenges associated with treating language models as RL policies and show how avoiding those challenges requires moving beyond the RL paradigm.

We start by observing that naive maximisation of the standard RL objective unavoidably leads to distribution collapse: turning the LM into a degenerate distribution. Then, we analyze KL-regularised RL, a widely used recipe for fine-tuning LMs, which additionally constrains the fine-tuned LM to stay close to its original distribution. We show that KL-regularised RL is equivalent to variational inference: approximating a Bayesian posterior which specifies how to update a prior LM to conform with evidence provided by the reward function.

Based on these observations, we sketch a Bayesian perspective that provides a first-principles derivation for KL-regularised RL. The Bayesian perspective also separates the modelling problem (defining a target distribution specifying the desired behaviour of an LM) and the inference problem (approximating that target distribution). Finally, it suggests that RL is not a good formal framework for thinking about fine-tuning LMs.

1 Introduction

Large language models (LMs), such as GPT-3 (Brown et al., 2020), tend to generate outputs that reflect undesirable features of their training data such as offensiveness (Gehman et al., 2020), social bias (Bender et al., 2021), harmfulness (Bai et al., 2022) or dishonesty (Lin et al., 2021). Addressing these biases and constraining LMs to be honest, helpful and harmless is an essential part of the problem of aligning LMs with human preferences (Askell et al., 2021). One intuitive approach to aligning LMs is reinforcement learning (RL): capturing human preferences as a reward function and fine-tuning the LM to maximise the reward expected under LM distribution. A practical recipe for implementing this idea is RL from human feedback (Ziegler et al., 2019): first, a reward model is trained to predict which of two texts a human prefers and then a pretrained LM is fine-tuned to maximise reward given by the reward model while also being penalized for Kullback-Leibler (KL) divergence from its initial distribution. However, despite immense popularity of RL from human feedback (Stiennon et al., 2020; Ouyang et al., 2022; Perez et al., 2022; Bai et al., 2022), the motivation for KL penalty is not widely understood.

In this paper, we discuss an underappreciated perspective on KL-regularised RL — the objective employed by RL from human feedback for fine-tuning LMs — which explains its empirical
success. We start with describing a problem that arises from naively applying the standard RL objective: distribution collapse. The optimal policy under the RL objective would be a minimal-entropy LM generating a small set of sequences that obtain the highest reward. Then, we discuss how KL-regularised RL avoids distribution collapse due to its KL penalty. This constraint, we argue, transforms the problem from RL to Bayesian inference: updating a prior to conform with evidence provided by the reward. The Bayesian perspective moves KL-regularised RL closer to other divergence-minimisation-based approaches to fine-tuning LMs (Khalifa et al., 2021) and, more broadly, to other divergence-minimisation-based accounts of control (Levine, 2018; Hafner et al., 2020). These divergence minimisation approaches naturally avoid the distribution collapse problem because they formalize the agent as a generative model. In contrast, RL avoids distribution collapse only with reward functions that make it equivalent to divergence minimisation. Therefore, we conclude, RL is not an adequate formal framework for problems such as fine-tuning LMs.

2 Fine-tuning language models using standard RL and distribution collapse

Let \( \mathcal{X} \) be the set of sequences of tokens from some vocabulary. An LM \( \pi \) can be seen as a probability distribution over \( \mathcal{X} \). While most modern LMs are autoregressive, for simplicity we will only talk about full sequences, e.g. \( \pi(x) \) denotes the probability of a sequence \( x \in \mathcal{X} \). Similarly, a reward function \( r \) assigns sequences \( x \in \mathcal{X} \) with scalar rewards. In practice, \( r(x) \) could represent human preferences we would like \( \pi \) to be aligned with, e.g. a non-offensiveness reward would assign low values to sequences that are offensive.

If \( \pi_\theta \) is our parametric LM (with parameters \( \theta \)), the RL objective for fine-tuning it with our reward function \( r \) is just the reward expected under LM distribution:

\[
J_{RL}(\theta) = \mathbb{E}_{x \sim \pi_\theta} r(x).
\]

Intuitively, maximising \( J_{RL}(\theta) \) means sampling a number of sequences from the LM and rewarding the LM for good sequences and penalising for bad ones (e.g. offensive sentences).

The problem with the RL objective is that it treats the LM as a policy, not as a generative model. While a generative model is supposed to capture a diverse distribution of samples, a policy is supposed to chose the optimal action. Since we don’t have a notion of state for LMs, the RL objective reduces to searching for \( x^* \), the sequence with highest reward. If there is one, the optimal policy \( \pi^* \) is a degenerate, deterministic generative model that puts entire probability mass on that single sequence:

\[
\pi^* = \arg\max_{\theta} J_{RL}(\theta) = \delta_{x^*},
\]

where \( \delta_{x^*} \) is a Dirac delta distribution centred on \( x^* \). If there are multiple optimal sequences \( x^* \), probability mass would be put only on them.

This failure mode is not purely theoretical. Empirically, distribution collapse induced by maximising reward manifests as decreased fluency and diversity of samples from the LM, which can be measured in terms of perplexity, entropy and the frequency of repetitions. Degeneration of this kind was observed in multiple language generation tasks ranging from translation (Choshen et al., 2019), summarisation (Paulus et al., 2018), story generation (Tambwekar et al., 2019), video captioning (Pa-sunuru and Bansal, 2017), dialogue (Jaques et al., 2019), to code generation (Korbak et al., 2021) and LM debiasing (Khalifa et al., 2021).

While the distribution collapse problem is exacerbated by RL failure modes such as insufficient exploration or reward hacking, it is distinct from exploration-exploitation trade-off or reward misspecification. Even with perfect exploration (if we sampled sequences uniformly from \( \mathcal{X} \) as opposed to sampling from \( \pi_\theta \)), the optimal policy will still put all probability mass on \( x^* \). Similarly, even if \( r \) perfectly captures human preferences across the whole space of possible sequences \( \mathcal{X} \) and if \( x^* \) is truly the best thing, we still would not want the LM to generate only \( x^* \).1 Essentially, the distribution collapse problem arises from the fact that

---

1There is a case to be made that in conditional generation (e.g. translation or summarisation) one really cares only about the single best output for a given context (e.g. summary of a document). There are still, however, substantial benefits of caring about distributional aspects in conditional generation. First, when the LM produces a full distribution, we are able to measure its uncertainty. For larger models, these uncertainty estimates happen to be well-calibrated and allow for safer deployment in high-stakes scenarios (Kadavath et al., 2022). Second, MAP estimates of the output distribution (the single, most likely output) are frequently of poor quality and can be substantially improved upon with decoding procedures taking into account the entire distribution, e.g. minimum Bayes risk in translation (Eikema and Aziz, 2020) or self-consistency chain-of-thought in question answering (Wang et al., 2022). Dohan et al. (2022) provide a unifying perspective on multi-step generation as latent variable modelling.
the RL objective for LM alignment is flawed: it does not care about preserving distributional properties of an LM and will always penalise the LM for putting any probability mass on non-optimal sequences until the LM collapses into a degenerate distribution.

3 Fine-tuning language models via KL-regularised RL

There is an obvious solution to the distribution collapse problem: including preserving distributional properties of an LM as part of the reward function. The notion of preserving distributional properties of an LM $\pi_0$ can be formalised as penalising for Kullback-Leibler (KL) divergence between $\pi_\theta$ and some other, pretrained LM $\pi_0$. Typically, $\pi_\theta$ is initialised to $\pi_0$ and then fine-tuned to maximise the following objective:

$$J_{KL-RL}(\theta) = E_{x \sim \pi_\theta}[r(x)] - \beta D_{KL}(\pi_\theta, \pi_0).$$

(3)

The first term in the right-hand side of (3) is equivalent to $J_{KL}(\theta)$ in (1) while the second additionally constrains $\pi_\theta$ to stay close (in terms of KL) to $\pi_0$. Almost always some reward needs to be sacrificed for that; the coefficient $\beta$ determines the trade-off of how much reward is needed to justify departing from $\pi_0$ by a certain distance. This objective is commonly used as part of a popular recipe for fine-tuning LMs termed “RL from Human Feedback” (RLHF) and works surprisingly well in practice (Ziegler et al., 2019; Stiennon et al., 2020; Perez et al., 2022; Bai et al., 2022). Earlier approaches to fine-tuning LMs employing this objective used the called it “conservative fine-tuning” (Jaques et al., 2017) or KL-control (Jaques et al., 2019). Here, we focus only on the policy optimisation part of this setup, which we term “KL-regularised RL”.

The KL-regularised RL objective (3) can easily be reformulated as just expected reward as in (1). We only have to define a new reward function $r'_\theta(x)$ which incorporates the original reward $r$ and the KL penalty, using the definition of KL divergence:

$$J_{KL-RL}(\theta) = E_{x \sim \pi_\theta}[r'_\theta(x)]$$

where

$$r'_\theta(x) = r(x) + \beta(\log \pi_0(x) - \log \pi_\theta(x)).$$

(4)

But is framing the maximisation of (4) as RL really necessary? In the next section, we will develop an alternative view of this objective – as an approximate solution to a Bayesian inference problem – and argue that it is a more appealing framing.

4 KL-regularised RL as variational inference

Fine-tuning a pretrained LM $\pi_0$ to align with preferences encoded by a reward function $r$ is essentially a Bayesian inference problem. Intuitively, Bayesian inference is the problem of updating a distribution to conform with new evidence. In our setting, we’re updating $\pi_\theta$, which is initially equal to a prior $\pi_0$ to conform with evidence provided by the assumption that $\pi_\theta$ is optimal in terms of $r$. A reward function can be represented as a distribution over $X$ that makes high-reward sequences more likely that low-reward sequences. A simple way of doing that is exponentiating reward $r$ and renormalizing it. Then, the posterior is given by:

$$\pi_{KL-RL}^*(x) = \frac{1}{Z} \pi_0(x) \exp(r(x)/\beta),$$

(5)

where $\pi_0$ is the prior, $\exp(r(x)/\beta)$ is the evidence provided by the reward function (scaled by temperature $\beta$) and $Z$ is a constant ensuring that $\pi_{KL-RL}^*$ is a normalised probability distribution. $\pi_{KL-RL}^*$ represents a version of $\pi_0$ updated to account for the reward $r$. As we demonstrate in the Appendix, it also happens to coincide with the optimal policy for $J_{KL-RL}$:

$$\pi_{KL-RL}^* = \arg\max_\theta J_{KL-RL}(\theta)$$

(6)

Moreover, the KL-regularised RL objective can be cast as minimising the KL divergence between the LM $\pi_\theta$ and this target distribution $\pi_{KL-RL}^*$:

$$J_{KL-RL}(\theta) \propto -D_{KL}(\pi_\theta, \pi_{KL-RL}^*)$$

(7)

This divergence is different from the KL penalty term $D_{KL}(\pi_\theta, \pi_0)$ in (3). Minimising this new divergence coincides with a variational inference (Blei et al., 2017), a well-known approach to approximating Bayesian inference. More formally, $J_{KL-RL}(\theta)$ is the evidence lower bound (ELBO) on the log likelihood of $\pi_\theta$ being optimal under $r$, assuming a prior $\pi_0$. Minimising this bound makes $\pi_\theta$ approximate the true posterior $\pi_{KL-RL}^*$. A derivation of these equalities can be found in the Appendix below.

Why is this picture insightful? It explains where the KL penalty term $\beta D_{KL}(\pi_\theta, \pi_0)$ in KL-regularised RL’s original objective comes from. It is necessary to transform the problem from RL to minimising a divergence from a target distribution $\pi_{KL-RL}^*$. This in turn makes the distributional character of an LM a first-class citizen which explains...
why KL-regularised RL is able to maintain the flu-
ency and diversity of the original LM $\pi_0$.

5 Separation of modelling and inference

The Bayesian perspective suggests that that align-
ing an LM with task preferences is a two-step
process. It consists of, first, defining a distri-
bution specifying the desired behaviour of your
LM, and second, solving the problem of sampling
from that posterior. These two steps roughly corre-
spond to modelling and inference in probabilistic
programming (Goodman and Stuhlmüller, 2014).
Modelling is encoding knowledge in probabilistic
terms (usually by defining a probabilistic graphical
model) while inference corresponds to using this
model to answer queries. It is hard to overstate how
useful — theoretically and practically — separat-
ing these two concerns could be. Let us discuss
these two steps, separately, below.

Modelling The LM is natively a probability distri-
bution and autoregressive models allow for both
sampling and evaluating likelihoods. Therefore,
most modelling decisions are usually around inter-
preting task preferences in probabilistic terms.
Turning a reward function $r$ into a distribution by
exponentiating it ($\frac{1}{Z} \exp(r(x))$) is the standard ap-
proach, but there are others. In some cases, task
preferences can be binary, for instance a dialogue
system might be required to never generate a curse
word (but is free to behave normally otherwise).
Then, following (Khalifa et al., 2021), one could
define $\pi^*(x) = \frac{1}{2} \pi_0(x) b(x)$, where $b(x) = 1$ if $x$
contains a curse and 0 otherwise. Then, sequences
$x$ containing curses have probability zero accord-
ging to $\pi^*$ (hence $\pi^*$ is non-cursing) but all other
strings keep the original probability $\pi_0(x)$ up to $Z$
(hence no degeneration).

Inference The posteriors mentioned above are
generally non-parametric: they might lie outside
the class of probability distributions representable
by parametric LMs. Designing an algorithm able
to generate samples matching this posterior distri-
bution constitute the inference problem. Broadly,
there are two families of algorithms for inference
on probabilistic graphical models: variational in-
ference and sampling-based approaches. Variational
inference tries to find the set of weights $\theta$ that give
rise to a distribution $\pi_\theta$ closest (in terms of KL)
to the true posterior. Sampling-based techniques,
such as MCMC (Brooks et al., 2011), do not rep-
resent the true posterior explicitly, but compute
samples from a distribution resembling the true
posterior. In the previous section, we have shown
that KL-regularised RL corresponds to inference
via variational inference. But sampling-based in-
fERENCE algorithms also have analogues for LMs in
decoding-time methods. Decoding-time methods
boil down to simulating a posterior, aligned LM $\pi^*$
by modifying the generation procedure applied on
top of the original LM $\pi_0$. The simplest example of
that is filtering (also known as rejection sampling):
if the LM generates an unacceptable sample, it is
discarded and a new sample is generated (Xu et
al., 2020). More elaborate decoding-time meth-
ods include weighted decoding (See et al., 2019)
or PPLM (Dathathri et al., 2019).

To summarise, the Bayesian view provides a
unifying perspective on fine-tuning and decoding-
time approaches to LM alignment. They mirror
variational inference and sampling-based inference
algorithms for probabilistic graphical models. But
a more fundamental advantage, to our mind, is the
separation of concerns between defining a desired
behaviour of an LM and approximating it. The
choice of posterior is independent of how it is go-
ing to be approximated. This, in turn, separates
two failure modes: misspecifying the model (i.e.
not capturing task preferences) and failing to ap-
proximate the model well enough.

6 Is RL a good framework for fine-tuning
language models?

There is a family of other divergence minimisa-
tion approaches to fine-tuning LMs which are not
equivalent to RL. Take Generative Distributional
Control (GDC) (Khalifa et al., 2021; Korbak et al.,
2022a), an approach to fine-tuning LMs that ob-
tains results comparable with KL-regularised RL
but minimises a slightly different divergence (for-
ward as opposed to reverse KL). However, $J_{GDC}(\theta)$
is no longer equivalent to RL (Korbak et al., 2022b)
because the expectation in forward KL divergence
is with respect to a $\pi_{KL-RL}^*$, not $\pi_\theta$. Similar-
larly, standard supervised training objective can be seen as
minimising $D_{KL}(\pi_{MLE}^*; \pi_\theta)$, a divergence from the
empirical distribution $\pi_{MLE}^*$ provided by the train-
ing set.

One can therefore mount a double dissociation
argument in favour of the divergence minimisa-
tion perspective on KL-regularised RL: RL with-
out KL divergence minimisation leads to degenera-
ation while KL divergence minimisation without RL works well. Therefore, it is the KL divergence minimisation aspect of KL-regularised RL that seems to account for its success, not the reward maximisation aspect. In consequence, calling it RL is just a redescription of it that happens to be correct under a particular choice of reward function $r'_q$. However, this redescription does not provide motivation for this choice of $r'_q$ and does not hold for alternative divergence minimisation approaches to fine-tuning LMs such as GDC (Khalifa et al., 2021).

The divergence minimisation perspective on KL-regularised RL we presented stems from a general framework known as control as inference (Levine, 2018). Control as inference provides a formalisation of intelligent decision making as inference on a probabilistic graphical model representing the agent, its preferences and environmental dynamics. While control as inference is typically considered with graphical models parameterised to make it equivalent to RL, it does not have to be. Moreover, there are frameworks such as active inference (Friston et al., 2010; Buckley et al., 2017) or action and perception as divergence minimisation (Hafner et al., 2020) that further generalise control as inference to a principle of minimising the KL divergence from a probability distribution representing desired behaviour of the agent. In contrast with RL, they conceptualise the agent as a generative model, not as a decision rule represented as a probability distribution out of convenience. Therefore, they naturally avoid the distribution collapse problem and preserve the distributional properties of the agent. What if RL simply isn’t an adequate formal framework for problems such as aligning LMs?

#### Limitations

In the paper, we discussed some limitations of standard approaches to using RL for fine-tuning LMs and sketched an alternative framing – based on Bayesian inference – of RLHF, a commonly used approach to RL fine-tuning. However, our discussion itself is limited in scope as we do not cover other shortcomings of RLHF and our own Bayesian proposal is not devoid of weaknesses. We take advantage of this section to examine these two sets of limitations.

**Other limitations of RLHF**  RLHF consists of (i) training a reward model to predict which of two texts a human prefers and (ii) fine-tuning a pre-trained LM to maximise reward given by the reward model. Our discussion focused on (ii) and took (i) as a given. But a reward model is always a proxy for the underlying task preferences and is limited in its ability to fit human feedback. Reward models are vulnerable to adversarial examples (Grosse et al., 2016; Hosseini et al., 2017) and LMs optimised against them can exploit these adversarial examples (Pan et al., 2022).

Moreover, training a reward model involves a multitude of non-technical design choices that shape the reward function the LM is optimised against. These design decisions involve data curation, annotation guideline preparation as well as annotator selection and compensation. Unintended bias can be introduced at each of these stages. For instance, crowdsource workers might be biased towards particular language varieties (Sap et al., 2019). More generally, preferences elicited from crowdsource workers might not represent the preferences of the general population due to selection effects. For instance, most studies using RLHF recruits crowdsource workers either solely from the United States (Bai et al., 2022) or from United States and Southeastern Asia (Stiennon et al., 2020; Ouyang et al., 2022). Crowdsource workers frequently disagree among themselves and with researchers conducting the study.\(^2\) This diversity of preferences makes the notion of a ground truth for the reward model problematic; see (Ouyang et al., 2022, sec. 5.3-5.3) for an extended discussion and (Gabriel, 2020) for a philosophical examination of the notion of ground truth human preferences.

**Limitations of the Bayesian perspective**  We argued that RL with KL penalties and, more broadly, aligning language models with human preferences, can be seen as Bayesian inference and that this perspective is a more insightful theoretical grounding for RLHF than the standard RL perspective. However, our proposal as laid down above is only preliminary and does not account for some empirical regularities found in RLHF experiments. For instance, Bai et al. (2022) found that expected reward $\mathbb{E}_{x \sim \pi_g} r(x)$ is approximately linear in $\sqrt{D_{KL}(\pi_\theta, \pi_0)}$ throughout RLHF training. The Bayesian perspective remains to be developed to explain why such a relationship holds. Moreover, the Bayesian perspective currently offers limited

\(^2\)The annotator-annotator agreement rates are 68% in (Stiennon et al., 2020) and 72% in (Ouyang et al., 2022) while the the annotator-researcher agreement rates are 77% in (Stiennon et al., 2020) and 63% in (Bai et al., 2022).
guidance for design choices in RLHF experiments such as hyperparameter selection.

Ethics statement

Our paper is a contribution to important lines of work on social bias in large language models and on aligning artificial intelligence with human preferences. The first line of work is primarily concerned with risks associated with an over-representation of certain hegemonic (e.g. sexist, racist, homophobic) viewpoints and voices present in the training data for large language models, which consists primarily of crawled, uncurated user-generated content. Deploying language models exhibiting social biases poses a risk of amplifying and perpetuating these biases (Sheng et al., 2019; Blodgett et al., 2020; Bender et al., 2021). The second line of work is concerned more broadly with ensuring that objectives that machine learning systems pursue are aligned with human values (Amodei et al., 2016; Russell, 2019). Large language models, due to their capabilities, can be a testbed for alignment techniques for future, more powerful machine learning systems (Askell et al., 2021; Bowman, 2021). Research on RLHF for fine-tuning LMs – such as our paper – can therefore be motivated by both narrower (social bias) and broader (alignment) considerations. As a theoretical contribution, our paper is not expected to pose significant risk. However, RLHF is a dual use technology: it can be diverted to malicious uses such as spreading misinformation or generating harmful content.

Acknowledgements

Tomasz Korbak was supported by the Leverhulme Doctoral Scholarship and OpenPhilantropy. This paper benefited from discussions with Hady Elsahar, Marc Dymetman, Germán Kruszewski, Jérémy Scheurer, Kyle McDonell and Laria Reynolds.

References

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. 2016. Concrete problems in ai safety.


Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndoousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. 2022. *Language models (mostly) know what they know*.


Tomasz Korbak, Hady Elsahar, Germán Kruszewski, and Marc Dymetman. 2022b. *On reinforcement learning and distribution matching for fine-tuning language models with no catastrophic forgetting*.


Appendix

Let’s assume we have a prior distribution over sequences of tokens \( \pi_0(x) \) and a reward function \( r \) which is (for technical reasons) always negative (from \(-\infty\) to 0). We can also represent \( r \) as a binary random variable \( \mathcal{O} \) (the optimality variable). \( \mathcal{O} = 1 \) if a certain LM \( \pi \) is optimal. We can define \( \mathcal{O} \) in terms of \( r \) as

\[
p(\mathcal{O} = 1|x) = \exp(r(x)), \tag{8}
\]

which is normalised because \( r(x) \) is always negative. For instance, if \( r(x) \) is a log probability that a sequence \( x \) is non-offensive, \( p(\mathcal{O} = 1|x) \) is a probability that \( x \) is non-offensive and the marginal \( p(\mathcal{O} = 1) \) is the average offensiveness score of \( \pi \) (or a probability that a random sample from \( \pi \) is non-offensive). The problem of aligning LMs can be seen as inferring \( p(x|\mathcal{O} = 1) \), a distribution over sequences of tokens conditioned on being non-offensive. This can be computed by applying Bayes rule as

\[
p(x|\mathcal{O} = 1) = \frac{p(\mathcal{O} = 1|x)p(x)}{p(\mathcal{O} = 1)} \tag{9}
\]

\[
= \frac{1}{Z} a(x) \exp(r(x)/\beta), \tag{10}
\]

where we chose the prior \( p(x) = \pi_0(x) \), redefined the marginal \( p(\mathcal{O} = 1) \) as the normalising constant \( Z \), used the definition of \( p(\mathcal{O} = 1|x) \) and chose \( \beta = 1 \). \( p(x|\mathcal{O} = 1) \) here is equivalent to \( \pi_{KL-RL}^* \), the optimal policy under objective in (3) (up to the choice of \( \beta \) which can be absorbed into \( r \)).

\( p(x|\mathcal{O} = 1) \) is a non-parametric distribution: it doesn’t have to lie in the family of distributions representable by a parametric model. In general, we’d like to find a parametric model \( \pi_\theta \) closest to \( \pi_{KL-RL}^* \). This can be formalised as finding \( \pi_\theta \) minimising \( D_{KL}(\pi_\theta, \pi_{KL-RL}) \). Here, however, we will derive this objective from a yet more general perspective: inferring a random latent variable \( x \) that best explains the assumption that certain LM \( \pi \) is optimal given a prior \( \pi_0(x) \). This can be seen as maximising the log-likelihood of \( \mathcal{O} = 1 \) via variational inference:

\[
\log p(\mathcal{O} = 1) = \log \sum_x p(\mathcal{O} = 1, x) \tag{11}
\]

\[
= \log \left[ \sum_x p(\mathcal{O} = 1|x)\pi_0(x) \right] \tag{12}
\]

\[
= \log \left[ \sum_x \pi_\theta(x)p(\mathcal{O} = 1|x)\pi_0(x) \frac{\pi_0(x)}{\pi_\theta(x)} \right] \tag{13}
\]

\[
\geq \sum_x \pi_\theta(x) \log \left[ \frac{p(\mathcal{O} = 1|x)\pi_0(x)}{\pi_\theta(x)} \right] \tag{14}
\]

\[
= \mathbb{E}_{x \sim \pi_\theta} \log \left[ \frac{\exp(r(x))\pi_0(x)}{\pi_\theta(x)} \right] \tag{15}
\]

In this derivation, we first introduce a latent variable \( x \) using the sum rule of probability (11), factorize a joint distribution (12), introduce a variational distribution \( \pi_\theta \) over that latent variable (13), use Jensen’s inequality to obtain a bound (ELBo) (14) and, finally in (15), use the definition of \( p(\mathcal{O} = 1|x) \). This new bound can be alternatively expressed in two different ways:

\[
\mathbb{E}_{x \sim \pi_\theta} [r(x)] - D_{KL}(\pi_\theta, \alpha), \tag{16}
\]

\[
-\mathbb{E}_{x \sim \pi_\theta} \log \frac{\pi_\theta(x)}{\pi_0(x) \exp(r(x))}. \tag{17}
\]

(16) is just KL-regularised RL objective \( J_{KL-RL}(\pi_\theta, \beta = 1) \). (17) is proportional (up to a constant \(-\log Z\)) to negative \( D_{KL}(\pi_\theta, \pi_{KL-RL}) \), where \( \pi_{KL-RL}^* = \frac{1}{Z} \pi_0(x) \exp(r(x)) \) is the target distribution (or optimal policy for \( J_{KL-RL}(\pi_\theta) \)). Their equivalence proves that KL-regularised reward maximisation is equivalent to minimising divergence from \( \pi_{KL-RL}^* \).

More broadly, the derivation above shows that \( J_{KL-RL}(\pi_\theta) \) can be derived from first principles under a framework called control-as-inference (Levine, 2018). The central idea here is to start from a Bayesian inference problem: inferring a distribution over \( x \) (an LM) that reconciles the assumption that this LM is optimal \( (p(\mathcal{O} = 1) = 1, \) which plays we role of evidence) with a prior \( \pi_0(x) \). KL-regularised RL arises as we solve this inference problem approximately via variational inference, i.e. by introducing a variational distribution \( \pi_\theta \) and optimising it to maximise a lower bound on \( \log p(\mathcal{O} = 1) \).