VIREL: A Variational Inference Framework for Reinforcement Learning

Matthew Fellows[*] **Anuj Mahajan**[*] **Tim G. J. Rudner** **Shimon Whiteson**
Department of Computer Science
University of Oxford

Abstract

Applying probabilistic models to reinforcement learning (RL) enables the uses of powerful optimisation tools such as variational inference in RL. However, existing inference frameworks and their algorithms pose significant challenges for learning optimal policies, for example, the lack of mode capturing behaviour in pseudo-likelihood methods, difficulties learning deterministic policies in maximum entropy RL based approaches, and a lack of analysis when function approximators are used. We propose VIREL, a theoretically grounded inference framework for RL that utilises a parametrised action-value function to summarise future dynamics of the underlying MDP, generalising existing approaches. VIREL also benefits from a mode-seeking form of KL divergence, the ability to learn deterministic optimal polices naturally from inference, and the ability to optimise value functions and policies in separate, iterative steps. Applying variational expectation-maximisation to VIREL, we show that the actor-critic algorithm can be reduced to expectation-maximisation, with policy improvement equivalent to an E-step and policy evaluation to an M-step. We derive a family of actor-critic methods from VIREL, including a scheme for adaptive exploration and demonstrate that our algorithms outperform state-of-the-art methods based on soft value functions in several domains.

1 Introduction

Efforts to combine reinforcement learning (RL) and probabilistic inference have a long history, spanning diverse fields such as control, robotics, and RL [64, 62, 46, 47, 27, 74, 75, 73, 36]. Formalising RL as probabilistic inference enables the application of many approximate inference tools to reinforcement learning, extending models in flexible and powerful ways [35]. However, existing methods at the intersection of RL and inference suffer from several deficiencies. Methods that derive from the pseudo-likelihood inference framework [12, 64, 46, 26, 44, 1] and use expectation-maximisation (EM) favour risk-seeking policies [34], which can be suboptimal. Yet another approach, the MERL inference framework [35] (which we refer to as MERLIN), derives from maximum entropy reinforcement learning (MERL) [33, 74, 75, 73]. While MERLIN does not suffer from the issues of the pseudo-likelihood inference framework, it presents different practical difficulties. These methods do not naturally learn deterministic optimal policies and constraining the variational policies to be deterministic renders inference intractable [47]. As we show by way of counterexample in Section 2.2, an optimal policy under the reinforcement learning objective is not guaranteed from the optimal MERL objective. Moreover, these methods rely on soft value functions which are sensitive to a pre-defined temperature hyperparameter.

Additionally, no existing framework formally accounts for replacing exact value functions with function approximators in the objective; learning function approximators is carried out independently of the inference problem and no analysis of convergence is given for the corresponding algorithms.

[*] Equal Contribution. Correspondence to matthew.fellows@cs.ox.ac.uk and anuj.mahajan@cs.ox.ac.uk.

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

This paper addresses these deficiencies. We introduce VIREL, an inference framework that translates the problem of finding an optimal policy into an inference problem. Given this framework, we demonstrate that applying EM induces a family of actor-critic algorithms, where the E-step corresponds exactly to policy improvement and the M-step exactly to policy evaluation. Using a variational EM algorithm, we derive analytic updates for both the model and variational policy parameters, giving a unified approach to learning parametrised value functions and optimal policies.

We extensively evaluate two algorithms derived from our framework against DDPG [38] and an existing state-of-the-art actor-critic algorithm, soft actor-critic (SAC) [25], on a variety of OpenAI gym domains [9]. While our algorithms perform similarly to SAC and DDPG on simple low dimensional tasks, they outperform them substantially on complex, high dimensional tasks.

The main contributions of this work are: 1) an exact reduction of entropy regularised RL to probabilistic inference using value function estimators; 2) the introduction of a theoretically justified general framework for developing inference-style algorithms for RL that incorporate the uncertainty in the optimality of the action-value function, $\hat{Q}_\omega(h)$, to drive exploration, but that can also learn optimal deterministic policies; and 3) a family of practical algorithms arising from our framework that adaptively balances exploration-driving entropy with the RL objective and outperforms the current state-of-the-art SAC, reconciling existing advanced actor critic methods like A3C [43], MPO [1] and EPG [10] into a broader theoretical approach.

2 Background

We assume familiarity with probabilistic inference [30] and provide a review in Appendix A.

2.1 Reinforcement Learning

Formally, an RL problem is modelled as a Markov decision process (MDP) defined by the tuple $\langle\mathcal{S},\mathcal{A},r,p,p_0,\gamma\rangle$ [54, 59], where $\mathcal{S}$ is the set of states and $\mathcal{A}\subseteq\mathbb{R}^n$ the set of available actions. An agent in state $s\in\mathcal{S}$ chooses an action $a\in\mathcal{A}$ according to the policy $a\sim\pi(\cdot|s)$, forming a state-action pair $h\in\mathcal{H}$, $h:=\langle s,a\rangle$. This pair induces a scalar reward according to the reward function $r_t:=r(h_t)\in\mathbb{R}$ and the agent transitions to a new state $s'\sim p(\cdot|h)$. The initial state distribution for the agent is given by $s_0\sim p_0$. We denote a sampled state-action pair at timestep t as $h_t:=\langle s_t,a_t\rangle$. As the agent interacts with the environment using π, it gathers a trajectory $\tau=(h_0,r_0,h_1,r_1,...)$. The value function is the expected, discounted reward for a trajectory, starting in state s. The action-value function or Q-function is the expected, discounted reward for each trajectory, starting in h, $Q^\pi(h):=\mathbb{E}_{\tau\sim p^\pi(\tau|h)}\left[\sum_{t=0}^\infty\gamma^t r_t\right]$, where $p^\pi(\tau|h):=p(s_1|h_0=h)\prod_{t'=1}^\infty p(s_{t'+1}|h_{t'})\pi(a_t|s_t)$. Any Q-function satisfies a Bellman equation $\mathcal{T}^\pi Q^\pi(\cdot)=Q^\pi(\cdot)$ where $\mathcal{T}^\pi\cdot:=r(h)+\gamma\mathbb{E}_{h'\sim p(s'|h)\pi(a'|s')}\left[\cdot\right]$ is the Bellman operator. We consider infinite horizon problems with a discount factor $\gamma\in[0,1)$. The agent seeks an optimal policy $\pi^*\in\arg\max_\pi J^\pi$, where

$$J^\pi=\mathbb{E}_{h\sim p_0(s)\pi(a|s)}\left[Q^\pi(h)\right].\tag{1}$$

We denote optimal Q-functions as $Q^*(\cdot):=Q^{\pi^*}(\cdot)$ and the set of optimal policies $\Pi^*:=\arg\max_\pi J^\pi$. The optimal Bellman operator is $\mathcal{T}^*\cdot:=r(h)+\gamma\mathbb{E}_{h'\sim p(s'|h)}\left[\max_{a'}(\cdot)\right]$.

2.2 Maximum Entropy RL

The MERL objective supplements each reward in the RL objective with an entropy term [61, 74, 75, 73], $J^\pi_{\text{merl}}:=\mathbb{E}_{\tau\sim p(\tau)}\left[\sum_{t=0}^{T-1}(r_t-c\log(\pi(a_t|s_t))\right]$. The standard RL, undiscounted objective is recovered for $c\to0$ and we assume $c=1$ without loss of generality. The MERL objective is often used to motivate the MERL inference framework (which we call MERLIN) [34], mapping the problem of finding the optimal policy, $\pi^*_{\text{merl}}(a|s)=\arg\max_\pi J^\pi_{\text{merl}}$, to an equivalent inference problem. A full exposition of this framework is given by Levine [35] and we discuss the graphical model of MERLIN in comparison to VIREL in Section 3.3. The inference problem is often solved using a message passing algorithm, where the log backward messages are called soft value functions due to their similarity to classic (hard) value functions [63, 48, 25, 24, 35]. The soft Q-function is defined as $Q^\pi_{\text{soft}}(h):=\mathbb{E}_{\tau\sim q^\pi(\tau|h)}\left[r_0+\sum_{t=1}^{T-1}(r_t-\log\pi(a_t|s_t))\right]$, where $q^\pi(\tau|h):=p(s_0|h)\prod_{t=0}^{T-1}p(s_{t+1}|h_t)\pi(a_t|s_t)$. The corresponding soft Bellman operator is $\mathcal{T}^\pi_{\text{soft}}\cdot:=r(h)+\mathbb{E}_{h'\sim p(s'|h)\pi(a'|s')}[\cdot-\log\pi(a'|s')]$. Several algorithms have been developed that mirror existing RL algorithms using soft Bellman equations, including maximum entropy policy gradients [35], soft Q-learning [24], and soft actor-critic (SAC) [25]. MERL is also compatible with methods that use recall traces [21].

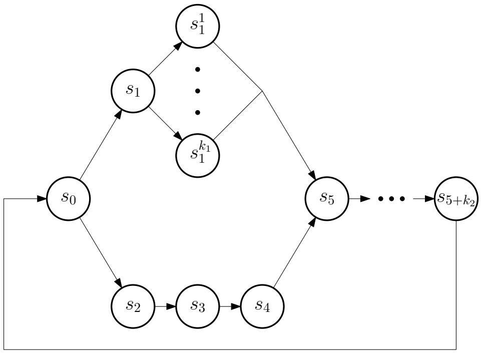

Figure 1: A discrete MDP counterexample for optimal policy under maximum entropy.

We now outline key drawbacks of MERLIN. It is well-understood that optimal policies under regularised Bellman operators are more stochastic than under their equivalent unregularised operators [20]. While this can lead to improved exploration, the optimal policy under these operators will still be stochastic, meaning optimal deterministic policies are not learnt naturally. This leads to two difficulties: 1) a deterministic policy can be constructed by taking the action $a^* = \arg\max_a \pi^*_{\text{merl}}(a|s)$, corresponding to the maximum a posteriori (MAP) policy, however, in continuous domains, finding the MAP policy requires optimising the Q-function approximator for actions, which is often a deep neural network. A common approximation is to use the mean of a variational policy instead; 2) even if we obtain a good approximation, as we show below by way of counterexample, the deterministic MAP policy is not guaranteed to be the optimal policy under J^π. Constraining the variational policies to the set of Dirac-delta distributions does not solve this problem either, since it renders the inference procedure intractable [47, 48].

Next, we demonstrate that the optimal policy under J^π cannot always be recovered from the MAP policy under J^π_{merl}. Consider the discrete state MDP as shown in Fig. 1, with action set $\mathcal{A} = \{a_1, a_2, a_1^1, \cdots a_1^{k_1}\}$ and state set $\mathcal{S} = \{s_0, s_1, s_2, s_3, s_4, s_1^1 \cdots s_1^{k_1}, s_5, \cdots s_{5+k_2}\}$. All state transitions are deterministic, with $p(s_1|s_0, a_1) = p(s_1|s_0, a_2) = p(s_1^i|s_1, a_1^i) = 1$. All other state transitions are deterministic and independent of action taken, that is, $p(s_j|\cdot, s_{j-1}) = 1\ \forall\ j > 2$ and $p(s_5|\cdot, s_1^i) = 1$. The reward function is $r(s_0, a_2) = 1$ and zero otherwise. Clearly the optimal policy under J^π has $\pi^*(a_2|s_0) = 1$. Define a maximum entropy reinforcement learning policy as π_{merl} with $\pi_{\text{merl}}(a_1|s_0) = p_1$, $\pi_{\text{merl}}(a_2|s_0) = (1 - p_1)$ and $\pi_{\text{merl}}(a_1^i|s_1) = p_1^i$. For π_{merl} and $k_2 >> 5$, we can evaluate J^π_{merl} for any scaling constant c and discount factor γ as:

$$J^\pi_{\text{merl}} = (1 - p_1)(1 - c\log(1 - p_1)) - p_1 \left(c\log p_1 + \gamma c \sum_{i=1}^{k} p_1^i \log p_1^i \right). \tag{2}$$

We now find the optimal MERL policy. Note that $p_1^i = \frac{1}{k}$ maximises the final term in Eq. (2). Substituting for $p_1^i = \frac{1}{k_1}$, then taking derivatives of Eq. (2) with respect to p_1, and setting to zero, we find $p_1^* = \pi^*_{\text{merl}}(a_1|s_0)$ as:

$$1 - c\log(1 - p_1^*) = \gamma c \log(k_1) - c\log p_1^*,$$
$$\implies p_1^* = \frac{1}{{k_1}^{-\gamma} \exp\left(\frac{1}{c}\right) + 1},$$

hence, for any ${k_1}^{-\gamma} \exp\left(\frac{1}{c}\right) < 1$, we have $p_1^* > \frac{1}{2}$ and so π^* cannot be recovered from π^*_{merl}, even using the mode action $a_1 = \arg\max_a \pi^*_{\text{merl}}(a|s_0)$. The degree to which the MAP policy varies from the optimal unregularised policy depends on both the value of c and k_1, the later controlling the number of states with sub-optimal reward. Our counterexample illustrates that when there are large regions of the state-space with sub-optimal reward, the temperature must be comparatively small to compensate, hence algorithms derived from MERLIN become very sensitive to temperature. As we discuss in Section 3.3, this problem stems from the fact that MERL policies optimise for expected reward and long-term expected entropy. While initially beneficial for exploration, this can lead to sub-optimal polices being learnt in complex domains as there is often too little a priori knowledge about the MDP to make it possible to choose an appropriate value or schedule for c.

Finally, a minor issue with MERLIN is that many existing models are defined for finite-horizon problems [35, 48]. While it is possible to discount and extend MERLIN to infinite-horizon problems, doing so is often nontrivial and can alter the objective [60, 25].

2.3 Pseudo-Likelihood Methods

A related but distinct approach is to apply Jensen's inequality directly to the RL objective J^π. Firstly, we rewrite Eq. (1) as an expectation over τ to obtain $J = \mathbb{E}_{h\sim p_0(s)\pi(a|s)}\left[Q^\pi(h)\right] = \mathbb{E}_{\tau\sim p(\tau)}\left[R(\tau)\right]$, where $R(\tau) = \sum_{t=0}^{T-1}\gamma^t r_t$ and $p(\tau) = p_0(s_0)\pi(a_0|s_o)\prod_{t=0}^{T-1}p(h_{t+1}|h_t)$. We then treat $p(R,\tau) = R(\tau)p(\tau)$ as a joint distribution, and if rewards are positive and bounded, Jensen's inequality can be applied, enabling the derivation of an evidence lower bound (ELBO). Inference algorithms such as EM can then be employed to find a policy that optimises the pseudo-likelihood objective [12, 64, 46, 26, 44, 1]. Pseudo-likelihood methods can also be extended to a model-based setting by defining a prior over the environment's transition dynamics. Furmston & Barber [19] demonstrate that the posterior over all possible environment models can be integrated over to obtain an optimal policy in a Bayesian setting.

Many pseudo-likelihood methods minimise $\mathrm{KL}(p_\mathcal{O} \parallel p_\pi)$, where p_π is the policy to be learnt and $p_\mathcal{O}$ is a target distribution monotonically related to reward [35]. Classical RL methods minimise $\mathrm{KL}(p_\pi \parallel p_\mathcal{O})$. The latter encourages learning a mode of the target distribution, while the former encourages matching the moments of the target distribution. If the optimal policy can be represented accurately in the class of policy distributions, optimisation converges to a global optimum and the problem is fully observable, the optimal policy is the same in both cases. Otherwise, the pseudo-likelihood objective reduces the influence of large negative rewards, encouraging risk-seeking policies.

3 VIREL

Before describing our framework, we state some relevant assumptions.

Definition 1 (Unique Maximum and Locally Smooth Function)**.** *Let $f:\mathcal{X}\rightarrow\mathcal{Y}$ be a function with a unique maximum $f(x^*) = \sup_x f$ and a bounded domain $\mathcal{X}$ and range $\mathcal{Y}$. Let f be locally smooth about x^*, is., $\exists\,\Delta > 0\ s.t. f(x) \in \mathbb{C}^2\ \forall\, x \in \{x|\|x-x^*\| < \Delta\}$.*

Assumption 1. *The optimal action-value function for the reinforcement learning problem is finite and strictly positive, i.e., $0 < Q^*(h) < \infty\ \forall\, h \in \mathcal{H}$.*

Any MDP for which rewards are lower bounded and finite, that is, $R \subset [r_{\min}, \infty)$, satisfies Assumption 1. To see this, we can construct a new MDP by adding $r_{\min}$ to the reward function, ensuring that all rewards are positive and hence the optimal action-value function for the reinforcement learning problem is finite and strictly positive. This does not affect the optimal solution. Now we introduce a function approximator $\hat{Q}_\omega(h) \approx Q^\pi(h)$ parametrised by $\omega \in \Omega$.

Assumption 2 (Exact Representability Under Optimisation)**.** *Our function approximator can represent the optimal Q-function, i.e., $\exists\,\omega^* \in \Omega\ s.t.\ Q^*(\cdot) = \hat{Q}_{\omega^*}(\cdot)$.*

In Appendix F.1, we extend the work of Bhatnagar et al. [6] to continuous domains, demonstrating that Assumption 2 can be neglected if projected Bellman operators are used.

Assumption 3 (Local Smoothness of Q-functions)**.** *For ω^* parametrising $Q^*(h)$ in Assumption 2, $Q_{\omega^*}(h)$ has a unique maximum and is locally smooth under Definition 1 for actions in any state.*

This assumption is formally required for the strict convergence of a Boltzmann to a Dirac-delta distribution and, as we discuss in Appendix F.4, is of more mathematical than practical concern.

3.1 Objective Specification

We now define an objective that we motivate by satisfying three desiderata: ① In the limit of maximising our objective, a deterministic optimal policy can be recovered and the optimal Bellman equation is satisfied by our function approximator; ② when our objective is not maximised, stochastic policies can be recovered that encourage effective exploration of the state-action space; and ③ our objective permits the application of powerful and tractable optimisation algorithms from variational inference that optimise the risk-neutral form of KL divergence, $\mathrm{KL}(p_\pi \parallel p_\mathcal{O})$, introduced in Section 2.3.

Firstly, we define the residual error $\varepsilon_\omega := \frac{c}{p}\|\mathcal{T}_\omega\hat{Q}_\omega(h) - \hat{Q}_\omega(h)\|_p^p$ where $\mathcal{T}_\omega = \mathcal{T}^{\pi_\omega}\cdot := r(h) + \gamma\mathbb{E}_{h'\sim p(s'|h)\pi_\omega(a'|s')}\left[\cdot\right]$ is the Bellman operator for the Boltzmann policy with temperature ε_ω:

$$\pi_\omega(a|s) := \frac{\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right)}{\int_\mathcal{A}\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right)da}. \tag{3}$$

We assume $p = 2$ and $c = \frac{1}{|\mathcal{H}|}$ without loss of generality. Our main result in Theorem 2 proves that finding a ω^* that reduces the residual error to zero, i.e., $\varepsilon_{\omega^*} = 0$, is a sufficient condition for learning an optimal Q-function $\hat{Q}_{\omega^*}(h) = Q^*(h)$. Additionally, the Boltzmann distribution $\pi_\omega(a|s)$ tends towards a Dirac-delta distribution $\pi_\omega(a|s) = \delta(a = \arg\max'_a \hat{Q}_{\omega^*}(a', s))$ whenever $\varepsilon_\omega \to 0$ (see Theorem 1), which is an optimal policy. The simple objective $\arg\min(\mathcal{L}(\omega)) := \arg\min(\varepsilon_\omega)$ therefore satisfies ①. Moreover, when our objective is not minimised, we have $\varepsilon_\omega > 0$ and from Eq. (3) we see that $\pi_\omega(a|s)$ is non-deterministic *for all non-optimal* ω. $\mathcal{L}(\omega)$ therefore satisfies ② as any agent following $\pi_\omega(a|s)$ will continue exploring until the RL problem is solved. To generalise our framework, we extend $\mathcal{T}_\omega \cdot$ to any operator from the set of target operators $\mathcal{T}_\omega \cdot \in \mathbb{T}$:

Definition 2 (Target Operator Set). *Define $\mathbb{T}$ to be the set of target operators such that an optimal Bellman operator for $\hat{Q}_\omega(h)$ is recovered when the Boltzmann policy in Eq.* (3) *is greedy with respect to $\hat{Q}_\omega(h)$, i.e., $\mathbb{T} := \{\mathcal{T}_\omega \cdot \,|\, \lim_{\varepsilon_\omega \to 0} \pi_\omega(a|s) \implies \mathcal{T}_\omega \hat{Q}_\omega(h) = \mathcal{T}^* \hat{Q}_\omega(h)\}$.*

As an illustration, we prove in Appendix C that the Bellman operator $\mathcal{T}^{\pi_\omega} \cdot$ introduced above is a member of $\mathbb{T}$ and can be approximated by several well-known RL targets. We also discuss how $\mathcal{T}^{\pi_\omega} \cdot$ induces a constraint on Ω due to its recursive definition. As we show in Section 3.2, there exists an ω in the constrained domain that maximises the RL objective under these conditions, so an optimal solution is always feasible. Moreover, we provide an analysis in Appendix F.5 to establish that such a policy is an attractive fixed point for our algorithmic updates, even when we ignore this constraint. Off-policy operators will not constrain Ω: by definition, the optimal Bellman operator $\mathcal{T}^* \cdot$ is a member of $\mathbb{T}$ and does not constrain Ω; similarly, we derive an off-policy operator based on a Boltzmann distribution with a diminishing temperature in Appendix F.2 that is a member of $\mathbb{T}$. Observe that soft Bellman operators are not members of $\mathbb{T}$ as the optimal policy under J^π_{merl} is not deterministic, hence algorithms such as SAC cannot be derived from the VIREL framework.

One problem remains: calculating the normalisation constant to sample directly from the Boltzmann distribution in Eq. (3) is intractable for many MDPs and function approximators. As such, we look to variational inference to learn an approximate *variational policy* $\pi_\theta(a|s) \approx \pi_\omega(a|s)$, parametrised by $\theta \in \Theta$ with finite variance and the same support as $\pi_\omega(a|s)$. This suggests optimising a new objective that penalises $\pi_\theta(a|s)$ when $\pi_\theta(a|s) \neq \pi_\omega(a|s)$ but still has a global maximum at $\varepsilon_\omega = 0$. A tractable objective that meets these requirements is the evidence lower bound (ELBO) on the unnormalised potential of the Boltzmann distribution, defined as $\{\omega^*, \theta^*\} \in \arg\max_{\omega,\theta} \mathcal{L}(\omega, \theta)$,

$$\mathcal{L}(\omega, \theta) := \mathbb{E}_{s\sim d(s)}\left[\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right] + \mathscr{H}(\pi_\theta(a|s))\right], \tag{4}$$

where $q_\theta(h) := d(s)\pi_\theta(a|s)$ is a variational distribution, $\mathscr{H}(\cdot)$ denotes the differential entropy of a distribution, and $d(s)$ is any arbitrary sampling distribution with support over $\mathcal{S}$. From Eq. (4), maximising our objective with respect to ω is achieved when $\varepsilon_\omega \to 0$ and hence $\mathcal{L}(\omega, \theta)$ satisfies ① and ②. As we show in Lemma 1, $\mathscr{H}(\cdot)$ in Eq. (4) causes $\mathcal{L}(\omega, \theta) \to -\infty$ whenever $\pi_\theta(a|s)$ is a Dirac-delta distribution for all $\varepsilon_\omega > 0$. This means our objective heavily penalises premature convergence of our variational policy to greedy Dirac-delta policies except under optimality. We discuss a probabilistic interpretation of our framework in Appendix B, where it can be shown that $\pi_\omega(a|s)$ characterises our model's uncertainty in the optimality of $\hat{Q}_\omega(h)$.

We now motivate $\mathcal{L}(\omega, \theta)$ from an inference perspective: In Appendix D.1, we write $\mathcal{L}(\omega, \theta)$ in terms of the log-normalisation constant of the Boltzmann distribution and the KL divergence between the action-state normalised Boltzmann distribution, $p_\omega(h)$, and the variational distribution, $q_\theta(h)$:

$$\mathcal{L}(\omega, \theta) = \ell(\omega) - \text{KL}(q_\theta(h) \parallel p_\omega(h)) - \mathscr{H}(d(s)), \tag{5}$$

$$\text{where} \quad \ell(\omega) := \log \int_{\mathcal{H}} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) dh, \quad p_\omega(h) := \frac{\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right)}{\int_{\mathcal{H}} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) dh}.$$

As the KL divergence in Eq. (5) is always positive and the final entropy term has no dependence on ω or θ, maximising our objective for θ always reduces the KL divergence between $\pi_\omega(a|s)$ and $\pi_\theta(a|s)$ for any $\varepsilon_\omega > 0$, with $\pi_\theta(a|s) = \pi_\omega(a|s)$ achieved under exact representability (see Theorem 3). This yields a tractable way to estimate $\pi_\omega(a|s)$ at any point during our optimisation procedure by maximising $\mathcal{L}(\omega, \theta)$ for θ. From Eq. (5), we see that our objective satisfies ③, as we minimise the mode-seeking direction of KL divergence, $\mathrm{KL}(q_\theta(h) \parallel p_\omega(h))$, and our objective is an ELBO, which is the starting point for inference algorithms [30, 4, 17]. When the RL problem is solved and $\varepsilon_\omega = 0$, our objective tends towards infinity for *any* variational distribution that is non-deterministic (see Lemma 1). This is of little consequence, however, as whenever $\varepsilon_\omega = 0$, our approximator is the optimal value function, $\hat{Q}_{\omega^*}(h) = Q^*(h)$ (Theorem 2), and hence, $\pi^*(a|s)$ can be inferred exactly by finding $\max_{a'} \hat{Q}_{\omega^*}(a', s)$ or by using the policy gradient $\nabla_\theta \mathbb{E}_{d(s)\pi_\theta(a|s)}\left[\hat{Q}_{\omega^*}(h)\right]$ (see Section 4.2).

3.2 Theoretical Results

We now formalise the intuition behind ①-③. Theorem 1 establishes the emergence of a Dirac-delta distribution in the limit of $\varepsilon_\omega \to 0$. To the authors' knowledge, this is the first rigorous proof of this result. Theorem 2 shows that finding an optimal policy that maximises the RL objective in Eq. (1) reduces to finding the Boltzmann distribution associated with the parameters $\omega^* \in \arg\max_\omega \mathcal{L}(\omega, \theta)$. The existence of such a distribution is a sufficient condition for the policy to be optimal. Theorem 3 shows that whenever $\varepsilon_\omega > 0$, maximising our objective for θ always reduces the KL divergence between $\pi_\omega(a|s)$ and $\pi_\theta(a|s)$, providing a tractable method to infer the current Boltzmann policy.

Theorem 1 (Convergence of Boltzmann Distribution to Dirac Delta). *Let $p_\varepsilon : \mathcal{X} \to [0, 1]$ be a Boltzmann distribution with temperature $\varepsilon \in \mathbb{R}_{\geq 0}$, $p_\varepsilon(x) = \frac{\exp\left(\frac{f(x)}{\varepsilon}\right)}{\int_\mathcal{X} \exp\left(\frac{f(x)}{\varepsilon}\right)dx}$, where $f : \mathcal{X} \to \mathcal{Y}$ is a function that satisfies Definition 1. In the limit $\varepsilon \to 0$, $p_\varepsilon(x) \to \delta(x = \sup_{x'} f(x'))$.*

Proof. See Appendix D.2 □

Lemma 1 (Lower and Upper limits of $\mathcal{L}(\omega, \theta)$). *i) For any $\varepsilon_\omega > 0$ and $\pi_\theta(a|s) = \delta(a^*)$, we have $\mathcal{L}(\omega, \theta) = -\infty$. ii) For $\hat{Q}_\omega(h) > 0$ and any non-deterministic $\pi_\theta(a|s)$, $\lim_{\varepsilon_\omega \to 0} \mathcal{L}(\omega, \theta) = \infty$.*

Proof. See Appendix D.3. □

Theorem 2 (Optimal Boltzmann Distributions as Optimal Policies). *For ω^* that maximises $\mathcal{L}(\omega, \theta)$ defined in Eq.* (4)*, the corresponding Boltzmann policy induced must be optimal, i.e., $\{\omega^*, \theta^*\} \in \arg\max_{\omega,\theta} \mathcal{L}(\omega, \theta) \implies \pi_{\omega^*}(a|s) \in \Pi^*$.*

Proof. See Appendix D.3. □

Theorem 3 (Maximising the ELBO for θ). *For any $\varepsilon_\omega > 0$, $\max_\theta \mathcal{L}(\omega, \theta) = \mathbb{E}_{d(s)}\left[\min_\theta \mathrm{KL}(\pi_\theta(a|s) \parallel \pi_\omega(a|s))\right]$ with $\pi_\omega(a|s) = \pi_\theta(a|s)$ under exact representability.*

Proof. See Appendix D.4. □

3.3 Comparing VIREL and MERLIN Frameworks

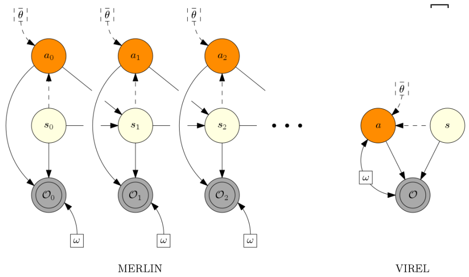

Figure 2: Graphical models for MERLIN and VIREL (variational approximations are dashed).

To compare MERLIN and VIREL, we consider the probabilistic interpretation of the two models discussed in Appendix B; introducing a binary variable $\mathcal{O} \in \{0, 1\}$ defines a graphical model for our inference problem whenever $\varepsilon_\omega > 0$. Comparing the graphs in Fig. 2, observe that MERLIN models exponential *cumulative* rewards over entire trajectories. By contrast, VIREL's variational policy models a single step and a function approximator is used to model future *expected* rewards. The resulting KL divergence minimisation for MERLIN is therefore much more sensitive to the value of temperature, as this affects how much future entropy influences the variational policy. For VIREL, temperature is defined by the model, and updates to the variational policy will not be as sensitive to errors in its value or linear scaling as its influence only extends to a single interaction. We hypothesise that VIREL may afford advantages in higher dimensional domains where there is greater chance of encountering large regions of state-action space with sub-optimal reward; like our counterexample from Section 2, c must be comparatively small to balance the influence of entropy in these regions to prevent MERLIN algorithms from learning sub-optimal policies.

Theorem 1 demonstrates that, unlike in MERLIN, VIREL naturally learns optimal deterministic policies directly from the optimisation procedure while still maintaining the benefits of stochastic policies in training. While Boltzmann policies with fixed temperatures have been proposed before [49], as we discuss in Appendix B, the adaptive temperature ε_ω in VIREL's Boltzmann policy has a unique interpretation, characterising the model's uncertainty in the optimality of $\hat{Q}_\omega(h)$; both $\pi_\omega(a|s)$ and its variational approximation $\pi_\theta(a|s)$ have an adaptive variance that reduces as $\hat{Q}_\omega(h)\rightarrow Q^*(h)$, allowing us to benefit from uncertainty-driven exploration when sampling under $\pi_\theta(a|s)$.

4 Actor-Critic and EM

We now apply the expectation-maximisation (EM) algorithm [13, 23] to optimise our objective $\mathcal{L}(\omega,\theta)$. (See Appendix A for an exposition of this algorithm.) In keeping with RL nomenclature, we refer to $\hat{Q}_\omega(h)$ as the *critic* and $\pi_\theta(a|s)$ as the *actor*. We establish that the expectation (E-) step is equivalent to carrying out policy improvement and the maximisation (M-)step to policy evaluation. This formulation reverses the situation in most pseudo-likelihood methods, where the E-step is related to policy evaluation and the M-step is related to policy improvement, and is a direct result of optimising the forward KL divergence, $\text{KL}(q_\theta(h)\parallel p_\omega(h|\mathcal{O}))$, as opposed to the reverse KL divergence used in pseudo-likelihood methods. As discussed in Section 2.3, this mode-seeking objective prevents the algorithm from learning risk-seeking policies. We now introduce an extension to Assumption 2 that is sufficient to guarantee convergence.

Assumption 4 (Universal Variational Representability)**.** *Every Boltzmann policy can be represented as $\pi_\theta(a|s)$, i.e., $\forall\,\omega\in\Omega\;\exists\,\theta\in\Theta$ s.t. $\pi_\theta(a|s)=\pi_\omega(a|s)$.*

Assumption 4 is strong but, like in variational inference, our variational policy $\pi_\theta(a|s)$ provides a useful approximation when Assumption 4 does not hold. As we discuss in Appendix F.1, using projected Bellman errors also ensures that our M-step always converges no matter what our current policy is.

4.1 Variational Actor-Critic

In the E-step, we keep the parameters of our critic ω_k constant while updating the actor's parameters by maximising the ELBO with respect to θ: $\theta_{k+1}\leftarrow\arg\max_\theta\mathcal{L}(\omega_k,\theta)$. Using gradient ascent with step size α_{actor}, we optimise $\varepsilon_{\omega_k}\mathcal{L}(\omega_k,\theta)$ instead, which prevents ill-conditioning and does not alter the optimal solution, yielding the update (see Appendix E.1 for full derivation):

E-Step (Actor): $\theta_{i+1}\leftarrow\theta_i+\alpha_{\text{actor}}\left(\varepsilon_{\omega_k}\nabla_\theta\mathcal{L}(\omega_k,\theta)\right)|_{\theta=\theta_i}$,

$$\varepsilon_{\omega_k}\nabla_\theta\mathcal{L}(\omega_k,\theta)=\mathbb{E}_{s\sim d(s)}\left[\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\hat{Q}_{\omega_k}(h)\nabla_\theta\log\pi_\theta(a|s)\right]+\varepsilon_{\omega_k}\nabla_\theta\mathscr{H}(\pi_\theta(a|s))\right].\quad(6)$$

In the M-step, we maximise the ELBO with respect to ω while holding the parameters θ_{k+1} constant. Hence expectations are taken with respect to the variational policy found in the E-step: $\omega_{k+1}\leftarrow\arg\max_\omega\mathcal{L}(\omega,\theta_{k+1})$. We use gradient ascent with step size $\alpha_{\text{critic}}(\varepsilon_{\omega_i})^2$ to optimise $\mathcal{L}(\omega,\theta_{k+1})$ to prevent ill-conditioning, yielding (see Appendix E.2 for full derivation):

M-Step (Critic): $\omega_{i+1}\leftarrow\omega_i+\alpha_{\text{critic}}(\varepsilon_{\omega_i})^2\nabla_\omega\mathcal{L}(\omega,\theta_{k+1})|_{\omega=\omega_i}$,

$$(\varepsilon_{\omega_i})^2\nabla_\omega\mathcal{L}(\omega,\theta_{k+1})=\varepsilon_{\omega_i}\mathbb{E}_{d(s)\pi_{\theta_{k+1}}(a|s)}\left[\nabla_\omega\hat{Q}_\omega(h)\right]-\mathbb{E}_{d(s)\pi_{\theta_{k+1}}(a|s)}\left[\hat{Q}_{\omega_i}(h)\right]\nabla_\omega\varepsilon_\omega.\quad(7)$$

4.2 Discussion

From an RL perspective, the E-step corresponds to training an actor using a policy gradient method [56] with an adaptive entropy regularisation term [69, 43]. The M-step update corresponds to a policy evaluation step, as we seek to reduce the MSBE in the second term of Eq. (7). We derive $\nabla_\omega\varepsilon_\omega$ exactly in Appendix E.3. Note that this term depends on $(\mathcal{T}_\omega\hat{Q}_\omega(h)-\hat{Q}_\omega(h))\nabla_\omega\mathcal{T}_\omega\hat{Q}_\omega(h)$, which typically requires evaluating two independent expectations. For convergence guarantees, techniques such as residual gradients [2] or GTD2/TDC [6] need to be employed to obtain an unbiased estimate of this term. If guaranteed convergence is not a priority, dropping gradient terms allows us to use direct methods [55], which are often simpler to implement. We discuss these methods further in Appendix F.3 and provide an analysis in Appendix F.5 demonstrating that the corresponding updates act as a variational approximation to Q-learning [68, 42]. A key component of our algorithm is the behaviour when $\varepsilon_{\omega^*}=0$; under this condition, there is no M-step update (both $\varepsilon_{\omega_k}=0$ and $\nabla_\omega\varepsilon_\omega=0$) and $Q_{\omega^*}(h)=Q^*(h)$ (see Theorem 2), so our E-step reduces exactly to a policy gradient step, $\theta_{k+1}\leftarrow\theta_k+\alpha_{\text{actor}}\mathbb{E}_{h\sim d(s)\pi_\theta(a|s)}\left[Q^*(h)\nabla_\theta\log\pi_\theta(a|s)\right]$, recovering the optimal policy in the limit of convergence, that is, $\pi_\theta(a|s)\rightarrow\pi^*(a|s)$.

From an inference perspective, the E-step improves the parameters of our variational distribution to reduce the gap between the current Boltzmann posterior and the variational policy, $\text{KL}(\pi_\theta(a|s))\parallel\pi_{\omega_k}(a|s))$ (see Theorem 3). This interpretation makes precise the intuition that how much we can improve our policy is determined by how similar $\hat{Q}_{\omega_k}(h)$ is to $Q^*(h)$, limiting policy improvement to the complete E-step: $\pi_{\theta_{k+1}}(a|s) = \pi_{\omega_k}(a|s)$. We see that the common greedy policy improvement step, $\pi_{\theta_{k+1}}(a|s) = \delta(a \in \arg\max_{a'}(\hat{Q}_{\omega_k}(a', s)))$ acts as an approximation to the Boltzmann form in Eq. (3), replacing the softmax with a hard maximum.

If Assumption 4 holds and any constraint induced by $\mathcal{T}_\omega\cdot$ does not prevent convergence to a complete E-step, the EM algorithm alternates between two convex optimisation schemes, and is guaranteed to converge to at least a local optimum of $\mathcal{L}(\omega, \theta)$ [71]. In reality, we cannot carry out complete E- and M-steps for complex domains, and our variational distributions are unlikely to satisfy Assumption 4. Under these conditions, we can resort to the empirically successful variational EM algorithm [30], carrying out partial E- and M-steps instead, which we discuss further in Appendix F.3.

4.3 Advanced Actor-Critic Methods

A family of actor-critic algorithms follows naturally from our framework: 1) we can use powerful inference techniques such as control variates [22] or variance-reducing baselines by subtracting any function that does not depend on the action [50], e.g., $V(s)$, from the action-value function, as this does not change our objective, 2) we can manipulate Eq. (6) to obtain variance-reducing gradient estimators such as EPG [11], FPG [15], and SVG0 [28], and 3) we can take advantage of $d(s)$ being any general decorrelated distribution by using replay buffers [42] or empirically successful asynchronous methods that combine several agents' individual gradient updates at once [43]. As we discuss in Appendix E.4, the manipulation required to derive the estimators in 2) is not strictly justified in the classic policy gradient theorem [56] and MERL formulation [25].

MPO is a state-of-the-art EM algorithm derived from the pseudo-likelihood objective [1]. In its derivation, policy evaluation does not naturally arise from either of its EM steps and must be carried out separately. In addition, its E step is approximated, giving rise to the the one step KL regularised update. As we demonstrate in Appendix G, under the probabilistic interpretation of our model, including a prior of the form $p_\phi(h) = \mathcal{U}(s)\pi_\phi(a|s)$ in our ELBO and specifying a hyper-prior $p(\omega)$, the MPO objective with an adaptive regularisation constant can be recovered from VIREL:

$$\mathcal{L}^{\text{MPO}}(\omega, \theta, \phi) = \mathbb{E}_{s\sim d(s)}\left[\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right] - \text{KL}(\pi_\theta(a|s) \parallel \pi_\phi(a|s))\right] + \log p(\omega).$$

We also show in Appendix G that applying the (variational) EM algorithm from Section 4 yields the MPO updates with the missing policy evaluation step and without approximation in the E-step.

5 Experiments

We evaluate our EM algorithm using the direct method approximation outlined in Appendix F.3 with $\mathcal{T}_\omega$, ignoring constraints on Ω. The aim of our evaluation is threefold: Firstly, as explained in Section 3.1, algorithms using soft value functions cannot be recovered from VIREL. We therefore demonstrate that using hard value functions does not affect performance. Secondly, we provide evidence for our hypothesis stated in Section 3.3 that using soft value functions may harm performance in higher dimensional tasks. Thirdly, we show that even under all practical approximations discussed, the algorithm derived in Section 4 still outperforms advanced actor-critic methods.

We compare our methods to the state-of-the-art SAC[2] and DDPG [38] algorithms on MuJoCo tasks in OpenAI gym [9] and in rllab [14]. We use SAC as a baseline because Haarnoja et al. [25] show that it outperforms PPO [52], Soft Q-Learning [24], and TD3 [18]. We compare to DDPG [38] because, like our methods, it can learn deterministic optimal policies. We consider two variants: In the first one, called *virel*, we keep the scale of the entropy term in the gradient update for the variational policy constant α; in the second, called *beta*, we use an estimate $\hat{\varepsilon}_\omega$ of ε_ω to scale the corresponding term in Eq. (25). We compute $\hat{\varepsilon}_\omega$ using a buffer to draw a fixed number of samples N_ε for the estimate.

To adjust for the relative magnitude of the first term in Eq. (25) with that of ε_ω scaling the second term, we also multiply the estimate $\hat{\varepsilon}_\omega$ by a scalar $\lambda \approx \frac{1-\gamma}{r_{avg}}$, where r_{avg} is the average reward observed; λ^{-1} roughly captures the order of magnitude of the first term and allows $\hat{\varepsilon}_\omega$ to balance policy changes between exploration and exploitation. We found performance is poor and unstable without λ. To reduce variance, all algorithms use a value function network $V(\phi)$ as a baseline and a Gaussian

[2] We use implementations provided by the authors `https://github.com/haarnoja/sac` for v1 and `https://github.com/vitchyr/rlkit` for v2.

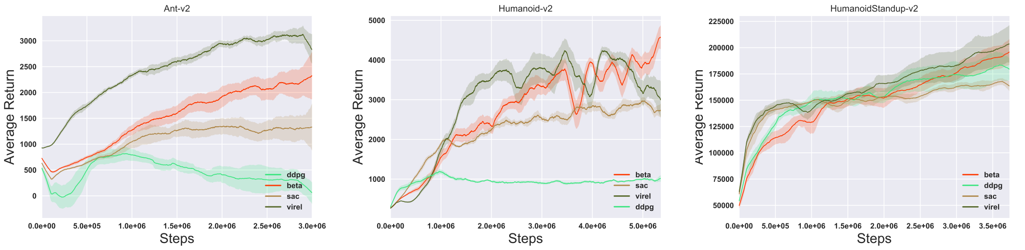

Figure 3: Training curves on continuous control benchmarks gym-Mujoco-v2 : High-dimensional domains.

policy, which enables the use of the reparametrisation trick. Pseudocode can be found in Appendix H. All experiments use 5 random initialisations and parameter values are given in Appendix I.1.

Fig. 3 gives the training curves for the various algorithms on high-dimensional tasks for on gym-mujoco-v2. In particular, in Humanoid-v2 (action space dimensionality: 17, state space dimensionality: 376) and Ant-v2 (action space dimensionality: 8, state space dimensionality: 111), DDPG fails to learn any reasonable policy. We believe that this is because the Ornstein-Uhlenbeck noise that DDPG uses for exploration is insufficiently adaptive in high dimensions. While SAC performs better, *virel* and *beta* still significantly outperform it. As hypothesised in Section 3.3, we believe that this performance advantage arises because the gap between optimal unregularised policies and optimal variational policies learnt under MERLIN is sensitive to temperature c. This effect is exacerbated in high dimensions where there may be large regions of the state-action space with sub-optimal reward. All algorithms learn optimal policies in simple domains, the training curves for which can be found in Fig. 8 in Appendix I.3. Thus, as the state-action dimensionality increases, algorithms derived from VIREL outperform SAC and DDPG.

Fujimoto et al. [18] and van Hasselt et al. [67] note that using the minimum of two randomly initialised action-value functions helps mitigate the positive bias introduced by function approximation in policy gradient methods. Therefore, a variant of SAC uses two soft critics. We compare this variant of SAC to two variants of *virel*: *virel1*, which uses two hard Q-functions and *virel2*, which uses one hard and one soft Q-function. We scale the rewards so that the means of the Q-function estimates in *virel2* are approximately aligned. Fig. 4 shows the training curves on three gym-Mujoco-v1 domains, with additional plots shown in Fig. 7 in Appendix I.2. Again, the results demonstrate that *virel1* and *virel2* perform on par with SAC in simple domains like Half-Cheetah and outperform it in challenging high-dimensional domains like humanoid-gym and -rllab (17 and 21 dimensional action spaces, 376 dimensional state space).

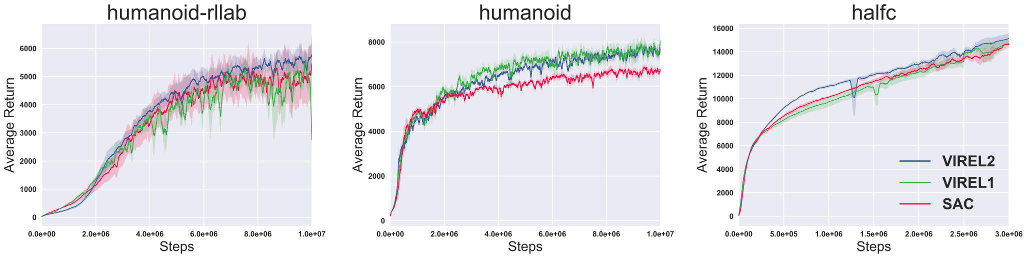

Figure 4: Training curves on continuous control benchmarks gym-Mujoco-v1.

6 Conclusion and Future Work

This paper presented VIREL, a novel framework that recasts the reinforcement learning problem as an inference problem using function approximators. We provided strong theoretical justifications for this framework and compared two simple actor-critic algorithms that arise naturally from applying variational EM on the objective. Extensive empirical evaluation shows that our algorithms perform on par with current state-of-the-art methods on simple domains and substantially outperform them on challenging high dimensional domains. As immediate future work, our focus is to find better estimates of ε_ω to provide a principled method for uncertainty based exploration; we expect it to help attain sample efficiency in conjunction with various methods like [39, 40]. Another avenue of research would extend our framework to multi-agent settings, in which it can be used to tackle the sub-optimality induced by representational constraints used in MARL algorithms [41].

7 Acknowledgements

This project has received funding from the European Research Council (ERC) under the European Unions Horizon 2020 research and innovation programme (grant agreement number 637713). The experiments were made possible by a generous equipment grant from NVIDIA. Matthew Fellows is funded by the EPSRC. Anuj Mahajan is funded by Google DeepMind and the Drapers Scholarship. Tim G. J. Rudner is funded by the Rhodes Trust and the EPSRC. We would like to thank Yarin Gal for helpful comments.

References

[1] Abdolmaleki, A., Springenberg, J. T., Tassa, Y., Munos, R., Heess, N., and Riedmiller, M. Maximum a posteriori policy optimisation. In *International Conference on Learning Representations*, 2018. URL `https://openreview.net/forum?id=S1ANxQW0b`.

[2] Baird, L. Residual algorithms: Reinforcement learning with function approximation. *Machine Learning-International Workshop Then Conference-*, (July):30–37, 1995. ISSN 00043702. doi: 10.1.1.48.3256.

[3] Bass, R. *Real Analysis for Graduate Students*. Createspace Ind Pub, 2013. ISBN 9781481869140. URL `https://books.google.co.uk/books?id=s6mVlgEACAAJ`.

[4] Beal, M. J. *Variational algorithms for approximate Bayesian inference*. PhD thesis, 2003.

[5] Bertsekas, D. *Constrained Optimization and Lagrange Multiplier Methods*. Athena scientific series in optimization and neural computation. Athena Scientific, 1996. ISBN 9781886529045. URL `http://web.mit.edu/dimitrib/www/Constrained-Opt.pdf`.

[6] Bhatnagar, S., Precup, D., Silver, D., Sutton, R. S., Maei, H. R., and Szepesvári, C. Convergent temporal-difference learning with arbitrary smooth function approximation. In Bengio, Y., Schuurmans, D., Lafferty, J. D., Williams, C. K. I., and Culotta, A. (eds.), *Advances in Neural Information Processing Systems 22*, pp. 1204–1212. Curran Associates, Inc., 2009.

[7] Bishop, C. M. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006. ISBN 0387310738.

[8] Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. Variational Inference: A Review for Statisticians, 2017. ISSN 1537274X.

[9] Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., and Zaremba, W. Openai gym. *CoRR*, abs/1606.01540, 2016. URL `http://arxiv.org/abs/1606.01540`.

[10] Ciosek, K. and Whiteson, S. Expected Policy Gradients. *The Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18)*, 2018.

[11] Ciosek, K. and Whiteson, S. Expected policy gradients for reinforcement learning. *journal submission, arXiv preprint arXiv:1801.03326*, 2018.

[12] Dayan, P. and Hinton, G. E. Using expectation-maximization for reinforcement learning. *Neural Computation*, 9(2):271–278, 1997. doi: 10.1162/neco.1997.9.2.271. URL `https://doi.org/10.1162/neco.1997.9.2.271`.

[13] Dempster, A. P., Laird, N. M., and Rubin, D. B. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38, 1977.

[14] Duan, Y., Chen, X., Houthooft, R., Schulman, J., and Abbeel, P. Benchmarking deep reinforcement learning for continuous control. In *International Conference on Machine Learning*, pp. 1329–1338, 2016.

[15] Fellows, M., Ciosek, K., and Whiteson, S. Fourier policy gradients. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1486–1495, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR. URL `http://proceedings.mlr.press/v80/fellows18a.html`.

[16] Foerster, J., Farquhar, G., Al-Shedivat, M., Rocktäschel, T., Xing, E., and Whiteson, S. DiCE: The infinitely differentiable Monte Carlo estimator. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1529–1538, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR. URL `http://proceedings.mlr.press/v80/foerster18a.html`.

[17] Fox, C. W. and Roberts, S. J. A tutorial on variational Bayesian inference. *Artificial Intelligence Review*, pp. 1–11, 2010. ISSN 0269-2821. doi: 10.1007/s10462-011-9236-8. URL `papers2://publication/uuid/1B6D2DDA-67F6-4EEE-9720-2907FFB14789`.

[18] Fujimoto, S., van Hoof, H., and Meger, D. Addressing function approximation error in actor-critic methods. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1587–1596, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR. URL `http://proceedings.mlr.press/v80/fujimoto18a.html`.

[19] Furmston, T. and Barber, D. Variational Methods For Reinforcement Learning. *In AISTATS*, pp. 241–248, 2010. ISSN 15324435.

[20] Geist, M., Scherrer, B., and Pietquin, O. A theory of regularized Markov decision processes. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2160–2169, Long Beach, California, USA, 09–15 Jun 2019. PMLR. URL `http://proceedings.mlr.press/v97/geist19a.html`.

[21] Goyal, A., Brakel, P., Fedus, W., Lillicrap, T. P., Levine, S., Larochelle, H., and Bengio, Y. Recall traces: Backtracking models for efficient reinforcement learning. *CoRR*, abs/1804.00379, 2018. URL `http://arxiv.org/abs/1804.00379`.

[22] Gu, S., Lillicrap, T., Ghahramani, Z., Turner, R. E., and Levine, S. Q-Prop: Sample-Efficient Policy Gradient with An Off-Policy Critic. pp. 1–13, 2016. URL `http://arxiv.org/abs/1611.02247`.

[23] Gunawardana, A. and Byrne, W. Convergence theorems for generalized alternating minimization procedures. *J. Mach. Learn. Res.*, 6:2049–2073, December 2005. ISSN 1532-4435. URL `http://dl.acm.org/citation.cfm?id=1046920.1194913`.

[24] Haarnoja, T., Tang, H., Abbeel, P., and Levine, S. Reinforcement learning with deep energy-based policies. In Precup, D. and Teh, Y. W. (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1352–1361, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR. URL `http://proceedings.mlr.press/v70/haarnoja17a.html`.

[25] Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1861–1870, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR. URL `http://proceedings.mlr.press/v80/haarnoja18b.html`.

[26] Hachiya, H., Peters, J., and Sugiyama, M. Efficient sample reuse in em-based policy search. In Buntine, W., Grobelnik, M., Mladenić, D., and Shawe-Taylor, J. (eds.), *Machine Learning and Knowledge Discovery in Databases*, pp. 469–484, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg. ISBN 978-3-642-04180-8.

[27] Heess, N., Silver, D., and Teh, Y. W. Actor-critic reinforcement learning with energy-based policies. In Deisenroth, M. P., Szepesvári, C., and Peters, J. (eds.), *Proceedings of the Tenth European Workshop on Reinforcement Learning*, volume 24 of *Proceedings of Machine Learning Research*, pp. 45–58, Edinburgh, Scotland, 30 Jun–01 Jul 2013. PMLR. URL `http://proceedings.mlr.press/v24/heess12a.html`.

[28] Heess, N., Wayne, G., Silver, D., Lillicrap, T., Erez, T., and Tassa, Y. Learning continuous control policies by stochastic value gradients. In Cortes, C., Lawrence, N. D., Lee, D. D., Sugiyama, M., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 28*, pp. 2944–2952. Curran Associates, Inc., 2015.

[29] Heess, N., Wayne, G., Silver, D., Lillicrap, T., Tassa, Y., and Erez, T. Learning Continuous Control Policies by Stochastic Value Gradients. pp. 1–13, 2015. ISSN 10495258. URL http://arxiv.org/abs/1510.09142.

[30] Jordan, M. I. (ed.). *Learning in Graphical Models*. MIT Press, Cambridge, MA, USA, 1999. ISBN 0-262-60032-3.

[31] Kelly, J. *Generalized Functions*, chapter 4, pp. 111–124. John Wiley & Sons, Ltd, 2008. ISBN 9783527618897. doi: 10.1002/9783527618897.ch4. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/9783527618897.ch4.

[32] Kingma, D. P. and Welling, M. Auto-Encoding Variational Bayes PPT. *Proceedings of the 2nd International Conference on Learning Representations (ICLR)*, 2014. ISSN 1312.6114v10. URL http://arxiv.org/abs/1312.6114.

[33] Koller, D. and Parr, R. Policy iteration for factored mdps. In *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence*, UAI'00, pp. 326–334, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc. ISBN 1-55860-709-9. URL http://dl.acm.org/citation.cfm?id=2073946.2073985.

[34] Levine, S. *Motor Skill Learning with Trajectory Methods*. PhD thesis, 2014. URL https://people.eecs.berkeley.edu/{~}svlevine/papers/thesis.pdf.

[35] Levine, S. Reinforcement Learning and Control as Probabilistic Inference: Tutorial and Review. 2018. URL https://arxiv.org/pdf/1805.00909.pdf.

[36] Levine, S. and Koltun, V. Variational policy search via trajectory optimization. In Burges, C. J. C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems 26*, pp. 207–215. Curran Associates, Inc., 2013.

[37] Liberzon, D. *Calculus of Variations and Optimal Control Theory: A Concise Introduction*. Princeton University Press, Princeton, NJ, USA, 2011. ISBN 0691151873, 9780691151878.

[38] Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., and Wierstra, D. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

[39] Mahajan, A. and Tulabandhula, T. Symmetry learning for function approximation in reinforcement learning. *arXiv preprint arXiv:1706.02999*, 2017.

[40] Mahajan, A. and Tulabandhula, T. Symmetry detection and exploitation for function approximation in deep rl. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, pp. 1619–1621. International Foundation for Autonomous Agents and Multiagent Systems, 2017.

[41] Mahajan, A., Rashid, T., Samvelyan, M., and Whiteson, S. Maven: Multi-agent variational exploration, 2019. URL https://arxiv.org/abs/1910.07483.

[42] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., and Hassabis, D. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015. ISSN 14764687. doi: 10.1038/nature14236.

[43] Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., Silver, D., and Kavukcuoglu, K. Asynchronous methods for deep reinforcement learning. In Balcan, M. F. and Weinberger, K. Q. (eds.), *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 1928–1937, New York, New York, USA, 20–22 Jun 2016. PMLR. URL http://proceedings.mlr.press/v48/mniha16.html.

[44] Neumann, G. Variational inference for policy search in changing situations. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, pp. 817–824, USA, 2011. Omnipress. ISBN 978-1-4503-0619-5. URL http://dl.acm.org/citation.cfm?id=3104482.3104585.

[45] Pearlmutter, B. A. Fast exact multiplication by the hessian. *Neural Computation*, 6:147–160, 1994.

[46] Peters, J. and Schaal, S. Reinforcement learning by reward-weighted regression for operational space control. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, pp. 745–750, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-793-3. doi: 10.1145/1273496.1273590. URL `http://doi.acm.org/10.1145/1273496.1273590`.

[47] Rawlik, K., Toussaint, M., and Vijayakumar, S. Approximate inference and stochastic optimal control. *CoRR*, abs/1009.3958, 2010. URL `http://arxiv.org/abs/1009.3958`.

[48] Rawlik, K., Toussaint, M., and Vijayakumar, S. On stochastic optimal control and reinforcement learning by approximate inference. In *Robotics: Science and Systems*, 2012.

[49] Sallans, B. and Hinton, G. E. Reinforcement learning with factored states and actions. *J. Mach. Learn. Res.*, 5:1063–1088, dec 2004. ISSN 1532-4435. URL `http://dl.acm.org/citation.cfm?id=1005332.1016794`.

[50] Schulman, J., Heess, N., Weber, T., and Abbeel, P. Gradient estimation using stochastic computation graphs. In Cortes, C., Lawrence, N. D., Lee, D. D., Sugiyama, M., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 28*, pp. 3528–3536. Curran Associates, Inc., 2015.

[51] Schulman, J., Levine, S., Abbeel, P., Jordan, M., and Moritz, P. Trust region policy optimization. 37:1889–1897, 07–09 Jul 2015. URL `http://proceedings.mlr.press/v37/schulman15.html`.

[52] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347, 2017.

[53] Silver, D., Lever, G., Heess, N., Degris, T., Wierstra, D., and Riedmiller, M. Deterministic Policy Gradient Algorithms. *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pp. 387–395, 2014. ISSN 1938-7228.

[54] Sutton, R. S. and Barto, A. G. Sutton & Barto Book: Reinforcement Learning: An Introduction. *MIT Press, Cambridge, MA, A Bradford Book*, 1998. ISSN 10459227. doi: 10.1109/TNN.1998.712192.

[55] Sutton, R. S. and Barto, A. G. *Introduction to Reinforcement Learning*. MIT Press, Cambridge, MA, USA, 2nd edition, 2017. ISBN 0262193981.

[56] Sutton, R. S., Mcallester, D., Singh, S., and Mansour, Y. Policy Gradient Methods for Reinforcement Learning with Function Approximation. *Advances in Neural Information Processing Systems 12*, pp. 1057–1063, 1999. ISSN 0047-2875. doi: 10.1.1.37.9714.

[57] Sutton, R. S., Maei, H. R., Precup, D., Bhatnagar, S., Silver, D., Szepesvári, C., and Wiewiora, E. Fast gradient-descent methods for temporal-difference learning with linear function approximation. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pp. 993–1000, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-516-1. doi: 10.1145/1553374.1553501. URL `http://doi.acm.org/10.1145/1553374.1553501`.

[58] Sutton, R. S., Maei, H. R., and Szepesvári, C. A convergent o(n) temporal-difference algorithm for off-policy learning with linear function approximation. In Koller, D., Schuurmans, D., Bengio, Y., and Bottou, L. (eds.), *Advances in Neural Information Processing Systems 21*, pp. 1609–1616. Curran Associates, Inc., 2009.

[59] Szepesvári, C. Algorithms for Reinforcement Learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 4(1):1–103, 2010. ISSN 1939-4608. doi: 10.2200/S00268ED1V01Y201005AIM009. URL `http://www.morganclaypool.com/doi/abs/10.2200/S00268ED1V01Y201005AIM009`.

[60] Thomas, P. Bias in natural actor-critic algorithms. In Xing, E. P. and Jebara, T. (eds.), *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pp. 441–448, Bejing, China, 22–24 Jun 2014. PMLR. URL `http://proceedings.mlr.press/v32/thomas14.html`.

[61] Todorov, E. Linearly-solvable markov decision problems. In Schölkopf, B., Platt, J. C., and Hoffman, T. (eds.), *Advances in Neural Information Processing Systems 19*, pp. 1369–1376. MIT Press, 2007.

[62] Toussaint, M. Robot trajectory optimization using approximate inference. In *Proceedings of the 26th Annual International Conference on Machine Learning - ICML '09*, pp. 1–8, 2009. ISBN 9781605585161. doi: 10.1145/1553374.1553508. URL `https://homes.cs.washington.edu/{~}todorov/courses/amath579/reading/Toussaint.pdfhttp://portal.acm.org/citation.cfm?doid=1553374.1553508`.

[63] Toussaint, M. Probabilistic inference as a model of planned behavior. *Kunstliche Intelligenz*, 3, 01 2009.

[64] Toussaint, M. and Storkey, A. Probabilistic inference for solving discrete and continuous state Markov Decision Processes. *Proceedings of the 23rd international conference on Machine learning - ICML '06*, pp. 945–952, 2006. doi: 10.1145/1143844.1143963. URL `http://portal.acm.org/citation.cfm?doid=1143844.1143963`.

[65] Tsitsiklis, J. N. and Van Roy, B. An analysis of temporal-difference learning with function approximation. *IEEE Transactions on Automatic Control*, 42(5):674–690, 1997. ISSN 00189286. doi: 10.1109/9.580874.

[66] Turner, R. E. and Sahani, M. *Two problems with variational expectation maximisation for time series models*, pp. 104–124. Cambridge University Press, 2011. doi: 10.1017/CBO9780511984679.006.

[67] van Hasselt, H., Guez, A., and Silver, D. Deep Reinforcement Learning with Double Q-learning. 2015. ISSN 00043702. doi: 10.1016/j.artint.2015.09.002. URL `http://arxiv.org/abs/1509.06461`.

[68] Watkins, C. J. C. H. and Dayan, P. Q-learning. *Machine Learning*, 8(3-4):279–292, 1992. ISSN 0885-6125. doi: 10.1007/BF00992698. URL `http://link.springer.com/10.1007/BF00992698`.

[69] Williams, R. J. and Peng, J. Function optimization using connectionist reinforcement learning algorithms. *Connection Science*, 3(3):241–268, 1991. doi: 10.1080/09540099108946587. URL `https://doi.org/10.1080/09540099108946587`.

[70] Williams, R. J., Baird, L. C., and III. Analysis of some incremental variants of policy iteration: First steps toward understanding actor-critic learning systems, 1993.

[71] Wu, C. F. J. On the Convergence Properties of the EM Algorithm'. *Source: The Annals of Statistics The Annals of Statistics*, 11(1):95–103, 1983.

[72] Yang, Z., Xie, Y., and Wang, Z. A theoretical analysis of deep q-learning. *CoRR*, abs/1901.00137, 2019. URL `http://arxiv.org/abs/1901.00137`.

[73] Ziebart, B. D. *Modeling Purposeful Adaptive Behavior with the Principle of Maximum Causal Entropy*. PhD thesis, 2010. URL `http://www.cs.cmu.edu/{~}bziebart/publications/thesis-bziebart.pdf`.

[74] Ziebart, B. D., Maas, A., Bagnell, J. A., and Dey, A. K. Maximum entropy inverse reinforcement learning. In *Proceedings of the 23rd National Conference on Artificial Intelligence - Volume 3*, AAAI'08, pp. 1433–1438. AAAI Press, 2008. ISBN 978-1-57735-368-3. URL `http://dl.acm.org/citation.cfm?id=1620270.1620297`.

[75] Ziebart, B. D., Bagnell, J. A., and Dey, A. K. Modeling interaction via the principle of maximum causal entropy. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10, pp. 1255–1262, USA, 2010. Omnipress. ISBN 978-1-60558-907-7. URL `http://dl.acm.org/citation.cfm?id=3104322.3104481`.

A A Brief Review of EM and Variational Inference

Fig. 5 shows the representation of a generative graphical model that produces observations x from a distribution $x \sim p_\omega(x|h)$, has hidden variables h, and is parameterised by a set of parameters, ω. In learning a model, we often seek the parameters that maximises the log-marginal-likelihood (LML), which can be found by marginalising the joint distribution $p_\omega(x,h)$ over hidden variables:

$$\ell_\omega(x) \coloneqq \log p_\omega(x) = \log\left(\int_{\mathcal{H}} p_\omega(x,h)dh\right). \tag{8}$$

In many cases, we also need to infer the corresponding posterior,

$$p_\omega(h|x) = \frac{p_\omega(x,h)}{\int_{\mathcal{H}} p_\omega(x,h)dh}.$$

Evaluating the marginal likelihood in Eq. (8) and obtain the corresponding posterior, however, is intractable for most distributions. To compute the marginal likelihood and ω^*, we can use the EM algorithm [13] and variational inference (VI). We review these two methods now.

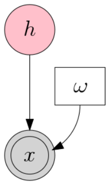

Figure 5: Graphical model of inference problem.

For any valid probability distribution $q(h)$ with support over h we can rewrite the LML as a difference of two divergences [30],

$$\begin{aligned}\ell_\omega(x) =& \int_{\mathcal{H}} q(h)\log\left(\frac{p_\omega(x,h)}{q(h)}\right)dh - \int_{\mathcal{H}} q(h)\log\left(\frac{p_\omega(h|x)}{q(h)}\right)dh,\\ =&\mathcal{L}(\omega, q(h)) + \mathrm{KL}(q(h) \parallel p_\omega(h|x)),\end{aligned} \tag{9}$$

where $\mathcal{L}(\omega, q(h)) \coloneqq \int_{\mathcal{H}} q(h)\log\left(\frac{p_\omega(x,h)}{q(h)}\right)dh$ is known as the evidence lower bound (ELBO). Intuitively, as $\mathrm{KL}(q(h) \parallel p_\omega(h|x)) \geq 0$, it follows that $\ell_\omega(x) \geq \mathrm{ELBO}\,(q(h);\omega)$, hence $\ell_\omega(x) \geq \mathrm{ELBO}\,(q(h);\omega)$ is a lower bound for the LML. The derivation of this bound can also be viewed as applying Jensen's inequality directly to Eq. (8) [8]. Note that when the ELBO and marginal likelihood are identical, the resulting KL divergence between the function $q(h)$ and the posterior $p(h|x)$ is zero, implying that $q(h) = p_\omega(h|x)$.

Maximising the LML now reduces to maximising the ELBO, which can be achieved iteratively using EM [13, 71]; an expectation step (E-step) finds the posterior for the current set of model parameters and then a maximisation step (M-step) maximises the ELBO with respect to ω while keeping $q(h)$ fixed as the posterior from the E-step.

As finding the exact posterior in the E-step is still typically intractable, we resort to variational inference (VI), a powerful tool for approximating the posterior using a parametrised variational distribution $q_\theta(h)$ [30, 4]. VI aims to reduce the KL divergence between the true posterior and the variational distribution, $\mathrm{KL}(q_\theta(h) \parallel p_\omega(h|x))$. Typically VI never brings this divergence to zero but nonetheless yields useful posterior approximations. As minimising $\mathrm{KL}(q_\theta(h) \parallel p_\omega(h|x))$ is equivalent to maximising the ELBO for the variational distribution (see Eq. (23) from Theorem 3), the variational E-step amounts to maximising the ELBO with respect to θ while keeping ω constant. The variational EM algorithm can be summarised as:

$$\text{Variational E-Step: } \theta_{k+1} \leftarrow \arg\max_\theta \mathcal{L}(\omega_k, \theta),$$

$$\text{Variational M-Step: } \omega_{k+1} \leftarrow \arg\max_\omega \mathcal{L}(\omega, \theta_{k+1}).$$

B A Probabilistic Interpretation of VIREL

We now motivate our inference procedure and Boltzmann distribution $\pi_\omega(a|s)$ from a probabilistic perspective, demonstrating that $\pi_\omega(a|s)$ can be interpreted as an action-posterior that characterises the uncertainty our model has in the optimality of $\hat{Q}_\omega(h)$. Moreover, maximising $\mathcal{L}(\omega,\theta)$ for θ is equivalent to carrying our variational inference on the graphical model in Fig. 6 for any $\varepsilon_\omega > 0$.

B.1 Model Specification

Like previous work, we introduce a binary variable $\mathcal{O} \in \{0, 1\}$ in order to define a formal graphical model for our inference problem when $\varepsilon_\omega > 0$. The likelihood of $\mathcal{O}$ therefore takes the form of a Bernoulli distribution:

$$p_\omega(\mathcal{O}|h) = y_\omega(h)^{\mathcal{O}}(1 - y_\omega(h))^{(1-\mathcal{O})},$$

where

$$y_\omega(h) \coloneqq \exp\left(\frac{\hat{Q}_\omega(h) - \max_{a'} \hat{Q}_\omega(a', s)}{\varepsilon_\omega}\right).$$

In most existing frameworks, $\mathcal{O} = 1$ is understood to be the event that the agent is acting optimally [35, 63]. As we are using function approximators in VIREL, $\mathcal{O} = 1$ can be interpreted as the event that the agent is behaving optimally under $\hat{Q}_\omega(h)$. Exploring the semantics of $\mathcal{O}$ further, consider the likelihood when $\mathcal{O} = 1$:

$$p_\omega(\mathcal{O} = 1|h) = \exp\left(\frac{\hat{Q}_\omega(h) - \max_{a'} \hat{Q}_\omega(a', s)}{\varepsilon_\omega}\right),$$

Observe that $0 \le p_\omega(\mathcal{O} = 1|\cdot) \le 1 \ \forall\ \omega \in \Omega\ s.t.\ \varepsilon_\omega > 0$. For any state s, we have $p_\omega(\mathcal{O} = 1|s, a^*) = 1$ for any action a^* that is optimal under $\hat{Q}_\omega(h)$ in the sense that it is the greedy action $a^* \in \arg\max_a \hat{Q}_\omega(h)$. If we find $p_\omega(\mathcal{O} = 1|h) = 1\ \forall\ h \in \mathcal{H}$, then all observed state-action pairs have been generated from a greedy policy $\pi(a|s) = \delta(a \in \arg\max_{a'} \hat{Q}_\omega(a'|s))$. From Theorem 2, the closer the residual error ε_ω is to zero, the closer $\hat{Q}_\omega(h)$ becomes to representing an optimal action-value function. When $\varepsilon_\omega \approx 0$, any a observed such that $p_\omega(\mathcal{O} = 1|a, \cdot) = 1$ will be very nearly an action sampled from an optimal policy, that is $a \sim \pi(a|\cdot) \approx \delta(a \in \arg\max_{a'} Q^*(a'|\cdot))$. We caution readers that in the limit $\varepsilon_\omega \to 0$, our likelihood is not well-defined for any $a \in \arg\max_{a'} \hat{Q}_\omega(a', s)$. Without loss of generality, we condition on optimality for the rest of this section, writing $\mathcal{O}$ in place of $\mathcal{O} = 1$. Defining the function $y_\omega(s) \coloneqq \exp\left(-\frac{\max_{a'} \hat{Q}_\omega(a',s)}{\varepsilon_\omega}\right)$, our likelihood takes the convenient form:

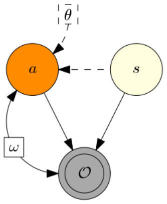

Figure 6: Graphical model for VIREL (variational approximation dashed)

$$p_\omega(\mathcal{O}|h) = \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) y_\omega(s),$$

Defining the prior distribution as the uniform distribution $p(h) = \mathcal{U}(h)$ completes our model, the graph for which is shown in Fig. 6. Using Bayes' rule, we find our posterior distribution is:

$$\begin{aligned} p_\omega(h|\mathcal{O}) &= \frac{p_\omega(\mathcal{O}|h)p(h)}{p_\omega(\mathcal{O})}, \\ &= \frac{p_\omega(\mathcal{O}|h)p(h)}{\int_{\mathcal{H}} p_\omega(\mathcal{O}|h)p(h)dh}, \\ &= \frac{\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) y_\omega(s)}{\int_{\mathcal{H}} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) y_\omega(s) dh}. \end{aligned} \tag{10}$$

We can also derive our action-posterior, $p_\omega(a|s, \mathcal{O})$, which we will find to be equivalent to the Boltzmann policy from Eq. (3). Using Bayes' rule, it follows:

$$p_\omega(a|s, \mathcal{O}) = \frac{p_\omega(h|\mathcal{O})}{p_\omega(s|\mathcal{O})}.$$

Now, we find $p_\omega(s|\mathcal{O})$ by marginalising our posterior over actions. Substituting $p_\omega(s|\mathcal{O}) = \int p_\omega(h|\mathcal{O})da$ yields :

$$p_\omega(a|s, \mathcal{O}) = \frac{p_\omega(h|\mathcal{O})}{\int_{\mathcal{A}} p_\omega(h|\mathcal{O})da}.$$

Substituting for our posterior from Eq. (10), we obtain:

$$
\begin{aligned}
p_\omega(a|s,\mathcal{O}) &= \frac{\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) y_\omega(s)}{\int_\mathcal{A} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) y_\omega(s) da} \cdot \frac{\int_\mathcal{H} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) y_\omega(s) dh}{\int_\mathcal{H} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) y_\omega(s) dh}, \\
&= \frac{\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) y_\omega(s)}{\left(\int_\mathcal{A} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) da\right) y_\omega(s)}, \\
&= \frac{\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right)}{\int_\mathcal{A} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) da}, \\
&= \pi_\omega(a|s),
\end{aligned}
$$

proving that our action-posterior is exactly the Boltzmann policy introduced in Section 3.1. From a Bayesian perspective, the action-posterior $p_\omega(a|s,\mathcal{O})$ characterises the uncertainty we have in deducing the optimal action for a given state s under $\hat{Q}_\omega(h)$; whenever $\varepsilon_\omega \approx 0$ and hence $\hat{Q}_\omega(h) \approx Q^*(h)$, the uncertainty will be very small as $p_\omega(a|s,\mathcal{O})$ will have near-zero variance, approximating a Dirac-delta distribution. Our model is therefore highly confident that the maximum-a-posteriori (MAP) action $a \in \arg\max_{a'} \hat{Q}_\omega(a', s)$ is an optimal action, with all of the probability mass being close to this point. In light of this, we can interpret the greedy policy $\pi_\omega(a|s) = \delta(a \in \arg\max_{a'} \hat{Q}_\omega(a', s))$ as one that always selecting the MAP action across all states.

As our model incorporates the uncertainty in the optimality of $\hat{Q}_\omega(h)$ into the variance of $\pi_\omega(a|s)$, we can benefit directly by sampling trajectories from $\pi_\omega(a|s)$ which drives exploration to gather data that is beneficial to reducing the residual error ε_ω. Unfortunately, calculating the normalisation constant $\int_\mathcal{A} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) da$ is intractable for most function approximators and MDPs of interest. As such, we resort to variational inference, a powerful technique to infer an approximation to a posterior distribution from a tractable family of variational distributions [30, 4, 8]. As before $\pi_\theta(a|s)$ is known as the variational policy, is parametrised by $\theta \in \Theta$ and with the same support as $\pi_\omega(a|s)$. Like in Section 3.1, we define a variational distribution as $q_\theta(h) := d(s)\pi_\theta(a|s)$, where $d(s)$ is an arbitrary sampling distribution with support over $\mathcal{S}$. We fix $d(s)$, as in our model-free paradigm we do not learn the state transition dynamics and only seek to infer the action-posterior.

The goal of variational inference is to find $q_\theta(h)$ closest in KL-divergence to $p_\omega(h|\mathcal{O})$, giving an objective:

$$
\theta^* \in \arg\min_\theta \mathrm{KL}(q_\theta(h) \parallel p_\omega(h|\mathcal{O})).
$$

This objective still requires the intractable computation of $\int \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) y_\omega(s) dh$. Using Eq. (9), we can overcome this by writing the KL divergence in terms of the ELBO:

$$
\mathrm{KL}(q_\theta(h) \parallel p_\omega(h|\mathcal{O})) = \ell_\omega - \mathcal{L}_\omega(\theta),
$$

$$
\text{where} \quad \ell_\omega := \log \int_\mathcal{H} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) y_\omega(s) dh, \quad \mathcal{L}_\omega(\theta) := \mathbb{E}_{h\sim q_\theta(h)}\left[\log\left(\frac{\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) y_\omega(s)}{q_\theta(h)}\right)\right].
$$

We see that minimising the KL-divergence for θ is equivalent to maximising the ELBO for θ, which is tractable. This affords a new objective:

$$
\theta^* \in \arg\max_\theta \mathcal{L}_\omega(\theta).
$$

Expanding the ELBO yields:

$$
\begin{aligned}
\mathcal{L}_\omega(\theta) &= \mathbb{E}_{h\sim q_\theta(h)}\left[\log\left(\frac{\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right)y_\omega(s)}{q_\theta(h)}\right)\right],\\
&= \mathbb{E}_{s\sim d(s)}\left[\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\log\left(\frac{\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right)y_\omega(s)}{q_\theta(h)}\right)\right]\right],\\
&= \mathbb{E}_{s\sim d(s)}\left[\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right]+\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\log y_\omega(s)\right]-\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\log(\pi_\theta(a|s)d(s))\right]\right],\\
&= \mathbb{E}_{s\sim d(s)}\left[\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right]+\log y_\omega(s)-\log d(s)-\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\log\pi_\theta(a|s)\right]\right],\\
&= \mathbb{E}_{s\sim d(s)}\left[\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right]-\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\log\pi_\theta(a|s)\right]\right]+\mathbb{E}_{s\sim d(s)}\left[\log\left(\frac{y_\omega(s)}{d(s)}\right)\right],\\
&= \mathbb{E}_{s\sim d(s)}\left[\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right]+\mathscr{H}(\pi_\theta(a|s))\right]+\mathbb{E}_{s\sim d(s)}\left[\log\left(\frac{y_\omega(s)}{d(s)}\right)\right].
\end{aligned}
$$

As the final term $\mathbb{E}_{s\sim d(s)}\left[\log\left(\frac{y_\omega(s)}{d(s)}\right)\right]$ has no dependency on θ, we can neglect it from our objective, recovering the VIREL objective from Eq. (4):

$$
\mathcal{L}_\omega(\theta) = \mathbb{E}_{s\sim d(s)}\left[\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right]+\mathscr{H}(\pi_\theta(a|s))\right].
$$

Finally, Theorem 3 guarantees that minimising $\mathcal{L}_\omega(\theta)$ always minimises the expected KL divergence between $\pi_\omega(a|s)$ and $\pi_\theta(a|s)$, allowing us to learn a variational approximation for the action-posterior.

C A Discussion of the Target Set $\mathbb{T}$

We now prove that the Bellman operator for the Boltzmann policy, $\mathcal{T}^{\pi_\omega}\boldsymbol{\cdot} := r(h) + \gamma\mathbb{E}_{h'\sim p(s'|h)\pi_\omega(a'|s')}\left[\boldsymbol{\cdot}\right]$, is a member of $\mathbb{T}$. Taking the limit $\varepsilon_\omega \to 0$ of $\mathcal{T}^{\pi_\omega}\hat{Q}_\omega(h)$, we find:

$$
\lim_{\varepsilon_\omega\to 0}\mathcal{T}^{\pi_\omega}\hat{Q}_\omega(h) = r(h) + \lim_{\varepsilon_\omega\to 0}\gamma\mathbb{E}_{h'\sim p(s'|h)\pi_\omega(a'|s')}\left[\hat{Q}_\omega(h')\right].
$$

From Theorem 2, evalutating $\lim_{\varepsilon_\omega\to 0}\gamma\mathbb{E}_{h'\sim p(s'|h)\pi_\omega(a'|s')}\left[\boldsymbol{\cdot}\right]$ recovers a Dirac-delta distribution:

$$
\begin{aligned}
\lim_{\varepsilon_\omega\to 0}\mathcal{T}^{\pi_\omega}\hat{Q}_\omega(h) &= r(h) + \gamma\mathbb{E}_{h'\sim p(s'|h)\delta(a'=\arg\max_a\hat{Q}_\omega(a,s))}\left[\hat{Q}_\omega(h')\right],\\
&= r(h) + \gamma\mathbb{E}_{h'\sim p(s'|h)}\left[\max_{a'}(\hat{Q}_\omega(h'))\right],\\
&= \mathcal{T}^*\hat{Q}_\omega(h).
\end{aligned}
$$

which is sufficient to demonstrate membership of $\mathbb{T}$.

Observe that using $\mathcal{T}^{\pi_\omega}\boldsymbol{\cdot}$ implies $\hat{Q}_\omega(h)$ cannot represent the true Q-function of any $\pi_\omega(a|s)$ except for the optimal Q-function. To see this, imagine there exists some $\varepsilon_\omega > 0$ such that $Q^{\pi_\omega}(\cdot) = \hat{Q}(\cdot)$. Under these conditions, it holds that $\mathcal{T}^{\pi_\omega}\hat{Q}(\cdot) = \hat{Q}(\cdot) \implies \varepsilon_\omega = 0$, which is a contradiction. More generally, as $\pi_\omega(a|s)$ is defined in terms of ε_ω, which itself depends on $\pi_\omega(a|s)$ from the definition of $\mathcal{T}^{\pi_\omega}\boldsymbol{\cdot}$, any ω satisfying this recursive definition forms a constrained set $\Omega^c \subseteq \Omega$. Crucially, we show in Theorem 2 that there always exists some $\omega^* \in \Omega^c$ such that $\hat{Q}_{\omega^*}$ can represent the action-value function for an optimal policy. Note that there may exist other policies that are not Boltzmann distributions such that $\hat{Q}_\omega(h) = Q^\pi(h)$ for some $\omega \in \Omega^c$. We discuss operators that don't constrain Ω in Appendix F.2.

Finally, we can approximate $\mathcal{T}^{\pi_\omega}$ using any TD target sampled from $\pi_\omega(a|s)$ (see Sutton & Barto [55] for an overview of TD methods). Likewise, the optimum Bellman operator $\mathcal{T}^*\cdot = r(h) + \gamma\mathbb{E}_{h'\sim p(s'|h)}\left[\max_{a'}(\cdot)\right]$ is by definition a member of $\mathbb{T}$ and can be approximated using the Q-learning target [68].

D Proofs for Section 3

D.1 Derivation of Lower Bound in terms of KL Divergence

We need to show that

$$\mathcal{L}(\omega,\theta) = \ell(\omega) - \mathrm{KL}(q_\theta(h) \parallel p_\omega(h)) - \mathscr{H}(d(s)), \tag{11}$$

$$\text{where} \quad \ell(\omega) \coloneqq \log \int_{\mathcal{H}} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) dh, \quad p_\omega(h) \coloneqq \frac{\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right)}{\int_{\mathcal{H}} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) dh}.$$

Starting with the LHS of Eq. (11), and recalling the definition of $\mathcal{L}(\omega,\theta)$ from Eq. (4), we have:

$$\mathcal{L}(\omega,\theta) = \mathbb{E}_{s\sim d(s)}\left[\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right] - \mathscr{H}(\pi_\theta(a|s))\right].$$

Expanding the definition of differential entropy:

$$\begin{aligned}
\mathcal{L}(\omega,\theta) &= \mathbb{E}_{s\sim d(s)}\left[\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right] - \mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\log \pi_\theta(a|s)\right]\right],\\
&= \mathbb{E}_{s\sim d(s)}\left[\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right] - \mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\log\left(\frac{\pi_\theta(a|s)d(s)}{p_\omega(h)}\cdot\frac{p_\omega(h)}{d(s)}\right)\right]\right],\\
&= \mathbb{E}_{s\sim d(s)}\Bigg[\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right] - \mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\log\left(\frac{q_\theta(h)}{p_\omega(h)}\right)\right]\\
&\qquad - \mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\log p_\omega(h)\right] + \log(d(s))\Bigg],\\
&= \mathbb{E}_{h\sim q_\theta(h)}\left[\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right] - \mathrm{KL}(q_\theta(h) \parallel p_\omega(h)) - \mathscr{H}(d(s)) - \mathbb{E}_{h\sim q_\theta(h)}\left[\log p_\omega(h)\right].
\end{aligned}$$

Substituting for the definition of $p_\omega(h)$ in the final term yields our desired result:

$$\begin{aligned}
\mathcal{L}(\omega,\theta) &= \mathbb{E}_{h\sim q_\theta(h)}\left[\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right] - \mathrm{KL}(\pi_\theta(a|s) \parallel \pi_\omega(a|s)) - \mathscr{H}(d(s))\\
&\qquad - \mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\log\left(\frac{\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right)}{\int_{\mathcal{H}}\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right)dh}\right)\right]\Bigg],\\
&= \mathbb{E}_{h\sim q_\theta(h)}\left[\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right] - \mathrm{KL}(\pi_\theta(a|s) \parallel \pi_\omega(a|s)) - \mathscr{H}(d(s))\\
&\qquad - \mathbb{E}_{h\sim q_\theta(h)}\left[\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right] + \log\int_{\mathcal{H}}\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right)dh\Bigg],\\
&= \ell(\omega) - \mathrm{KL}(q_\theta(h) \parallel p_\omega(h)) - \mathscr{H}(d(s)).
\end{aligned}$$

D.2 Convergence of Boltzmann Distribution to Dirac-Delta

Theorem 1 (Convergence of Boltzmann Distribution to Dirac Delta)**.** *Let $p_\varepsilon : \mathcal{X} \to [0,1]$ be a Boltzmann distribution with temperature $\varepsilon \in \mathbb{R}_{\geq 0}$*

$$p_\varepsilon(x) = \frac{\exp\left(\frac{f(x)}{\varepsilon}\right)}{\int_{\mathcal{X}}\exp\left(\frac{f(x)}{\varepsilon}\right)dx},$$

where $f : \mathcal{X} \to \mathcal{Y}$ is a function with a unique maximum $f(x^*) = \sup_x f$ and a bounded domain $\mathcal{X}$ and range $\mathcal{Y}$. Let f be locally smooth about x^*, that is $\exists\, \Delta > 0\ s.t. f(x) \in \mathbb{C}^2\ \forall\, x \in \{x \| \|x - x^*\| < \Delta\ \}$. In the limit $\varepsilon \to 0$, $p_\varepsilon(x) \to \delta(x^*)$, that is:

$$\lim_{\varepsilon \to 0} \int_{\mathcal{X}} \varphi(x) p_\varepsilon(x) dx = \varphi(x^*), \tag{12}$$

for any smooth test function $\varphi \in \mathbb{C}_0^\infty(\mathcal{X})$.

Proof. Firstly, we define the auxiliary function to be

$$g(x) \coloneqq f(x) - f(x^*).$$

Note, $g(x) \leq 0$ with equality at $g(x^*) = 0$. Substituting $f(x) = g(x) + f(x^*)$ into $p_\varepsilon(x)$:

$$\begin{aligned} p_\varepsilon(x) =& \frac{\exp\left(\frac{g(x)+f(x^*)}{\varepsilon}\right)}{\int_{\mathcal{X}} \exp\left(\frac{g(x)+f(x^*)}{\varepsilon}\right) dx}, \\ =& \frac{\exp\left(\frac{g(x)}{\varepsilon}\right) \exp\left(\frac{f(x^*)}{\varepsilon}\right)}{\int_{\mathcal{X}} \exp\left(\frac{g(x)}{\varepsilon}\right) \exp\left(\frac{f(x^*)}{\varepsilon}\right) dx}, \\ =& \frac{\exp\left(\frac{g(x)}{\varepsilon}\right)}{\int_{\mathcal{X}} \exp\left(\frac{g(x)}{\varepsilon}\right) dx}. \end{aligned} \tag{13}$$

Now, substituting Eq. (13) into the limit in Eq. (12) yields:

$$\lim_{\varepsilon \to 0} \int_{\mathcal{X}} \varphi(x) p_\varepsilon(x) dx = \lim_{\varepsilon \to 0} \left(\int_{\mathcal{X}} \varphi(x) \frac{\exp\left(\frac{g(x)}{\varepsilon}\right)}{\int_{\mathcal{X}} \exp\left(\frac{g(x)}{\varepsilon}\right) dx} dx \right). \tag{14}$$

Using the substitution $u \coloneqq \frac{(x^* - x)}{\sqrt{\varepsilon}}$ to transform the integrals in Eq. (14), we obtain

$$\begin{aligned} \lim_{\varepsilon \to 0} \int_{\mathcal{X}} \varphi(x) p_\varepsilon(x) dx &= \lim_{\varepsilon \to 0} \left(\int_{\mathcal{U}} \varphi(x^* - \sqrt{\varepsilon} u) \frac{\exp\left(\frac{g(x^* - \sqrt{\varepsilon} u)}{\varepsilon}\right)}{\int_{\mathcal{U}} \exp\left(\frac{g(x^* - \sqrt{\varepsilon} u)}{\varepsilon}\right) \sqrt{\varepsilon} du} \sqrt{\varepsilon} du \right), \\ &= \lim_{\varepsilon \to 0} \left(\frac{\int_{\mathcal{U}} \varphi(x^* - \sqrt{\varepsilon} u) \exp\left(\frac{g(x^* - \sqrt{\varepsilon} u)}{\varepsilon}\right) du}{\int_{\mathcal{U}} \exp\left(\frac{g(x^* - \sqrt{\varepsilon} u)}{\varepsilon}\right) du} \right). \end{aligned} \tag{15}$$

We now find $\lim_{\varepsilon \to 0} \left(\frac{g(x^* - \sqrt{\varepsilon} u)}{\varepsilon}\right)$. Denoting the partial derivative $\partial_{\sqrt{\varepsilon}} \coloneqq \frac{\partial}{\partial \sqrt{\varepsilon}}$ and using L'Hôpital's rule to the second derivative with respect to $\sqrt{\epsilon}$, we find the limit as:

$$\begin{aligned} \lim_{\varepsilon \to 0} \left(\frac{g(x^* - \sqrt{\varepsilon} u)}{\varepsilon} \right) &= \lim_{\varepsilon \to 0} \left(\frac{\partial_{\sqrt{\varepsilon}} g(x^* - \sqrt{\varepsilon} u)}{\partial_{\sqrt{\varepsilon}} \varepsilon} \right), \\ &= \lim_{\varepsilon \to 0} \left(\frac{\partial_{\sqrt{\varepsilon}} f(x^* - \sqrt{\varepsilon} u)}{\partial_{\sqrt{\varepsilon}} \varepsilon} \right), \\ &= \lim_{\varepsilon \to 0} \left(\frac{-u^\top \nabla f(x^* - \sqrt{\varepsilon} u)}{2\sqrt{\varepsilon}} \right), \\ &= \lim_{\varepsilon \to 0} \left(\frac{-\partial_{\sqrt{\varepsilon}} \left(u^\top \nabla f(x^* - \sqrt{\varepsilon} u) \right)}{\partial_{\sqrt{\varepsilon}} (2\sqrt{\varepsilon})} \right), \\ &= \lim_{\varepsilon \to 0} \left(\frac{u^\top \nabla^2 f(x^* - \sqrt{\varepsilon} u) u}{2} \right), \\ &= \frac{u^\top \nabla^2 f(x^*) u}{2}. \end{aligned}$$

The integrand in the numerator in Eq. (15) therefore converges pointwise to $\varphi(x^*) \exp\left(\frac{u^\top \nabla^2 f(x^*) u}{2}\right)$, that is

$$\lim_{\varepsilon \to 0} \left(\varphi(x^* - \sqrt{\varepsilon} u) \exp\left(\frac{g(x^* - \sqrt{\varepsilon} u)}{\varepsilon} \right) \right) = \varphi(x^*) \exp\left(\frac{u^\top \nabla^2 f(x^*) u}{2} \right), \tag{16}$$

and the integrand in the denominator converges pointwise to $\exp\left(\frac{u^\top \nabla^2 f(x^*) u}{2}\right)$, that is

$$\lim_{\varepsilon \to 0} \left(\exp\left(\frac{g(x^* - \sqrt{\varepsilon} u)}{\varepsilon} \right) \right) = \exp\left(\frac{u^\top \nabla^2 f(x^*) u}{2} \right). \tag{17}$$

From the second order sufficient conditions for $f(x^*)$ to be a maximum, we have $u^\top \nabla^2 f(x^*) u \leq 0$ $\forall\ u \in \mathcal{U}$ with equality only when $u = 0$ [37]. This implies that Eq. (16) and Eq. (17) are both bounded functions.

By definition, we have $g(x^* - \sqrt{\epsilon} u) \leq 0\ \forall\ u \in \mathcal{U}$, which implies that $|\exp\left(\frac{g(x^* - \sqrt{\varepsilon} u)}{\varepsilon}\right)| \leq 1$. Consequently, the integrand in the numerator of Eq. (15) is dominated by $\|\varphi(\cdot)\|_\infty$, that is

$$\left| \varphi(x^* - \sqrt{\varepsilon} u) \exp\left(\frac{g(x^* - \sqrt{\varepsilon} u)}{\varepsilon} \right) \right| \leq \|\varphi(\cdot)\|_\infty, \tag{18}$$

and the integrand in the denominator is dominated by 1, that is

$$\left| \exp\left(\frac{g(x^* - \sqrt{\varepsilon} u)}{\varepsilon} \right) \right| \leq 1. \tag{19}$$

Together Eqs. (16) to (19) are the sufficient conditions for applying the dominated convergence theorem [3], allowing us to commute all limits and integrals in Eq. (15), yielding our desired result:

$$\begin{aligned}
\lim_{\varepsilon \to 0} \int_{\mathcal{X}} \varphi(x) p_\varepsilon(x) dx &= \lim_{\varepsilon \to 0} \left(\frac{\int_{\mathcal{U}} \varphi(x^* - \sqrt{\varepsilon} u) \exp\left(\frac{g(x^* - \sqrt{\varepsilon} u)}{\varepsilon} \right) du}{\int_{\mathcal{U}} \exp\left(\frac{g(x^* - \sqrt{\varepsilon} u)}{\varepsilon} \right) du} \right), \\
&= \frac{\int_{\mathcal{U}} \lim_{\varepsilon \to 0} \left(\varphi(x^* - \sqrt{\varepsilon} u) \exp\left(\frac{g(x^* - \sqrt{\varepsilon} u)}{\varepsilon} \right) \right) du}{\int_{\mathcal{U}} \lim_{\varepsilon \to 0} \left(\exp\left(\frac{g(x^* - \sqrt{\varepsilon} u)}{\varepsilon} \right) \right) du}, \\
&= \frac{\int_{\mathcal{U}} \varphi(x^*) \exp\left(u^\top \nabla^2 f(x^*) u \right) du}{\int_{\mathcal{U}} \exp\left(u^\top \nabla^2 f(x^*) u \right) du}, \\
&= \varphi(x^*) \frac{\int_{\mathcal{U}} \exp\left(u^\top \nabla^2 f(x^*) u \right) du}{\int_{\mathcal{U}} \exp\left(u^\top \nabla^2 f(x^*) u \right) du}, \\
&= \varphi(x^*).
\end{aligned}$$

□

D.3 Optimal Boltzmann Distributions as Optimal Policies

Lemma 1 (Lower and Upper limits of $\mathcal{L}(\omega, \theta)$)**.** *i) For any $\varepsilon_\omega > 0$ and $\pi_\theta(a|s) = \delta(a^*)$, we have $\mathcal{L}(\omega, \theta) = -\infty$. ii) For $\hat{Q}_\omega(\cdot) > 0$ and any non-deterministic $\pi_\theta(a|s)$, $\lim_{\varepsilon_\omega \to 0} \mathcal{L}(\omega, \theta) = \infty$.*

Proof. To prove i), we substitute $\pi_\theta(a|s) = \delta(a^*)$ into $\mathcal{L}(\omega, \theta)$ from Eq. (4), yielding:

$$\begin{aligned}
\mathcal{L}(\omega, \theta) &= \mathbb{E}_{s \sim d(s)} \left[\mathbb{E}_{a \sim \delta(a^*)} \left[\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega} \right] + \mathcal{H}(\delta(a^*)) \right], \\
&= \mathbb{E}_{s \sim d(s)} \left[\frac{\hat{Q}_\omega(a^*, s)}{\varepsilon_\omega} + \mathcal{H}(\delta(a^*)) \right],
\end{aligned} \tag{20}$$

We now prove that $\mathscr{H}(\delta(a^*)) = -\infty$ for any a^*. Let $p : \mathcal{X} \to [0, 1]$ be any zero-mean, unit variance distribution. Using a transformation of variables, we have $\mathcal{A} = \sigma\mathcal{X} + a^*$ and hence $p(a) = \frac{1}{\sigma}p(\sigma x - a^*)$. We can therefore write our Dirac-delta distribution as

$$\delta(a^*) = \lim_{\sigma\to 0} p(a) = \lim_{\sigma\to 0} \frac{1}{\sigma}p(\sigma x - a^*).$$

Substituting into the definition of differential entropy, we obtain:

$$\begin{aligned}
\mathscr{H}(\delta(a^*)) &= \lim_{\sigma\to 0} \mathscr{H}(p(a)) \\
&= \lim_{\sigma\to 0} \mathscr{H}\left(\frac{1}{\sigma}p(\sigma x - a^*)\right), \\
&= -\lim_{\sigma\to 0} \int_{\mathcal{A}} \frac{1}{\sigma}p(\sigma x - a^*) \log\left(\frac{1}{\sigma}p(\sigma x - a^*)\right) da, \\
&= -\lim_{\sigma\to 0} \int_{\mathcal{A}} \frac{1}{\sigma}p(\sigma x - a^*) \log\left(p(\sigma x - a^*)\right) da + \lim_{\sigma\to 0} \int_{\mathcal{A}} \frac{1}{\sigma}p(\sigma x - a^*) \log\left(\sigma\right) da, \\
&= -\int_{\mathcal{A}} \delta(a^*) \log\left(p(-a^*)\right) da + \lim_{\sigma\to 0} \log\left(\sigma\right), \\
&= -\log(p(-a^*)) + \lim_{\sigma\to 0} \log\left(\sigma\right), \qquad (21) \\
&= -\infty.
\end{aligned}$$

Substituting for $\mathscr{H}(\delta(a^*))$ from Eq. (21) in Eq. (20) yields our desired result:

$$\begin{aligned}
\mathcal{L}(\omega, \theta) &= \mathbb{E}_{s\sim d(s)}\left[\frac{\hat{Q}_\omega(a^*, s)}{\varepsilon_\omega}\right] + \mathbb{E}_{s\sim d(s)}\left[\mathcal{H}(\delta(a^*))\right], \\
&= \frac{\mathbb{E}_{s\sim d(s)}\left[\hat{Q}_\omega(a^*, s)\right]}{\varepsilon_\omega} + (\lim_{\sigma\to 0} \log\left(\sigma\right) - \log(p(-a^*)))\mathbb{E}_{s\sim d(s)}\left[1\right], \\
&= -\infty,
\end{aligned}$$

where our final line follows from the first term being finite for any $\varepsilon_\omega > 0$.

To prove ii), we take the limit $\varepsilon_\omega \to 0$ of $\mathcal{L}(\omega, \theta)$ in Eq. (4):

$$\begin{aligned}
\lim_{\varepsilon_\omega\to 0} \mathcal{L}(\omega, \theta) &= \lim_{\varepsilon_\omega\to 0}\left(\frac{\mathbb{E}_{d(s)\pi_\theta(a|s)}\left[\hat{Q}_\omega(h)\right]}{\varepsilon_\omega} + \mathbb{E}_{d(s)}\left[\mathscr{H}(\pi_\theta(a|s))\right]\right), \\
&= \lim_{\varepsilon_\omega\to 0}\left(\frac{\mathbb{E}_{d(s)\pi_\theta(a|s)}\left[\hat{Q}_\omega(h)\right]}{\varepsilon_\omega}\right) + \mathbb{E}_{d(s)}\left[\mathscr{H}(\pi_\theta(a|s))\right], \\
&= \infty.
\end{aligned}$$

where our last line follows from $\mathscr{H}(\pi_\theta(a|s))$ being finite for any non-deterministic $\pi_\theta(a|s)$ and $\hat{Q}_\omega(\cdot) > 0 \implies \mathbb{E}_{d(s)\pi_\theta(a|s)}\left[\hat{Q}_\omega(h)\right] > 0$.

□

Theorem 2 (Optimal Boltzmann Distributions as Optimal Policies)**.** *For any pair* $\{\omega^*, \theta^*\}$ *that maximises* $\mathcal{L}(\omega, \theta)$ *defined in Eq.* (4)*, the corresponding variational policy induced must be optimal, i.e.* $\{\omega^*, \theta^*\} \in \arg\max_{\omega,\theta} \mathcal{L}(\omega, \theta) \implies \pi_{\omega^*}(a|s) \in \Pi^*$. *Moreover, any* θ^* *s.t.* $\pi_{\theta^*}(a|s) = \pi_{\omega^*}(a|s) \implies \theta^* \in \arg\max_{\omega,\theta} \mathcal{L}(\omega, \theta)$.

Proof. Our proof is structured as follows: Firstly, we prove that $\varepsilon_{\omega^*} = 0$ is both a necessary and sufficient condition for any $\omega^* \in \arg\max_{\omega,\theta} \mathcal{L}(\omega, \theta)$ with $\hat{Q}_{\omega^*}(\cdot) > 0$. We then verify that $\hat{Q}_{\omega^*}(\cdot) > 0$ is satisfied by our framework and $\varepsilon_{\omega^*} = 0$ is feasible. Finally, we prove that $\varepsilon_{\omega^*} = 0$ is sufficient for $\pi_{\omega^*}(a|s) \in \Pi^*$.

To prove necessity, assume there exists an optimal ω^* such that $\varepsilon_{\omega^*} \neq 0$. As $\varepsilon_\omega \geq 0$, it must be that $\varepsilon_{\omega^*} > 0$. Consider $\mathcal{L}(\omega, \theta)$ as defined in Eq. (4):

$$\mathcal{L}(\omega, \theta) = \frac{\mathbb{E}_{d(s)\pi_\theta(a|s)}\left[\hat{Q}_\omega(h)\right]}{\varepsilon_\omega} + \mathbb{E}_{d(s)}\left[\mathscr{H}(\pi_\theta(a|s))\right].$$

As $\pi_\theta(a|s)$ has finite variance, $\mathscr{H}(\pi_\theta(a|s))$ is upper bounded, and as $\hat{Q}_\omega(\cdot)$ is upper bounded, $\mathbb{E}_{d(s)\pi_\theta(a|s)}\left[\hat{Q}_\omega(h)\right]$ is upper bounded too. Together, this implies that $\mathbb{E}_{d(s)\pi_\theta(a|s)}\left[\hat{Q}_\omega(h)\right]$ is upper bounded for $\varepsilon_{\omega^*} > 0$. From Assumption 2, there exists $\omega^\diamond \in \Omega$ such that $\varepsilon_{\omega^\diamond} = 0$. From Lemma 1, there exists θ^* such that $\lim_{\varepsilon_{\omega^*} \to 0} \mathcal{L}(\omega^\diamond, \theta^*) = \infty$, implying $\mathcal{L}(\omega^*, \theta^*) < \mathcal{L}(\omega^\diamond, \theta^*)$ which is a contradiction.

To prove sufficiency, we take $\arg\max_\omega \mathcal{L}(\omega, \theta)$:

$$\begin{aligned}
\arg\max_\omega \mathcal{L}(\omega, \theta) &= \arg\max_\omega \left(\mathbb{E}_{d(s)\pi_\theta(a|s)}\left[\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right] + \mathbb{E}_{d(s)}\left[\mathscr{H}(\pi_\theta(a|s))\right]\right), \\
&= \arg\max_\omega \left(\mathbb{E}_{d(s)\pi_\theta(a|s)}\left[\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right]\right), \\
&= \arg\max_\omega \left(\frac{\mathbb{E}_{d(s)\pi_\theta(a|s)}\left[\hat{Q}_\omega(h)\right]}{\varepsilon_\omega}\right).
\end{aligned}$$

Assume that $\textcircled{i}$ $\hat{Q}_{\omega^*}(\cdot) > 0$. It then follows:

$$\begin{aligned}
\arg\max_\omega \mathcal{L}(\omega, \theta) &= \arg\max_\omega \left(\frac{\mathbb{E}_{d(s)\pi_\theta(a|s)}\left[\hat{Q}_\omega(h)\right]}{\varepsilon_\omega}\right), \\
& \arg\min_\omega \left(\frac{\varepsilon_\omega}{\mathbb{E}_{d(s)\pi_\theta(a|s)}\left[\hat{Q}_\omega(h)\right]}\right), \\
&= \arg\min_\omega \varepsilon_\omega,
\end{aligned}$$

which, as $\varepsilon_\omega \geq 0$, is satisfied for any $\omega^* \in \Omega \; s.t. \; \varepsilon_{\omega^*} = 0$, proving sufficiency.

Assume now $\textcircled{ii}$ $\hat{Q}_{\omega^*}(\cdot)$ is locally smooth with a unique maximum over actions according to Definition 1. Under this condition we can apply Theorem 1 and our Boltzmann distribution tends towards a Dirac-delta function:

$$\pi_{\omega^*}(a|s) = \lim_{\varepsilon_\omega \to 0}\left(\frac{\exp\left(\frac{\hat{Q}_{\omega^*}(h)}{\varepsilon_\omega}\right)}{\int \exp\left(\frac{\hat{Q}_{\omega^*}(h)}{\varepsilon_\omega}\right) da}\right) = \delta(a = \arg\max_{a'} \hat{Q}_{\omega^*}(s, a')), \tag{22}$$

which is a greedy policy w.r.t. $\hat{Q}_{\omega^*}(\cdot)$. From Definition 2, when $\lim_{\varepsilon_\omega \to 0} \pi_\omega(a|s)$ we have $\mathcal{T}_\omega \hat{Q}_\omega(h) = \mathcal{T}^* \hat{Q}_\omega(h)$. Substituting into $\varepsilon_{\omega^*} = 0$ shows our our function approximator must satisfy an optimal Bellman equation:

$$\begin{aligned}
\varepsilon_{\omega^*} &= \frac{c}{p}\|\mathcal{T}^*\hat{Q}_\omega(h) - \hat{Q}_\omega(h)\|_p^p = 0, \\
&\implies \mathcal{T}^*\hat{Q}_{\omega^*}(\cdot) = \hat{Q}_{\omega^*}(\cdot),
\end{aligned}$$

hence $\hat{Q}_{\omega^*}(\cdot) = Q^*(\cdot)$. Under Assumption 2, we see that there exists $\omega^* \in \Omega \; s.t. \; \varepsilon_{\omega^*} = 0$ for $\hat{Q}_{\omega^*}(\cdot) = Q^*(\cdot)$, hence $\varepsilon_{\omega^*} = 0$ is feasible. Moreover, our assumptions $\textcircled{i}$ and $\textcircled{ii}$ are satisfied for

$\hat{Q}_{\omega^*}(\cdot) = Q^*(\cdot)$ under Assumptions 2 and 3 respectively. Substituting for $\hat{Q}_{\omega^*}(\cdot) = Q^*(\cdot)$ into $\pi_{\omega^*}(a|s)$ from Eq. (22) we recover our desired result:

$$\omega^* \in \arg\max_{\omega} \mathcal{L}(\omega, \theta)$$
$$\implies \pi_{\omega^*}(a|s) = \delta(a = \arg\max_{a'} Q^*(s, a')) \in \Pi^*.$$

From Lemma 1, we have that $\mathcal{L}(\omega, \theta) \to \infty = \max_{\omega,\theta} \mathcal{L}(\omega, \theta)$ when $\varepsilon_\omega = 0$ for any $\theta^* \in \Theta$ such that the variational policy is non-deterministic, hence

$$\{\omega^*, \theta^*\} \in \arg\max_{\omega,\theta} \mathcal{L}(\omega, \theta) \implies \pi_{\omega^*}(a|s) \in \Pi^*,$$

as required. □

D.4 Maximising the ELBO for θ

Theorem 3 (Maximising the ELBO for θ)**.** *Maximsing $\mathcal{L}(\omega, \theta)$ for θ with $\varepsilon_\omega > 0$ is equivalent to minimising the expected KL divergence between $\pi_\omega(a|s)$ and $\pi_\theta(a|s)$, i.e. for any $\varepsilon_\omega > 0$, $\max_\theta \mathcal{L}(\omega, \theta) = \min_\theta \mathbb{E}_{d(s)} [\mathrm{KL}(\pi_\theta(a|s) \parallel \pi_\omega(a|s))]$ with $\pi_\omega(a|s) = \pi_\theta(a|s)$ under exact representability.*

Proof. Firstly, we write $\mathcal{L}(\omega, \theta)$ in terms of $\ell(\omega)$ and $\mathrm{KL}(q_\theta(h) \parallel p_\omega(h))$ from Eq. (5):

$$\mathcal{L}(\omega, \theta) = \ell(\omega) - \mathrm{KL}(q_\theta(h) \parallel p_\omega(h)),$$

which implies

$$\begin{aligned}\max_\theta \mathcal{L}(\omega, \theta) &= \max_\theta \left(\ell(\omega) - \mathrm{KL}(q_\theta(h) \parallel p_\omega(h))\right), \\ &= \min_\theta \left(\mathrm{KL}(q_\theta(h) \parallel p_\omega(h))\right). \end{aligned} \tag{23}$$

for any $\varepsilon_\omega > 0$. Define

$$p_\omega(s) := \frac{\int_{\mathcal{A}} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) da}{\int_{\mathcal{H}} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) dh}.$$

We now decompose $p_\omega(h)$ as $p_\omega(h) := \pi_\omega(a|s) p_\omega(s)$:

$$\begin{aligned} p_\omega(h) &= \frac{\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right)}{\int_{\mathcal{H}} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) dh}, \\ &= \frac{\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right)}{\int_{\mathcal{H}} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) dh} \cdot \frac{\int_{\mathcal{A}} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) da}{\int_{\mathcal{A}} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) da}, \\ &= \frac{\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right)}{\int_{\mathcal{A}} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) da} \cdot \frac{\int_{\mathcal{A}} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) da}{\int_{\mathcal{H}} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) dh}, \\ &= \pi_\omega(a|s) p_\omega(s). \end{aligned}$$

Substituting for $p_\omega(h) = \pi_\omega(a|s) p_\omega(s)$ and $q_\theta(h) = d(s)\pi_\theta(a|s)$ into the KL divergence from Eq. (23) yields:

$$\begin{aligned} \mathrm{KL}(q_\theta(h) \parallel p_\omega(h)) &= \mathbb{E}_{d(s)\pi_\theta(a|s)}\left[\log\left(\frac{d(s)\pi_\theta(a|s)}{p_\omega(s)\pi_\omega(a|s)}\right)\right], \\ &= \mathbb{E}_{d(s)\pi_\theta(a|s)}\left[\log\left(\frac{d(s)}{p_\omega(s)}\right)\right] + \mathbb{E}_{d(s)\pi_\theta(a|s)}\left[\log\left(\frac{\pi_\theta(a|s)}{\pi_\omega(a|s)}\right)\right], \\ &= \mathbb{E}_{d(s)}\left[\log\left(\frac{d(s)}{p_\omega(s)}\right)\right] \mathbb{E}_{\pi_\theta(a|s)}[1] + \mathbb{E}_{d(s)\pi_\theta(a|s)}\left[\log\left(\frac{\pi_\theta(a|s)}{\pi_\omega(a|s)}\right)\right], \\ &= \mathbb{E}_{d(s)}\left[\log\left(\frac{d(s)}{p_\omega(s)}\right)\right] + \mathbb{E}_{d(s)}\left[\mathbb{E}_{\pi_\theta(a|s)}\left[\log\left(\frac{\pi_\theta(a|s)}{\pi_\omega(a|s)}\right)\right]\right], \\ &= \mathrm{KL}(d(s) \parallel p_\omega(s)) + \mathbb{E}_{d(s)}\left[\mathrm{KL}(\pi_\theta(a|s) \parallel \pi_\omega(a|s))\right]. \end{aligned} \tag{24}$$

Observe that the first term in Eq. (24) does not depend on θ, hence taking the minimum yields our desired result:

$$\max_{\theta}\mathcal{L}(\omega,\theta)=\min_{\theta}\left(\mathrm{KL}(d(s)\parallel p_\omega(s))+\mathbb{E}_{d(s)}\left[\mathrm{KL}(\pi_\theta(a|s)\parallel\pi_\omega(a|s))\right]\right),$$
$$=\min_{\theta}\mathbb{E}_{d(s)}\left[\mathrm{KL}(\pi_\theta(a|s)\parallel\pi_\omega(a|s))\right].$$

Since $\mathrm{KL}(\pi_\theta(a|s)\parallel\pi_\omega(a|s))\geq 0$, it follows that under exact representability, that is there exists $\theta\in\Theta$ s.t. $\pi_\theta(a|s)=\pi_\omega(a|s)$ and hence $\mathrm{KL}(\pi_\theta(a|s)\parallel\pi_\omega(a|s))=0$, we have $\min_\theta\mathbb{E}_{d(s)}\left[\mathrm{KL}(\pi_\theta(a|s)\parallel\pi_\omega(a|s))\right]=0$. □

E Deriving the EM Algorithm

E.1 E-Step

Here we provide a full derivation of our E-step of our variational actor-critic algorithm. The ELBO for our model from Eq. (4) with ω_k fixed is:

$$\mathcal{L}(\omega_k,\theta)=\mathbb{E}_{s\sim d(s)}\left[\frac{\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\hat{Q}_{\omega_k}(h)\right]}{\varepsilon_{\omega_k}}+\mathscr{H}(\pi_\theta(a|s))\right].$$

Taking derivatives of the with respect to θ yields:

$$\nabla_\theta\mathcal{L}(\omega_k,\theta)=\mathbb{E}_{s\sim d(s)}\left[\frac{\nabla_\theta\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\hat{Q}_{\omega_k}(h)\right]}{\varepsilon_{\omega_k}}+\nabla_\theta\mathscr{H}(\pi_\theta(a|s))\right],$$
$$=\mathbb{E}_{s\sim d(s)}\left[\frac{\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\hat{Q}_{\omega_k}(h)\nabla_\theta\log\pi_\theta(a|s)\right]}{\varepsilon_{\omega_k}}+\nabla_\theta\mathscr{H}(\pi_\theta(a|s))\right],$$

where we have used the log-derivative trick [56] in deriving the final line. Note that in this form, when $\varepsilon_{\omega_k}\approx 0$, our gradient signal becomes very large. To prevent ill-conditioning, we multiply our objective by the constant ε_{ω_k}. As $\varepsilon_{\omega_k}>0$ for all non-optimal ω_k (see Theorem 2), this will not change the solution to the E-step optimisation. Our gradient becomes:

$$\varepsilon_{\omega_k}\nabla_\theta\mathcal{L}(\omega_k,\theta)=\mathbb{E}_{s\sim d(s)}\left[\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\hat{Q}_{\omega_k}(h)\nabla_\theta\log\pi_\theta(a|s)\right]+\varepsilon_{\omega_k}\nabla_\theta\mathscr{H}(\pi_\theta(a|s))\right],\tag{25}$$

as required.

E.2 M-Step

Here we provide a full derivation of our M-step of our variational actor-critic algorithm. The ELBO for our model from Eq. (4) with θ_{k+1} fixed is:

$$\mathcal{L}(\omega,\theta_{k+1})=\mathbb{E}_{d(s)}\left[\frac{\mathbb{E}_{\pi_{\theta_{k+1}}(a|s)}\left[\hat{Q}_{\omega}(h)\right]}{\varepsilon_{\omega}}+\mathscr{H}(\pi_{\theta_{k+1}}(a|s))\right]$$

Taking derivatives of the with respect to ω yields:

$$\nabla_\omega\mathcal{L}(\omega,\theta_{k+1})=\mathbb{E}_{d(s)\pi_{\theta_{k+1}}(a|s)}\left[\nabla_\omega\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right)\right],$$
$$=\mathbb{E}_{d(s)\pi_{\theta_{k+1}}(a|s)}\left[\frac{\nabla_\omega\hat{Q}_\omega(h)}{\varepsilon_\omega}-\frac{\hat{Q}_\omega(h)}{(\varepsilon_\omega)^2}\nabla_\omega\varepsilon_\omega\right],$$
$$=\frac{1}{\varepsilon_\omega}\mathbb{E}_{d(s)\pi_{\theta_{k+1}}(a|s)}\left[\nabla_\omega\hat{Q}_\omega(h)\right]-\frac{1}{(\varepsilon_\omega)^2}\mathbb{E}_{d(s)\pi_{\theta_{k+1}}(a|s)}\left[\hat{Q}_\omega(h)\right]\nabla_\omega\varepsilon_\omega,$$

where we note that ε_ω does not depend on h, which allowed us to move it in and out of the expectation in deriving the final line. The gradient depends on terms up to $\frac{1}{(\varepsilon_\omega)^2}$, and so we multiply our objective by $(\varepsilon_{\omega_i})^2$ to prevent ill-conditioning when $\varepsilon_\omega \approx 0$. As $(\varepsilon_{\omega_i})^2 > 0$ for all non-convergent ω^*, this does not change the solution to our M-step optimisation and can be seen as introducing an adaptive step size which supplements α_{critic}. Observe that $\frac{\varepsilon_{\omega_i}}{\varepsilon_\omega}\Big|_{\omega=\omega_i} = 1$, which, with a slight abuse of notation, yields our desired result:

$$(\varepsilon_{\omega_i})^2\nabla_\omega\mathcal{L}(\omega,\theta_{k+1}) = \varepsilon_{\omega_i}\mathbb{E}_{d(s)\pi_{\theta_{k+1}}(a|s)}\left[\nabla_\omega\hat{Q}_\omega(h)\right] - \mathbb{E}_{d(s)\pi_{\theta_{k+1}}(a|s)}\left[\hat{Q}_\omega(h)\right]\nabla_\omega\varepsilon_\omega.$$

In general, calculating the exact gradient of ε_ω is non trivial. We now derive this update for three important cases:

E.3 Gradient of the Residual Error

We define $\beta_\omega(h) \coloneqq \mathcal{T}_\omega\hat{Q}_\omega(h) - \hat{Q}_\omega(h)$ and use the notation $\mathbb{E}[\cdot] \triangleq \mathbb{E}_{h\sim\mathcal{U}(h)}[\cdot]$. Taking the derivative yields:

$$\begin{aligned}\nabla_\omega\varepsilon_\omega &= \frac{1}{2|\mathcal{H}|}\nabla_\omega\|\beta_\omega(h)^2\|_2^2,\\ &= \frac{1}{2}\nabla_\omega\mathbb{E}\left[\beta_\omega(h)^2\right],\\ &= \mathbb{E}\left[\beta_\omega(h)\nabla_\omega\beta_\omega(h)\right].\end{aligned} \tag{26}$$

For targets that do not depend on $\pi_\omega(a|s)$, the gradient of $\nabla_\omega\beta_\omega(h)$ can be computed directly. As an example, consider the update for the Q-learning target:

$$\nabla_\omega\beta_\omega(h) = \mathbb{E}_{s'\sim p(s'|h)}\left[\nabla_\omega\hat{Q}_\omega(a^*,s')\right] - \nabla_\omega\hat{Q}_\omega(h),$$

where $a^* = \arg\max_a \hat{Q}(a,s')$.

For convenience, we denote the expectation $\mathbb{E}_{h'\sim p(s'|h)\pi_\omega(a'|s')}\left[\cdot\right]$ as $\mathbb{E}_\omega\left[\cdot\right]$. For the Bellman operator target $\mathcal{T}^{\pi_\omega}\hat{Q}_\omega(h) = r(h) + \gamma\mathbb{E}_\omega\left[\hat{Q}_\omega(h')\right]$ that depends on $\pi_\omega(a|s)$, we must solve a recursive equation for $\nabla_\omega\pi_\omega(a|s)$. Consider the gradient of $\beta_\omega(h)$ using $\mathcal{T}^{\pi_\omega}\cdot$:

$$\begin{aligned}\nabla_\omega\beta_\omega(h) &= \nabla_\omega\left(r(h) + \gamma\mathbb{E}_\omega\left[\hat{Q}_\omega(h')\right] - \hat{Q}_\omega(h)\right),\\ &= \nabla_\omega\gamma\mathbb{E}_\omega\left[\hat{Q}_\omega(h')\right] - \nabla_\omega\hat{Q}_\omega(h),\\ &= \gamma\mathbb{E}_\omega\left[(\nabla_\omega\log\pi_\omega(a'|s'))\hat{Q}_\omega(h') + \nabla_\omega\hat{Q}_\omega(h')\right] - \nabla_\omega\hat{Q}_\omega(h),\\ &= \gamma\mathbb{E}_\omega\left[(\nabla_\omega\log\pi_\omega(a'|s'))\hat{Q}_\omega(h')\right] + \gamma\mathbb{E}_\omega\left[\nabla_\omega\hat{Q}_\omega(h')\right] - \nabla_\omega\hat{Q}_\omega(h),\\ &= \gamma\mathbb{E}_\omega\left[(\nabla_\omega\log\pi_\omega(a'|s'))\hat{Q}_\omega(h')\right] + \Gamma_\omega(h),\end{aligned} \tag{27}$$

where $\Gamma_\omega(h) \coloneqq \gamma\mathbb{E}_\omega\left[\nabla_\omega\hat{Q}_\omega(h')\right] - \nabla_\omega\hat{Q}_\omega(h)$. Substituting Eq. (27) into Eq. (26), we obtain:

$$\begin{aligned}\nabla_\omega\varepsilon_\omega &= \mathbb{E}\left[\beta_\omega(h)\nabla_\omega\beta_\omega(h)\right],\\ &= \gamma\mathbb{E}\left[\beta_\omega(h)\mathbb{E}_\omega\left[(\nabla_\omega\log\pi_\omega(a'|s'))\hat{Q}_\omega(h')\right]\right] + \mathbb{E}\left[\beta_\omega(h)\Gamma_\omega(h)\right]\end{aligned} \tag{28}$$

To find an analytic expression for the first term of Eq. (28), we rely on the following theorem:

Theorem 4 (Analytic Expression for Derivative of Boltzmann Policy Under Expectation)**.** *If $\pi_\omega(a|s)$ is the Boltzmann policy defined in Eq.* (3)*, it follows that:*

$$\mathbb{E}\left[\beta_\omega(h)\mathbb{E}_\omega\left[(\nabla_\omega\log\pi_\omega(a'|s'))\hat{Q}_\omega(h')\right]\right] = \frac{\varepsilon_\omega\mathbb{E}\left[\beta_\omega(h)\Gamma_\omega(h)\right]\mathcal{E}_\omega\hat{Q}_\omega(h) + \mathcal{E}_\omega\left[\nabla_\omega\hat{Q}_\omega(h)\right]}{(\varepsilon_\omega)^2\left(1 + \gamma\mathbb{E}\left[\beta_\omega(h)\mathbb{E}_\omega\left[\hat{Q}_\omega(h')\right]\right]\right)},$$

where $\mathcal{E}_\omega$ is the operator $\mathcal{E}_\omega\cdot \coloneqq \mathbb{E}\left[\beta_\omega(h)\mathbb{E}_\omega\left[\hat{Q}_\omega(h')\mathcal{M}_\omega\cdot\right]\right]$ and $\mathcal{M}_\omega$ denotes the operator $\mathcal{M}_\omega[\cdot] \coloneqq \cdot - \mathbb{E}_{a\sim\pi_\omega(a|s)}\left[\cdot\right]$

Proof. consider the derivative $\pi_\omega(a|s)\nabla_\omega \log \pi_\omega(a|s)$:

$$\begin{aligned}
\pi_\omega(a|s)\nabla_\omega \log \pi_\omega(a|s) &= \nabla_\omega \pi_\omega(a|s), \\
&= \nabla_\omega \frac{\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right)}{\int_{\mathcal{A}} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) da}, \\
&= \nabla_\omega \left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) \frac{\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right)}{\int_{\mathcal{A}} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) da} \\
&\quad - \frac{\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right)}{\int_{\mathcal{A}} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) da} \cdot \frac{\int_{\mathcal{A}} \nabla_\omega \left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) da}{\int_{\mathcal{A}} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) da}, \\
&= \nabla_\omega \left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) \pi_\omega(a|s) - \pi_\omega(a|s) \int_{\mathcal{A}} \nabla_\omega \left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) \pi_\omega(a|s) da, \\
&= \nabla_\omega \left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) \pi_\omega(a|s) - \pi_\omega(a|s) \mathbb{E}_{a\sim\pi_\omega(a|s)}\left[\nabla_\omega \left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right)\right], \\
&= \pi_\omega(a|s) \left(\nabla_\omega \left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) - \mathbb{E}_{a\sim\pi_\omega(a|s)}\left[\nabla_\omega \left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right)\right]\right). \qquad (29)
\end{aligned}$$

Finding an expression for $\nabla_\omega \left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right)$, we have:

$$\nabla_\omega \left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right) = \frac{1}{(\varepsilon_\omega)^2}\left(\varepsilon_\omega \nabla_\omega \hat{Q}_\omega(h) - \hat{Q}_\omega(h)\nabla_\omega \varepsilon_\omega\right).$$

Substituting into Eq. (29), we obtain:

$$\begin{aligned}
\pi_\omega(a|s)\nabla_\omega \log \pi_\omega(a|s) &= \frac{\pi_\omega(a|s)}{(\varepsilon_\omega)^2}\Bigg(\varepsilon_\omega \nabla_\omega \hat{Q}_\omega(h) - \hat{Q}_\omega(h)\nabla_\omega \varepsilon_\omega \\
&\quad - \mathbb{E}_{a\sim\pi_\omega(a|s)}\left[\varepsilon_\omega \nabla_\omega \hat{Q}_\omega(h) - \hat{Q}_\omega(h)\nabla_\omega \varepsilon_\omega\right]\Bigg), \\
&= \frac{\pi_\omega(a|s)}{(\varepsilon_\omega)^2}\Bigg(\varepsilon_\omega \left(\nabla_\omega \hat{Q}_\omega(h) - \mathbb{E}_{a\sim\pi_\omega(a|s)}\left[\nabla_\omega \hat{Q}_\omega(h)\right]\right) \\
&\quad + \nabla_\omega \varepsilon_\omega \left(\mathbb{E}_{a\sim\pi_\omega(a|s)}\left[\hat{Q}_\omega(h)\right] - \hat{Q}_\omega(h)\right)\Bigg), \\
&= \frac{\pi_\omega(a|s)}{(\varepsilon_\omega)^2}\left(\varepsilon_\omega \mathcal{M}_\omega\left[\nabla_\omega \hat{Q}_\omega(h)\right] - \nabla_\omega \varepsilon_\omega \mathcal{M}_\omega \hat{Q}_\omega(h)\right),
\end{aligned}$$

where $\mathcal{M}_\omega$ denotes the operator $\mathcal{M}_\omega[\cdot] \coloneqq \cdot - \mathbb{E}_{a\sim\pi_\omega(a|s)}[\cdot]$. Dividing both sides by $\pi_\omega(a|s)$ yields:

$$\nabla_\omega \log \pi_\omega(a|s) = \frac{1}{(\varepsilon_\omega)^2}\left(\varepsilon_\omega \mathcal{M}_\omega\left[\nabla_\omega \hat{Q}_\omega(h)\right] - \nabla_\omega \varepsilon_\omega \mathcal{M}_\omega \hat{Q}_\omega(h)\right).$$

Now, substituting for $\nabla_\omega \varepsilon_\omega = \mathbb{E}\left[\beta_\omega(h)\nabla_\omega \beta_\omega(h)\right]$ from Eq. (26) yields:

$$\nabla_\omega \log \pi_\omega(a|s) = \frac{1}{(\varepsilon_\omega)^2}\left(\varepsilon_\omega \mathcal{M}_\omega\left[\nabla_\omega \hat{Q}_\omega(h)\right] - \mathbb{E}\left[\beta_\omega(h)\nabla_\omega \beta_\omega(h)\right] \mathcal{M}_\omega \hat{Q}_\omega(h)\right).$$

Now substituting for $\nabla_\omega \beta_\omega(h) = \gamma \mathbb{E}_\omega \left[(\nabla_\omega \log \pi_\omega(a'|s')) \hat{Q}_\omega(h') \right] + \Gamma_\omega(h)$ from Eq. (27), and re-arranging for $\nabla_\omega \log \pi_\omega(a|s)$:

$$\nabla_\omega \log \pi_\omega(a|s) = \frac{1}{(\varepsilon_\omega)^2} \Bigg(\varepsilon_\omega \mathcal{M}_\omega \left[\nabla_\omega \hat{Q}_\omega(h) \right] - \gamma \mathbb{E} \left[\beta_\omega(h) \mathbb{E}_\omega \left[(\nabla_\omega \log \pi_\omega(a'|s')) \hat{Q}_\omega(h') \right] \right] \\ + \mathbb{E} \left[\beta_\omega(h) \Gamma_\omega(h) \right] \mathcal{M}_\omega \hat{Q}_\omega(h) \Bigg),$$

$$\nabla_\omega \log \pi_\omega(a|s) + \gamma \mathbb{E} \left[\beta_\omega(h) \mathbb{E}_\omega \left[(\nabla_\omega \log \pi_\omega(a'|s')) \hat{Q}_\omega(h') \right] \right] = \frac{1}{(\varepsilon_\omega)^2} \Bigg(\varepsilon_\omega \mathcal{M}_\omega \left[\nabla_\omega \hat{Q}_\omega(h) \right] \\ + \mathbb{E} \left[\beta_\omega(h) \Gamma_\omega(h) \right] \mathcal{M}_\omega \hat{Q}_\omega(h) \Bigg).$$

Now, to obtain our desired result, we first multiply both sides by $\hat{Q}_\omega(h)$, take the expectation $\mathbb{E}_\omega$, multiply by $\beta_\omega(h)$ and finally take the expectation $\mathbb{E}$:

$$\mathbb{E} \left[\beta_\omega(h) \mathbb{E}_\omega \left[(\nabla_\omega \log \pi_\omega(a'|s')) \hat{Q}_\omega(h') \right] \right] \left(1 + \gamma \mathbb{E} \left[\beta_\omega(h) \mathbb{E}_\omega \left[\hat{Q}_\omega(h') \right] \right] \right) \\ = \frac{1}{(\varepsilon_\omega)^2} \Bigg(\varepsilon_\omega \mathbb{E} \left[\beta_\omega(h) \mathbb{E}_\omega \left[\hat{Q}_\omega(h') \mathcal{M}_\omega \left[\nabla_\omega \hat{Q}_\omega(h') \right] \right] \right] \\ + \mathbb{E} \left[\beta_\omega(h) \Gamma_\omega(h) \right] \mathbb{E} \left[\beta_\omega(h) \mathbb{E}_\omega \left[\hat{Q}_\omega(h') \mathcal{M}_\omega \hat{Q}_\omega(h') \right] \right] \Bigg).$$

$$\mathbb{E} \left[\beta_\omega(h) \mathbb{E}_\omega \left[(\nabla_\omega \log \pi_\omega(a'|s')) \hat{Q}_\omega(h') \right] \right] = \frac{\mathbb{E} \left[\beta_\omega(h) \mathbb{E}_\omega \left[\hat{Q}_\omega(h') \mathcal{M}_\omega \left[\nabla_\omega \hat{Q}_\omega(h') \right] \right] \right]}{\varepsilon_\omega \left(1 + \gamma \mathbb{E} \left[\beta_\omega(h) \mathbb{E}_\omega \left[\hat{Q}_\omega(h') \right] \right] \right)} \\ + \frac{\mathbb{E} \left[\beta_\omega(h) \Gamma_\omega(h) \right] \mathbb{E} \left[\beta_\omega(h) \mathbb{E}_\omega \left[\hat{Q}_\omega(h') \mathcal{M}_\omega \hat{Q}_\omega(h') \right] \right]}{(\varepsilon_\omega)^2 \left(1 + \gamma \mathbb{E} \left[\beta_\omega(h) \mathbb{E}_\omega \left[\hat{Q}_\omega(h') \right] \right] \right)}, \\ = \frac{\varepsilon_\omega \mathbb{E} \left[\beta_\omega(h) \Gamma_\omega(h) \right] \mathcal{E}_\omega \hat{Q}_\omega(h) + \mathcal{E}_\omega \left[\nabla_\omega \hat{Q}_\omega(h) \right]}{(\varepsilon_\omega)^2 \left(1 + \gamma \mathbb{E} \left[\beta_\omega(h) \mathbb{E}_\omega \left[\hat{Q}_\omega(h') \right] \right] \right)},$$

as required. □

Using Theorem 4 to substitute for $\mathbb{E} \left[\beta_\omega(h) \mathbb{E}_\omega \left[(\nabla_\omega \log \pi_\omega(a'|s')) \hat{Q}_\omega(h') \right] \right]$ into Eq. (27), we obtain the result:

$$\nabla \varepsilon_\omega = \frac{\varepsilon_\omega \mathbb{E} \left[\beta_\omega(h) \Gamma_\omega(h) \right] \mathcal{E}_\omega \hat{Q}_\omega(h) + \mathcal{E}_\omega \left[\nabla_\omega \hat{Q}_\omega(h) \right]}{(\varepsilon_\omega)^2 \left(1 + \gamma \mathbb{E} \left[\beta_\omega(h) \mathbb{E}_\omega \left[\hat{Q}_\omega(h') \right] \right] \right)} + \mathbb{E} \left[\beta_\omega(h) \Gamma_\omega(h) \right]. \tag{30}$$

The second term of Eq. (30) is the standard policy evaluation gradient and the first term changes $\pi_\omega(a|s)$ in the direction of increasing ε_ω. We see that all expectations in Eq. (30) can be approximated by sampling from our variational policy $\pi_\theta(a|s) \approx \pi_\omega(a|s)$. After a complete E-step, and under Assumption 4, we have $\pi_\theta(a|s) = \pi_\omega(a|s)$ and the gradient is exact.

While the first term in Eq. (30) is certainly tractable, it presents a formidable challenge for the programmer to implement, especially if unbiased estimates are required; several expressions which involve the multiplication of more than one expectation $\mathbb{E}_\omega$ need to be evaluated. In all of these cases, expectations approximated using the same data will introduce bias, however it is infeasible to sample more than once from the same state in the environment. Like in Sutton et al. [58], a solution to this problem is to learn a function approximator for one of the expectations that is updated at a slower rate than the other expectation. Alternatively, these function approximators can be updated using separate data batches from a replay buffer.

A radical approach is simply to neglect this gradient term, which we discuss in Appendix F.3. A more considered approach is to use an operator that does not constraint Ω. Consider the operator introduced in Appendix F.2,

$$\mathcal{T}_{\omega,k}\cdot = r(h) + \gamma \mathbb{E}_{\omega,k}\left[\cdot\right],$$

where we have used the shorthand for expectation $\mathbb{E}_{\omega,k}\left[\cdot\right] \coloneqq \mathbb{E}_{h'\sim p(s'|h)p_{\omega,k}(a'|s')}\left[\cdot\right]$ and the Boltzmann distribution is defined as

$$p_{\omega,k}(a|s) \coloneqq \frac{\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_k}\right)}{\int_{\mathcal{A}} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_k}\right) da}.$$

The incremental residual error is defined as $\varepsilon_{\omega,k} \coloneqq \frac{1}{2|\mathcal{H}|}\|\beta_{\omega,k}(h)\|_2^2 + \varepsilon_k$ and $\beta_{\omega,k}(h) \coloneqq \mathcal{T}_{\omega,k}\hat{Q}_\omega(h) - \hat{Q}_\omega(h)$. Taking gradients of $\varepsilon_{\omega,k}$ directly yields:

$$\nabla_\omega \varepsilon_{\omega,k} = \mathbb{E}\left[\beta_{\omega,k}(h)\nabla_\omega \beta_{\omega,k}(h)\right].$$

where

$$\begin{aligned}\nabla_\omega \beta_{\omega,k}(h) &= \nabla_\omega \mathbb{E}_{\omega,k}\left[\hat{Q}_\omega(h')\right] - \nabla_\omega \hat{Q}_\omega(h),\\ &= \nabla_\omega \mathbb{E}_{\omega,k}\left[\hat{Q}_\omega(h')\right] - \nabla_\omega \hat{Q}_\omega(h),\\ &= \mathbb{E}_{\omega,k}\left[\nabla_\omega \log p_{\omega,k}(a'|s') + \nabla_\omega \hat{Q}_\omega(h')\right] - \nabla_\omega \hat{Q}_\omega(h). \qquad (31)\end{aligned}$$

Now, $\nabla_\omega \log p_{\omega,k}(a'|s')$ can be computed directly as:

$$\begin{aligned}\nabla_\omega \log p_{\omega,k}(a'|s') &= \nabla_\omega \left(\frac{\hat{Q}_\omega(h')}{\varepsilon_k} - \log \int_{\mathcal{A}} \exp\left(\frac{\hat{Q}_\omega(h')}{\varepsilon_k}\right) da\right),\\ &= \frac{\nabla_\omega \hat{Q}_\omega(h')}{\varepsilon_k} - \int_{\mathcal{A}} \frac{\nabla_\omega \hat{Q}_\omega(h')}{\varepsilon_k} \frac{\exp\left(\frac{\hat{Q}_\omega(h')}{\varepsilon_k}\right)}{\int_{\mathcal{A}} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_k}\right) da} da,\\ &= \frac{\nabla_\omega \hat{Q}_\omega(h')}{\varepsilon_k} - \int_{\mathcal{A}} \frac{\nabla_\omega \hat{Q}_\omega(h)}{\varepsilon_k} p_{\omega,k}(a'|s') da,\\ &= \frac{\nabla_\omega \hat{Q}_\omega(h')}{\varepsilon_k} - \mathbb{E}_{a'\sim p_{\omega,k}(a'|s')}\left[\frac{\nabla_\omega \hat{Q}_\omega(h)}{\varepsilon_k}\right],\\ &= \mathcal{M}_{\omega,k}\left[\frac{\nabla_\omega \hat{Q}_\omega(h')}{\varepsilon_k}\right],\end{aligned}$$

where where $\mathcal{M}_{\omega,k}$ denotes the operator $\mathcal{M}_{\omega,k}[\cdot] \coloneqq \cdot - \mathbb{E}_{a\sim p_{\omega,k}(a|s)}\left[\cdot\right]$. Substituting into Eq. (31) yields:

$$\nabla_\omega \beta_{\omega,k}(h) = \mathbb{E}_{\omega,k}\left[\mathcal{M}_{\omega,k}\left[\frac{\nabla_\omega \hat{Q}_\omega(h')}{\varepsilon_k}\right] + \nabla_\omega \hat{Q}_\omega(h')\right] - \nabla_\omega \hat{Q}_\omega(h).$$

E.4 Discussion of E-step

We now explore the relationship between classical actor-critic methods and the E-step. The policy gradient theorem [56] derives an update for the derivative of the RL objective (1) with respect to the policy parameters

$$\nabla_\theta J(\theta) = \mathbb{E}_{s\sim\rho^\pi(s)}\left[\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[Q^\pi(h)\nabla_\theta \log \pi_\theta(a|s)\right]\right],$$

where $\rho^\pi(s)$ is the discounted-ergodic occupancy, defined formally in Ciosek & Whiteson [11], and in general not a normalised distribution. To obtain practical algorithms, we collect rollouts and treat them as samples from the steady-state distribution instead.

By contrast, the VIREL policy update in Eq. (25) involves an expectation over $d(s)$, which can be any sampling distribution decorrelated from π ensuring all states are visited infinitely often. As $\hat{Q}_\omega(h)$

is also independent of $\pi_\theta(a|s)$, we can move the gradient operator ∇_θ out of the inner integral to obtain

$$\mathbb{E}_{s\sim d(s)}\left[\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\hat{Q}_\omega(h)\nabla_\theta\log\pi_\theta(a|s)\right]\right]=\mathbb{E}_{s\sim d(s)}\left[\nabla_\theta\mathbb{E}_{a\sim\pi_\theta(a|s)}\left[\hat{Q}_\omega(h)\right]\right]$$

This transformation is essential in deriving powerful policy gradient methods such as Expected and Fourier Policy Gradients [10, 15] and holds for deterministic polices [53]. However, unlike in VIREL, it is not strictly justified in the classic policy gradient theorem [56] and MERL formulation [25].

F Relaxations and Approximations

F.1 Relaxation of Representability of Q-functions

In our analysis, Assumption 2 is required by Theorem 2 to ensure that a maximum to the optimisation problem exists, however it can be completely neglected provided that projected Bellman operators are used; moreover, if projected Bellman operators are used, our M-step is also always guaranteed to converge, even if our E-step does not. Consequently, we can terminate the algorithm by carrying out a complete M-step at any time using our variational approximation and still be guaranteed convergence to a sub-optimal point.

We now introduce the assumption that our action-value function approximator is three-times differentiable over Ω, which is required for convergence guarantees.

Assumption 5 (Universal Smoothness of $\hat{Q}_\omega(h)$)**.** *We require that $\hat{Q}_\omega(h)\in\mathbb{C}^3(\Omega)$ for all $h\in\mathcal{H}$,*

We now extend the analysis of Bhatnagar et al. [6] to continuous domains. Consider the local linearisation of the function approximator $\hat{Q}_\omega(h)\approx b_\omega^\top(h)\omega$, where $b_\omega(h):=\nabla_\omega\hat{Q}_\omega(h)$. We define the projection operator $\mathcal{P}_\omega Q(\cdot):=b_\omega^\top(h)\omega'$ where $\tilde{\omega}$ are the parameters that minimise the difference between the action-value function and the local linearisation:

$$\tilde{\omega}:=\arg\min_{\omega'}\frac{1}{2|\mathcal{H}|}\|Q(h)-b_\omega^\top(h)\omega'\|_2^2. \tag{32}$$

Using the notation $\mathbb{E}[\cdot]\triangleq\mathbb{E}_{h\sim\mathcal{U}(h)}[\cdot]$ and taking derivatives of Eq. (32) with respect to ω' yields:

$$\begin{aligned}\nabla_{\omega'}\frac{1}{2|\mathcal{H}|}\|Q(h)-\omega'^\top\|_2^2&=\frac{1}{2}\nabla_{\omega'}\mathbb{E}\left[(Q(h)-b_\omega^\top(h)\omega')^2\right],\\&=\frac{1}{2}\mathbb{E}\left[\nabla_{\omega'}(Q(h)^2-2b_\omega^\top(h)\omega'Q(h)+b_\omega^\top(h)\omega'b_\omega^\top(h)\omega'\right],\\&=\mathbb{E}\left[b_\omega(h)b_\omega^\top(h)\omega'-b_\omega(h)Q(h)\right].\end{aligned}$$

Equating to zero and solving for $\tilde{\omega}$, we obtain:

$$\tilde{\omega}=\mathbb{E}\left[b_\omega(h)b_\omega^\top(h)\right]^{-1}\mathbb{E}\left[b_\omega(h)Q(h)\right].$$

Substituting into our operator yields:

$$\mathcal{P}_\omega\cdot=b_\omega^\top(h)\mathbb{E}\left[b_\omega(h)b_\omega^\top(h)\right]^{-1}\mathbb{E}\left[b_\omega(h)\cdot\right].$$

We can therefore interpret $\mathcal{P}$ as an operator that projects an action-value function onto the tangent space of $\hat{Q}_\omega(h)$ at ω. For linear function approximators of the form $\hat{Q}_\omega(h)=b^\top(h)\omega$, the projection operator is independent of ω and projects Q directly onto the nearest function approximator and the operator [57].

We now replace the residual error in Section 3.1 with the projected residual error,

$$\varepsilon_\omega:=\frac{1}{2|\mathcal{H}|}\left\|\mathcal{P}_\omega\left(\mathcal{T}_\omega\hat{Q}_\omega(h)-\hat{Q}_\omega(h)\right)\right\|_2^2. \tag{33}$$

By definition, there always exists fixed point $\omega\in\Omega$ for which $\varepsilon_\omega=0$, which means that ε_ω now satisfies all requirements in Theorem 2 without Assumption 2. We can also carry out a complete partial variational M-step by minimising the surrogate ε_ω, keeping $\pi_\omega(a|s)=\pi_\theta(a|s)$ in all expectations. At convergence, we have $\varepsilon_\omega=0$ in this case.

We now derive the more convenient form of ε_ω from Lemma 1 in Bhatnagar et al. [6], extending this result to continuous domains. Let $\beta_\omega(h) := \mathcal{T}_\omega \hat{Q}_\omega(h) - \hat{Q}_\omega(h)$. Substituting into Eq. (33), we obtain:

$$
\begin{aligned}
2\varepsilon_\omega &= \frac{1}{|\mathcal{H}|} \left\| \mathcal{P}_\omega \beta_\omega(h) \right\|_2^2 , \\
&= \frac{1}{|\mathcal{H}|} \left\| b_\omega^\top(h) \mathbb{E}\left[b_\omega(h) b_\omega^\top(h) \right]^{-1} \mathbb{E}\left[b_\omega(h) \beta_\omega(h) \right] \right\|_2^2 , \\
&= \mathbb{E}\left[\mathbb{E}\left[b_\omega^\top(h) \beta_\omega(h) \right] \mathbb{E}\left[b_\omega(h) b_\omega^\top(h) \right]^{-1} b_\omega(h) b_\omega^\top(h) \mathbb{E}\left[b_\omega(h) b_\omega^\top(h) \right]^{-1} \mathbb{E}\left[\beta_\omega(h) b_\omega(h) \right] \right] , \\
&= \mathbb{E}\left[b_\omega^\top(h) \beta_\omega(h) \right] \mathbb{E}\left[b_\omega(h) b_\omega^\top(h) \right]^{-1} \mathbb{E}\left[b_\omega(h) b_\omega^\top(h) \right] \mathbb{E}\left[b_\omega(h) b_\omega^\top(h) \right]^{-1} \mathbb{E}\left[\beta_\omega(h) b_\omega(h) \right] , \\
&= \mathbb{E}\left[b_\omega^\top(h) \beta_\omega(h) \right] \mathbb{E}\left[b_\omega(h) b_\omega^\top(h) \right]^{-1} \mathbb{E}\left[\beta_\omega(h) b_\omega(h) \right] .
\end{aligned}
$$

Denoting $\zeta_\omega := \mathbb{E}\left[b_\omega(h) b_\omega^\top(h) \right]^{-1} \mathbb{E}\left[\beta_\omega(h) b_\omega(h) \right]$ following the analysis in [6], we find the derivative of ε_ω as:

$$
\nabla_\omega \varepsilon_\omega = \mathbb{E}\left[(\nabla_\omega \beta_\omega(h)) b_\omega^\top(h) \zeta_\omega \right] + \mathbb{E}\left[(\beta_\omega(h) - b_\omega^\top(h)\zeta_\omega) \nabla_\omega^2 \hat{Q}_\omega(h) \zeta_\omega \right] .
$$

Following the method of Pearlmutter [45], the multiplication between the Hessian and ζ_ω can be calculated in $O(n)$ time, which bounds the overall complexity of our algorithm. To avoid bias in our estimate, we learn a set of weights $\hat{\zeta} \approx \zeta_\omega$ on a slower timescale, which we update as:

$$
\hat{\zeta}_{k+1} \leftarrow \hat{\zeta}_k + \alpha_{\zeta k} \left(\beta_\omega(h) - b_\omega^\top(h) \zeta_k \right) b_\omega(h), \tag{34}
$$

where $\alpha_{\zeta k}$ is a step size chosen to ensure that $\alpha_{\zeta k} < \alpha_{\text{critic}}$. The weights are then used to find our gradient term:

$$
\nabla_\omega \varepsilon_\omega = \mathbb{E}\left[(\nabla_\omega \beta_\omega(h)) b_\omega^\top(h) \hat{\zeta} \right] + \mathbb{E}\left[(\beta_\omega(h) - b_\omega^\top(h)\hat{\zeta}) \nabla_\omega^2 \hat{Q}_\omega(h) \zeta_\omega \right] .
$$

In our framework, the term $\nabla \beta_\omega(h)$ is specific to our choice of operator. In Bhatnagar et al. [6], a TD-target is used and parameter updates for ω are given as:

$$
\begin{aligned}
\omega_{k+1} &= \mathfrak{P}\left(\omega_k + \alpha_{\omega k}(b_k - \gamma b_k') b_k^\top \hat{\zeta}_k - q_k \right) , \qquad (35) \\
q_k &:= \left(\beta_{\omega_k}(h_k) - b_k^\top \hat{\zeta}_k \right) \nabla_\omega^2 \hat{Q}_{\omega_k}(h_k) \hat{\zeta}_k
\end{aligned}
$$

where $b_k := b_{\omega_k}(h_k)$ and $\mathfrak{P}(\cdot)$ is an operator that projects ω_k into any arbitrary compact set with a smooth boundary, $\mathcal{C}$. The projection $\mathfrak{P}(\cdot)$ is introduced for mathematical formalism and, provided $\mathcal{C}$ is large enough to contain all solutions $\left\{ \omega | \mathbb{E}\left[\beta_\omega(h) \nabla_\omega \hat{Q}_\omega(h) \right] = 0 \right\} \subseteq \mathcal{C}$, has no bearing on the updates in practice. Under Assumption 5, provided the step size conditions $\sum_k^\infty \alpha_{\zeta k} = \sum_k^\infty \alpha_{\omega k} = \infty$, $\sum_k^\infty \alpha_{\zeta k}^2 <, \sum_k^\infty \alpha_{\omega k}^2 < \infty$ and $\lim_{k\to\infty} \frac{\alpha_{\zeta k}}{\alpha_{\omega k}} = 0$ hold and $\mathbb{E}[b_\omega(h) b_\omega^\top(h)]$ is non-singular $\forall \omega \in \Omega$, the analysis in Theorem 2 of Bhatnagar et al. [6] applies and the updates in Eqs. (34) and (35) are guaranteed to converge to the TD fixed point. This demonstrates using data sampled from any variational policy $\pi_\theta(a|s)$ to update ω_k as Eqs. (34) and (35), ω_k will converge to a fixed point.

F.2 Off-Policy Bellman Operators

As discussed in Section 3.1, using the Bellman operator $\mathcal{T}^{\pi_\omega}\cdot$ induces a constraint on the set of parameters Ω. While this constraint can be avoided using the optimal Bellman operator $\mathcal{T}^*\cdot := r(h) + \gamma \mathbb{E}_{h' \sim p(s'|h)} \left[\max_{a'}(\cdot) \right]$, evaluating $\max_{a'}(\hat{Q}_\omega(h'))$ may be difficult in large continuous domains. We now make a slight modification to our model in Section 3.1 to accommodate a Bellman operator that avoids these two practical difficulties.

Firstly, we introduce a new Boltzmann distribution $p_{\omega,k}(a|s)$:

$$
p_{\omega,k}(a|s) := \frac{\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_k} \right)}{\int_\mathcal{A} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_k} \right) da},
$$

where $\{\varepsilon_k\}$ is a sequence of positive constants $\varepsilon_k \geq 0$, $\lim_{k\to\infty} \varepsilon_k = 0$. We now introduce a new operator $\mathcal{T}_{\omega,k}\cdot$, defined as is the Bellman operator for $p_{\omega,k}(a|s)$:

$$\mathcal{T}_{\omega,k}\cdot := \mathcal{T}^{p_{\omega,k}}\cdot = r(h) + \gamma\mathbb{E}_{h'\sim p(s'|h)p_{\omega,k}(a'|s')}\left[\cdot\right]. \tag{36}$$

Let $\pi_{\omega,k}(a|s)$ be the Boltzmann policy:

$$\pi_{\omega,k}(a|s) := \frac{\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_{\omega,k}}\right)}{\int_\mathcal{A} \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_{\omega,k}}\right) da},$$

where the residual error $\varepsilon_{\omega,k} := \frac{c}{p}\|\mathcal{T}_{\omega,k}\hat{Q}_\omega(h) - \hat{Q}_\omega(h)\|_p^p + \varepsilon_k$. It is clear that $\mathcal{T}_{\omega,k}\cdot$ does not constrain Ω as ε_k has no dependency on ω and $\pi_{\omega,k}(a|s)$ is well defined for all $\omega \in \Omega$.

We now formally prove that $\min_\omega \lim_{k\to\infty} \varepsilon_{\omega,k} = \min_\omega \varepsilon_\omega$, and so minimising $\varepsilon_{\omega,k}$ is the same as minimising the objective ε_ω from Section 3.1 and that $\mathcal{T}_{\omega,k}\cdot \in \mathbb{T}$. We also prove that $\min_\omega \lim_{k\to\infty} \varepsilon_{\omega,k} = \lim_{k\to\infty} \min_\omega \varepsilon_{\omega,k}$ (i.e. that $\min$ and $\lim$ commute), which allows us to minimise our objective incrementally over sequences $\varepsilon_{\omega,k}$.

Theorem 5 (Incremental Optimisation of $\varepsilon_{\omega,k}$). *Let $\varepsilon_{\omega,k} := \frac{c}{p}\|\mathcal{T}_{\omega,k}\hat{Q}_\omega(h) - \hat{Q}_\omega(h)\|_p^p + \varepsilon_k$ and $\mathcal{T}_{\omega,k}$ be the Bellman operator defined in Eq.* (36). *It follows that i) $\mathcal{T}_{\omega,k}\cdot \in \mathbb{T}$, ii) $\min_\omega \lim_{k\to\infty} \varepsilon_{\omega,k} = \min_\omega \varepsilon_\omega$ and iii) $\min_\omega \lim_{k\to\infty} \varepsilon_{\omega,k} = \lim_{k\to\infty} \min_\omega \varepsilon_{\omega,k}$*

Proof. To prove i), we take the limit $\lim_{k\to\infty} \mathcal{T}_{\omega,k}\hat{Q}_\omega(h) = \mathcal{T}^*\hat{Q}_\omega(h)$:

$$\lim_{k\to\infty} \mathcal{T}_{\omega,k}\hat{Q}_\omega(h) = r(h) + \lim_{k\to\infty} \gamma\mathbb{E}_{h'\sim p(s'|h)p_{\omega,k}(a'|s')}\left[\hat{Q}_\omega(h)\right].$$

Observe that from Theorem 1, we have

$$\lim_{\varepsilon_k\to\infty} \gamma\mathbb{E}_{h'\sim p(s'|h)p_{\omega,k}(a'|s')}\left[\hat{Q}_\omega(h)\right] = \gamma\mathbb{E}_{h'\sim p(s'|h)\delta(a=\arg\max_{a'}(\hat{Q}_\omega(a',s))}\left[\hat{Q}_\omega(h)\right],$$

hence:

$$\begin{aligned}\lim_{k\to\infty} \mathcal{T}_{\omega,k}\hat{Q}_\omega(h) &= r(h) + \lim_{k\to\infty} \gamma\mathbb{E}_{h'\sim p(s'|h)p_{\omega,k}(a'|s')}\left[\hat{Q}_\omega(h)\right],\\ &= r(h) + \gamma\mathbb{E}_{h'\sim p(s'|h)\delta(a=\arg\max_{a'}(\hat{Q}_\omega(a',s))}\left[\hat{Q}_\omega(h)\right],\\ &= r(h) + \gamma\mathbb{E}_{s'\sim p(s'|h)}\left[\max_{a'}(\hat{Q}_\omega(h))\right],\\ &= \mathcal{T}^*\hat{Q}_\omega(h).\end{aligned}$$

Our operator is therefore constructed such that in the limit $k \to \infty$, we recover the optimal Bellman operator. Observe too that as $\frac{c}{p}\|\mathcal{T}_{\omega,k}\hat{Q}_\omega(h) - \hat{Q}_\omega(h)\|_p^p \geq 0$, we have $\varepsilon_{\omega,k} > 0$ for all $\varepsilon_k > 0$. From Theorem 1, we have $\pi_{\omega,k}(a|s) \to \delta(a = \arg\max_{a'}(\hat{Q}_\omega(a',s))$ when $\varepsilon_{\omega,k} = 0$, which therefore can only occur when $\lim_{k\to\infty} \varepsilon_k = 0$. Under this limit, we have $\lim_{k\to\infty} \mathcal{T}_{\omega,k} = \mathcal{T}^*$ and so $\mathcal{T}_{\omega,k} \in \mathbb{T}$, as required for i).

To prove ii), consider taking the limit of $\varepsilon_{\omega,k}$ directly:

$$\begin{aligned}\lim_{k\to\infty} \varepsilon_{\omega,k} &= \lim_{k\to\infty}\left(\frac{c}{p}\|\mathcal{T}_{\omega,k}\hat{Q}_\omega(h) - \hat{Q}_\omega(h)\|_p^p + \varepsilon_k\right),\\ &= \lim_{k\to\infty}\left(\frac{c}{p}\|\mathcal{T}_{\omega,k}\hat{Q}_\omega(h) - \hat{Q}_\omega(h)\|_p^p\right) + \varepsilon_\infty,\\ &= \frac{c}{p}\|\lim_{k\to\infty} \mathcal{T}_{\omega,k}\hat{Q}_\omega(h) - \hat{Q}_\omega(h)\|_p^p,\\ &= \frac{c}{p}\|\mathcal{T}^*\hat{Q}_\omega(h) - \hat{Q}_\omega(h)\|_p^p,\\ &= \varepsilon_\omega,\end{aligned} \tag{37}$$

as required.

To prove iii), let $\tilde{\omega}_k$ be the minimiser of $\varepsilon_{\omega,k}$, that is $\tilde{\omega}_k = \arg\min_\omega \varepsilon_{\omega,k}$. Let $\tilde{\omega}$ be the limit of all such sequences $\tilde{\omega} = \lim_{k\to\infty} \tilde{\omega}_k$ and let $\omega^* = \arg\min_\omega \varepsilon_\omega$. By definition, we have $\varepsilon_{\tilde{\omega}_k,k} \le \varepsilon_{\omega,k}$. Taking the limit $k \to \infty$ and then the min, we have:

$$\begin{aligned}\min \lim_{k\to\infty} \varepsilon_{\tilde{\omega}_k,k} &\le \min \lim_{k\to\infty} \varepsilon_{\omega,k}, \\ \implies \varepsilon_{\tilde{\omega},\infty} &\le \min \lim_{k\to\infty} \varepsilon_{\omega,k}. \end{aligned} \tag{38}$$

Using Assumption 2 and Eq. (37), it follows that the right hand side of Eq. (38) is $\min \lim_{k\to\infty} \varepsilon_{\omega,k} = \min \varepsilon_\omega = 0$, hence $\varepsilon_{\tilde{\omega},\infty} \le 0$. By definition, $\varepsilon_{\tilde{\omega},\infty} \ge 0$, and so equality must hold. It therefore follows $\lim_{k\to\infty} \min_\omega \varepsilon_{\omega,k} = \varepsilon_{\tilde{\omega},\infty} = 0$, which implies $\min_\omega \lim_{k\to\infty} \varepsilon_{\omega,k} = \lim_{k\to\infty} \min_\omega \varepsilon_\omega = 0$ as required. □

Overall, this result permits us to carry out separate optimisations over $\varepsilon_{\omega,k}$ while gradually increasing $k \to \infty$ to obtain the same result as minimising ε_ω directly. The advantage to this method is that each minimisation $\varepsilon_{\omega,k}$ involves the operator $\mathcal{T}_{\omega,k}$, which is tractable, mathematically convenient and does not constrain Ω. Note too that, as calculated in Appendix E.3, the gradient $\nabla_\omega \varepsilon_{\omega,k}$ is straightforward to implement in comparison with $\nabla_\omega \varepsilon_\omega$ using $\mathcal{T}^{\pi_\omega}$. We save investigating this operator further for future work.

F.3 Approximate Gradient Methods and Partial Optimisation

A common trick in policy evaluation is to use a direct method [2, 55]. Like in supervised methods [7], direct methods treat the term $\mathcal{T}_\omega \hat{Q}_\omega(h)$ as a fixed target, rather than a differential function. Introducing the notation $\mathbb{E}[\cdot] \triangleq \mathbb{E}_{h\sim\mathcal{U}(h)}[\cdot]$, the gradient can easily be derived as:

$$\begin{aligned}\nabla_\omega \varepsilon_\omega &= \frac{1}{2}\nabla_\omega \mathbb{E}\left[\left(\dashv\left[\mathcal{T}_\omega \hat{Q}_\omega(h)\right] - \hat{Q}_\omega(h)\right)^2\right], \\ &= -\mathbb{E}\left[\left(\hat{Q}_\omega(h) - \mathcal{T}_\omega \hat{Q}_\omega(h)\right)\nabla_\omega \hat{Q}_\omega(h)\right]\end{aligned}$$

where $\dashv[\cdot]$ is the stopgrad operator, which sets the gradient of its operand to zero, $\dashv[\cdot] = \cdot$, $\nabla\dashv[\cdot] = 0$ [16]. The direct method has no convergence guarantees, and indeed there exist several famous examples of divergence when used with classic RL targets [5, 65, 70], however its ubiquity in the RL community is testament to its ease of implementation and empirical success [42, 55]. We therefore see no reason why it should not be successful for VIREL, a claim which we verify in Section 5. In our setting, we replace our M-step with the simplified objective $\omega_{k+1} \leftarrow \arg\min_\omega \varepsilon_\omega$. This is justified because $\arg\min_\omega \varepsilon_\omega$ was the original objective motivated in Section 3.1 assuming we have access to as good enough variational policy $\pi_\omega(a|s) \approx \pi_\theta(a|s)$. More formally, our objective $\mathcal{L}(\omega,\theta)$ is maximised for any $\varepsilon_\omega \to 0$, so $\arg\min_\omega \varepsilon_\omega$ can be considered a surrogate objective for $\mathcal{L}(\omega,\theta)$. Using direct methods, M-step update becomes:

M-Step (Critic) direct: $\omega_{i+1} \leftarrow \omega_i - \alpha_{\text{critic}} \nabla_\omega \varepsilon_\omega|_{\omega=\omega_i}$,

$$\nabla_\omega \varepsilon_\omega = \mathbb{E}\left[\left(\hat{Q}_\omega(h) - \mathcal{T}_\omega \hat{Q}_\omega(h)\right)\nabla_\omega \hat{Q}_\omega(h)\right].$$

We can approximate $\mathcal{T}_\omega \hat{Q}_\omega(h)$ by sampling from the variational distribution $\pi_\theta(a|s)$ and by using any appropriate RL target. Another important approximation that we make is that we perform only partial E- and M-steps, halting optimisation before convergence. From a practical perspective, convergence can often only occur in a limit of infinite time steps anyway, and if good empirical performance result from taking partial E- and M-steps, computation may be wasted carrying out many sub-optimisation steps for little gain.

As analysed by Gunawardana & Byrne [23], such algorithms fall under the umbrella of the generalised alternating maximisation (GAM) framework, and convergence guarantees are specific to the form of function approximator and MDP. Like in many inference settings, we anticipate that most function approximators and MDPs of interest will not satisfy the conditions required to prove convergence, however variational EM procedures are known to be to empirically successful even when convergence properties are not guaranteed [23, 66]. We demonstrate in Section 5 that taking partial EM steps does not hinder our performance.

F.4 Local Smoothness of $\hat{Q}_{\omega^*}(\cdot)$

For Theorem 1 to hold, we require that $\hat{Q}_{\omega^*}(\cdot)$ is locally smooth about its maximum. Our choice of function approximator may prevent this condition from holding, for example, a neural network with ReLU elements can introduce a discontinuity in gradient at $\max_h \hat{Q}_{\omega^*}(h)$. In practice, a formal Dirac-delta function can only ever emerge in the limit of convergence $\varepsilon_\omega \to 0$. In finite time, we obtain, at best, a nascent delta function; that is a function with very small variance that is 'on the way to convergence' (see, for example, Kelly [31] for a formal definition). The mode of a nascent delta function therefore approximates the true Dirac-delta distribution. When $\hat{Q}_{\omega^*}(\cdot)$ is not locally smooth, functions that behave similarly to nascent delta functions will still emerge at finite time, the mode of which we anticipate provides an approximation to the hardmax behaviour we require for most RL settings.

We also require that $\hat{Q}_{\omega^*}(\cdot)$ has a single, unique global maximum for any state. In reality, optimal Q-functions may have more than one global maxima for a single state corresponding to the existence of multiple optimal policies. To ensure Assumption 3 strictly holds, we can arbitrarily reduce the reward for all but one optimal policy. We anticipate that this is unnecessary in practice, as our risk-neutral objective means that a variational policy will be encouraged fit to a single mode anyway. In addition, these assumptions are required to characterise behaviour under convergence to a solution and will not present a problem in finite time where $\hat{Q}_\omega(h)$ is very unlikely to have more than one global optimum.

F.5 Analysis of Approximate EM Algorithms

We now provide two separate analyses of our EM algorithm, replacing the Bellman operator $\mathcal{T}^{\pi_\omega}\cdot$ with its variational approximation $\mathcal{T}^{\pi_\theta}\cdot$ (effectively substituting for $\pi_\omega(\cdot|s) \approx \pi_\theta(\cdot|s)$ under expectation), thereby avoiding any constraints on Ω. In our first analysis, we make no simplifying assumptions on ε_ω, showing that our EM algorithm reduces exactly to policy iteration. In our second analysis, we use a direct method, treating $\mathcal{T}^{\pi_\theta}\cdot$ as a fixed target as outlined in Appendix F.3, showing that the algorithm reduces exactly to Q-learning.

In both analyses, we assume a complete E- and M- step can be carried out and our class of function approximators is rich enough to represent any action-value function. Let $\pi_{\theta_0}(a|\cdot)$ be any initial policy and $\hat{Q}_{\omega_0}(\cdot)$ an arbitrary initialisation of the function approximator. For notational convenience we write $\pi_k(a|\cdot) := \pi_{\theta_k}(a|\cdot)$.

Analysis with ω-Dependent Target As we prove in Theorem 2, we can always maximise our objective with respect to ω by finding ω^* s.t. $\varepsilon_{\omega^*} = 0$. This gives the M-step update:

$$\begin{aligned} \omega_1 &= \arg\min_\omega \varepsilon_\omega, \\ \implies \varepsilon_{\omega_1} &= 0, \\ \implies \mathcal{T}^{\pi_0}\hat{Q}_{\omega_1} &= \hat{Q}_{\omega_1}, \\ \implies \hat{Q}_{\omega_1} &= Q^{\pi_0}(\cdot). \end{aligned}$$

Our E-step amounts to calculating the Boltzmann distribution with $\varepsilon_{\omega_1} = 0$, which from Theorem 1 takes the form of a Dirac-delta distribution:

$$\pi_1(a|\cdot) = \delta\left(a = \arg\max_{a'} Q^{\pi_0}(a', \cdot)\right).$$

We can generalise to the kth EM update as:

$$\hat{Q}_{\omega_k}(\cdot) \leftarrow Q^{\pi_{k-1}}(\cdot), \tag{39}$$

$$\pi_k(a|\cdot) \leftarrow \delta\left(a = \arg\max_{a'} Q^{\pi_{k-1}}(a', \cdot)\right). \tag{40}$$

Together Eqs 39 and 40 are exactly the updates for policy iteration, an algorithm which is known to converge to the optimal policy [59, 55]. We therefore see that, even ignoring the constraint on Ω, the optimal solution is still an attractive fixed point when our algorithms are carried out exactly. Using partial E- and M-steps give a variational approximation to the complete EM algorithm. We now provide a similar analysis using the fixed target of direct methods introduced in Appendix F.3.

Analysis with Fixed Target Using a direct method, we replace the residual error with the fixed target residual error $\varepsilon_\omega \approx \varepsilon_{\omega,k} := \frac{c}{p}\|\mathcal{T}^{\pi_k}\hat{Q}_{\omega_k} - \hat{Q}_\omega\|_p^p$, giving the M-step update:

$$\omega_1 = \arg\min_\omega \varepsilon_{\omega,0}$$

which, under our assumption of representability, is achieved for

$$\begin{aligned} &\varepsilon_{\omega_1,0} = 0, \\ \implies &\hat{Q}_{\omega_1}(\cdot) = \mathcal{T}^{\pi_0}\hat{Q}_{\omega_0}. \end{aligned}$$

As with our ω-dependent target, the E-step amounts to calculating the Boltzmann distribution with $\varepsilon_{\omega_1,0} = 0$, which from Theorem 1 takes the form of a Dirac-delta distribution:

$$\pi_1(a|\cdot) = \delta\left(a = \arg\max_{a'} \hat{Q}_{\omega_1}(a', \cdot)\right).$$

We see that for any policy and function approximator, carrying a complete E- and M- step results in a deterministic policy being learnt in this approximate regime. We generalise to the kth EM updates for $k > 2$ as:

$$\begin{aligned} \pi_{k-1}(a|\cdot) &= \delta\left(a = \arg\max_{a'} \hat{Q}_{\omega_{k-1}}(a', \cdot)\right), \\ \omega_k = \arg\min_\omega \varepsilon_{\omega,k-1} &= \arg\min_\omega \frac{c}{p}\|\mathcal{T}^{\pi_{k-1}}\hat{Q}_{\omega_{k-1}} - \hat{Q}_\omega\|_p^p, \\ &\implies \varepsilon_{\omega_1,0} = 0, \\ &\implies \hat{Q}_{\omega_k}(\cdot) = \mathcal{T}^{\pi_{k-1}}\hat{Q}_{\omega_{k-1}}(\cdot), \\ &= r(\cdot) + \mathbb{E}_{s'|\cdot}\left[\max_{a'} \hat{Q}_{\omega_{k-1}}(s', a')\right], \\ &= \mathcal{T}^*\hat{Q}_{\omega_{k-1}}(\cdot). \end{aligned} \tag{41}$$

From Eq. (41), we see that the EM algorithm carries out Q-learning updates on our function approximator $\hat{Q}_{\omega_k}(\cdot) \leftarrow \mathcal{T}^*\hat{Q}_{\omega_{k-1}}(\cdot)$ for $k > 2$ [68]. See Yang et al. [72] for a theoretical exposition of Q-learning using function approximators. When partial EM steps are carried out, we can view this algorithm as a variational approximation to Q-learning.

G Recovering MPO

We now derive the MPO objective from our framework. Under the probabilistic interpretation in Appendix B, the objective can be derived using the prior $p_\phi(h) = \mathcal{U}(s)\pi_\phi(a|s)$ instead of the uniform distribution. Following the same analysis as in Appendix B, this yields an action-posterior:

$$p_{\omega,\phi}(a|s, \mathcal{O}) = \frac{\exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right)\pi_\phi(a|s)}{\int \exp\left(\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right)\pi_\phi(a|s)da}.$$

Again, following the same analysis as in Appendix B, our ELBO objective is:

$$\mathcal{L}(\omega, \theta, \phi) = \mathbb{E}_{d(s)}\left[\mathbb{E}_{\pi_\theta(a|s)}\left[\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right] - \mathrm{KL}(\pi_\theta(a|s) \parallel \pi_\phi(a|s))\right]. \tag{42}$$

Including a hyper-prior $p(\phi)$ over ϕ adds an additional term to $\mathcal{L}(\omega, \theta, \phi)$:

$$\mathcal{L}(\omega, \theta, \phi) = \mathbb{E}_{d(s)}\left[\mathbb{E}_{\pi_\theta(a|s)}\left[\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega}\right] - \mathrm{KL}(\pi_\theta(a|s) \parallel \pi_\phi(a|s))\right] + \log p(\phi).$$

which is exactly the MPO objective, with an adaptive scaling constant ε_ω to balance the influence of $\mathrm{KL}(\pi_\theta(a|s) \parallel \pi_\phi(a|s))$. Without loss of generality, we ignore the hyperprior and analyse Eq. (42) instead.

As discussed by Abdolmaleki et al. [1], the MPO objective is similar to the PPO [52] objective with the KL-direction reversed. In our E-step, we find a new variational distribution $\pi_{\theta_{k+1}}(a|s)$ that maximises the ELBO with ω_k fixed: Doing so yields an identical E-step to MPO. In parametric form, we can use gradient ascent and apply the same analysis as in Appendix E.1, obtaining an update

E-Step (MPO): $\theta_{i+1} \leftarrow \theta_i + \alpha_{\text{actor}} \left(\varepsilon_{\omega_k} \nabla_\theta \mathcal{L}(\omega_k, \phi_k, \theta)|_{\theta=\theta_i} \right)$,

$$\varepsilon_{\omega_k} \nabla_\theta \mathcal{L}(\omega_k, \phi_k, \theta) = \mathbb{E}_{d(s)} \left[\mathbb{E}_{\pi_\theta(a|s)} \left[\hat{Q}_{\omega_k}(h) \nabla_\theta \log \pi_\theta(a|s) \right] - \varepsilon_{\omega_k} \nabla_\theta \mathrm{KL}(\pi_\theta(a|s) \parallel \pi_{\phi_k}(a|s)) \right]. \tag{43}$$

As a point of comparison, Abdolmaleki et al. [1] motivate the update in Eq. (43) by carrying out a partial E-step, maximising the "one-step" KL-regularised pseudo-likelihood objective. In our framework, maximising Eq. (43) constitutes a full E-step, without requiring approximation.

In our M-step, we maximise the LML using the posterior derived from the E-step, yielding the update:

M-Step (MPO): $\omega_{k+1}, \phi_{k+1} \leftarrow \arg\max_{\omega,\phi} \mathcal{L}(\omega, \phi, \theta_{k+1})$,

$$\operatorname*{arg\,max}_{\omega,\phi} \mathcal{L}(\omega, \phi, \theta_{k+1}) = \operatorname*{arg\,max}_{\omega,\phi} \left(\mathbb{E}_{d(s)} \left[\mathbb{E}_{\pi_{\theta_{k+1}}} \left[\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega} \right] - \mathrm{KL}(\pi_{\theta_{k+1}} \parallel \pi_\phi(a|s)) \right] \right).$$

Maximising for ϕ can be achieved exactly by setting $\pi_\phi(a|s) = \pi_{\theta_{k+1}}(a|s)$, under which $\mathrm{KL}(\pi_{\theta_{k+1}} \parallel \pi_\phi(a|s)) = 0$. Maximising for ω is equivalent to finding $\arg\max_\omega \mathbb{E}_{d(s)\pi_{\theta_{k+1}}} \left[\frac{\hat{Q}_\omega(h)}{\varepsilon_\omega} \right]$, which accounts for the missing policy evaluation step, and can be implemented using the gradient ascent updates from Eq. (7). Setting $\pi_\phi(a|s) = \pi_{\theta_{k+1}}(a|s)$ is exactly the M-step update for MPO and, like in TRPO [51], means that $\pi_\phi(a|s)$ can be interpreted as the old policy, which is updated only after policy improvement. The objective in Eq. (42) therefore prevents policy improvement from straying too far from the old policy, adding a penalisation term $\mathrm{KL}(\pi_\theta(a|s) \parallel \pi_{\text{OLD}}(a|s))$ to the classic RL objective.

H Variational Actor-Critic Algorithm Pseudocode

Algorithms 1 and 2 show the pseudocode for the variational actor-critic algorithms *virel* and *beta* described in Section 5. The respective objectives are:

$$\begin{aligned}
J^V(\phi) =& \mathbb{E}_{s_t \sim \mathcal{D}} \left[\frac{1}{2} \left(V_\phi(s_t) - \mathbb{E}_{a_t \sim \pi_\theta} \left[Q_\omega(s_t, a_t) \right] \right)^2 \right], \\
J^Q(\omega) =& \mathbb{E}_{(h_t, r_t, s_{t+1}) \sim \mathcal{D}} \left[\frac{1}{2} \left(r_t + \gamma V_{\bar{\phi}}(s_{t+1}) - Q_\omega(h_t) \right)^2 \right], \\
J^{\pi^q}_{\mathit{virel}}(\theta) =& \mathbb{E}_{h_t \sim \mathcal{D}} \left[\log \pi_\theta(a_t|s_t) (\alpha - (Q_\omega(h_t) - V_{\bar{\phi}}(s_t))) \right], \\
J^{\pi^q}_{\mathit{beta}}(\theta) =& \mathbb{E}_{h_t \sim \mathcal{D}} \left[\log \pi_\theta(a_t|s_t) \left(\frac{1-\gamma}{r_{avg}} \varepsilon_\omega - (Q_\omega(h_t) - V_{\bar{\phi}}(s_t)) \right) \right].
\end{aligned}$$

Note that the derivative of the policy objectives can be found using the reparametrisation trick [32, 29], which we use for our implementation.

Algorithm 1 Variational Actor-Critic: *virel*

```
Initialize parameter vectors $\phi$, $\bar{\phi}$, $\theta$, $\omega$, $\mathcal{D} \leftarrow \{\}$

for each iteration do
    for each environment step do
        $a_t \sim \pi^q(a|s;\theta)$
        $s_{t+1} \sim p(s_{t+1}|s_t, a_t)$
        $\mathcal{D} \leftarrow \mathcal{D} \cup \{(s_t, a_t, r(s_t, a_t), s_{t+1})\}$
    end for
    for each gradient step do
        $\phi \leftarrow \phi - \lambda_V \hat{\nabla}_\phi J^V(\phi)$ (M-step)
        $\omega \leftarrow \omega - \lambda_Q \hat{\nabla}_\omega J^Q(\omega)$ (M-step)
        $\theta \leftarrow \theta - \lambda_{\pi^q} \hat{\nabla}_\theta J^{\pi^q}_{virel}(\theta)$ (E-step)
        $\bar{\phi} \leftarrow \tau\phi + (1-\tau)\bar{\phi}$
    end for
end for
```

Algorithm 2 Variational Actor-Critic: *beta*

```
Initialize parameter vectors $\phi$, $\bar{\phi}$, $\theta$, $\omega$, $\mathcal{D} \leftarrow \{\}$

for each iteration do
    for each environment step do
        $a_t \sim \pi^q(a|s;\theta)$
        $s_{t+1} \sim p(s_{t+1}|s_t, a_t)$
        $\mathcal{D} \leftarrow \mathcal{D} \cup \{(s_t, a_t, r(s_t, a_t), s_{t+1})\}$
    end for
    for each gradient step do
        $\varepsilon_\omega \leftarrow \mathbb{E}_{\mathcal{D}}\left[\left(r_t + \gamma V_{\bar{\phi}}(s_{t+1}) - Q_\omega(h_t)\right)^2\right]$
        $\phi \leftarrow \phi - \lambda_V \hat{\nabla}_\phi J^V(\phi)$ (M-step)
        $\omega \leftarrow \omega - \lambda_Q \hat{\nabla}_\omega J^Q(\omega)$ (M-step)
        $\theta \leftarrow \theta - \lambda_{\pi^q} \hat{\nabla}_\theta J^{\pi^q}_{\mathit{beta}}(\theta)$ (E-step)
        $\bar{\phi} \leftarrow \tau\bar{\phi} + (1-\tau)\bar{\phi}$
    end for
end for
```

I Experimental details

I.1 Parameter Values

Note that instead of specifying temperature c, we fix $c = 1$ for all implementations and scale reward.

Table 1: Summary of Experimental Parameter Values

PARAMETER	**VALUE**
Steps per evaluation	1000
Path Length	999
Discount factor	0.99
Mujoco-v2 Experiments:	
Batch size	128
Net size	300
$\lambda_\beta \approx \frac{1-\gamma}{r_{avg}}$	Humanoid 4e-4 All other 4e-3
Reward scale	Hopper, Half-Cheetah 5 Walker 3 All other 1
Value function learning rate	3e-4
Policy learning rate	3e-4
MLP layout as given in `https://github.com/vitchyr/rlkit`	
Mujoco-v1 Experiments:	
Values as used by Haarnoja et al. [25] in `https://github.com/haarnoja/sac`	

I.2 Additional MuJoCo-v1 Experiments

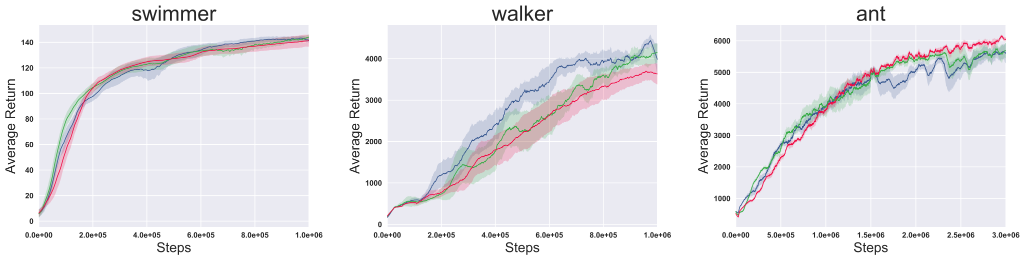

Figure 7: Training curves on additional continuous control benchmarks Mujoco-v1.

I.3 Additional MuJoCo-v2 Experiments

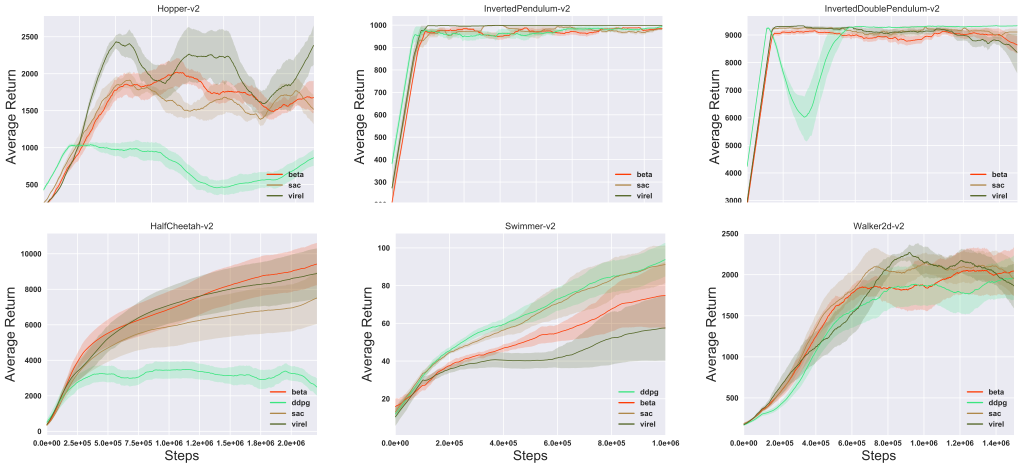

Figure 8: Training curves on additional continuous control benchmarks gym-Mujoco-v2.