Variational Information Maximizing Exploration

Xinghu Yao

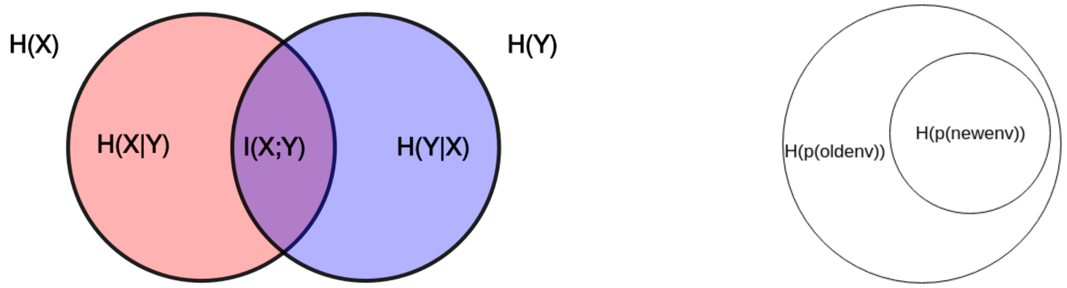
October 21, 2018

1 Exploration and Exploitation

Reinforcement learning is learning what to do–how to map situations to actions–so as to maximize numerical reward signal. And reinforcement learning algorithms involve a fundamental choice between Exploration and Exploitation. In exploration, the agent experiments with novel strategies that may improve returns in the long run; in exploitation, it maximizes rewards through behavior that is known to be successful. Variational Information Maximizing Exploration (VIME) is an exploration strategy based on maximizing of information gain about agent's belief of environment dynamics.

2 Intuition of VIME

Usually, there are three different kinds of approaches to exploration:
(1) Random exploration, which explore random actions, such as ϵ-greedy and softmax;
(2) Optimism in the face of uncertainty, which estimate uncertainty on value and prefer to explore states/actions with highest uncertainty;
(3) Information state space, which consider agent's information as part of its state and lookahead to see how information helps reward.
As a curiosity-driven strategy, VIME makes use of information gain about the agent's internal belief of the dynamics model as a driving force. We can model the environment dynamics by a probability model and the agent must take actions to maximize the information gain about the environment dynamics. Fig.1(a) shows the relationship between two variables X and Y.



(a) mutual information of X and Y.

(b) Mutual information of two environment.

Figure 1: Mutual Information.

Maximizing the reduction in uncertainty about the dynamics can be formalized as maximizing reduction of entropy

$$H(p(\text{oldenv})) - H(p(\text{newenv})) \tag{1}$$

where $p(\text{oldenv})$ and $p(\text{newenv})$ represent the distribution of old environment dynamics and new environment dynamics after the action. This is shown in Fig.1(b) where the difference between the larger circle and the smaller circle can be seen as the information gain of the action.

3 Methodology

3.1 Preliminaries

This paper assumes a finite-horizon discounted Markov decision process (MDP) defined by $(S, A, P, r, \rho_0, \gamma, T)$, in which $S \subseteq \mathbb{R}^n$ is a state set, $A \subseteq \mathbb{R}^m$ an action set, $P : S \times A \times S \to \mathbb{R}_{\geq 0}$ a transition probability distribution, $r : S \times A \to \mathbb{R}_{\geq 0}$ a bounded reward function, $\rho_0 : S \to \mathbb{R}_{\geq 0}$ an initial state distribution, $\gamma \in (0, 1]$ a discount factor, and T the horizon. States and actions viewed as random variables are abbreviated as S and A. The presented models are based on the optimization of a stochastic policy $\pi_a : S \times A \to \mathbb{R}_{\geq 0}$, parametrized by α. Let $\mu(\pi_a)$ denote its expected discounted return: $\mu(\pi_a) = \mathbb{E}_\tau[E_{t=0}^T \gamma^t r(s_t, a_t)]$, where $\tau = (s_0, a_0, ...)$ denotes the whole trajectory, $s_0 \sim \rho_0(s_0), a_t \sim \pi_\alpha(a_t|s_t)$, and $s_{t+1} \sim P(s_{t+1}|s_t, a_t)$.

3.2 Curiosity

The agent models the environment dynamics via a model $p(s_{t+1}|s_t, a_t; \theta)$, parameterized by the random variable θ with values $\theta \in \Theta$. Assuming a prior $p(\theta)$, it maintains a distribution over dynamic model through a distribution over θ, which is updated in a Bayesian manner. The history of the agent up until time step t is denoted as $\xi_t = s_1, a_1, ..., s_t$. According to curiosity-driven exploration, the agent should take actions that maximize the reduction in uncertainty about the dynamics. This can be formalized as maximizing reduction of entropy

$$H(\Theta|\xi_t, a_t) - H(\Theta|S_{t+1}, \xi_t, a_t) \tag{2}$$

through action a_t. According to information theory, Eq. (2) equals to the mutual information $I(S_{t+1}; \Theta|\xi_t, a_t)$ and we have the following equations

$$\begin{aligned} I(S_{t+1}; \Theta|\xi_t, a_t) &= \sum_{s_{t+1}, \theta} p(s_{t+1}, \theta|\xi_t, a_t) \text{log} \frac{p(s_{t+1}, \theta|\xi_t, a_t)}{p(s_{t+1}|\xi_t, a_t) p(\theta|\xi_t, a_t)} \\ &= \sum_{s_{t+1}} p(s_{t+1}|\xi_t, a_t) \sum_{\theta} p(\theta|s_{t+1}, \xi_t, a_t) \text{log} \frac{p(\theta|\xi_t, a_t, s_{t+1})}{p(\theta|\xi_t, a_t)} \\ &= \sum_{s_{t+1}} p(\theta|\xi_t, a_t) D_{\mathbb{KL}} \left[p(\theta|\xi_t, a_t, s_{t+1}) \| p(\theta|\xi_t, a_t) \right] \\ &= \mathbb{E}_{s_{t+1} \sim p(\cdot|\xi_t, a_t)} \left[D_{\mathbb{KL}} \left[p(\theta|\xi_t, a_t, s_{t+1}) \| p(\theta|\xi_t, a_t) \right] \right], \end{aligned} \tag{3}$$

the KL divergence from the agent's new belief over the dynamics model to the old one, taking expectation over all possible next states according to the true dynamics P. This KL divergence can be interpreted as information gain. By adding $T(s_{t+1}, \Theta|\xi_t, s_t)$ as a term of intrinsic reward to the external reward given by the environment dynamics, we can get the following trade-off between exploitation and exploration:

$$r'(s_t, a_t, s_{t+1}) = r(s_t, a_t) + \eta D_{\mathbb{KL}} \left[p(\theta|\xi_t, a_t, s_{t+1}) \| p(\theta|\xi_t, a_t) \right], \tag{4}$$

with $\eta \in \mathbb{R}_+$ a hyperparameter controlling the urge to explore. In conclusion, the biggest practical issue with maximizing information gain for exploration is that the computation of Eq. (4) requires calculating the posterior $p(\theta|\xi_t, a_t, s_{t+1})$, which is generally intractable.

4 Variational Bayes

To facilitate the subsequent discussion under a probabilistic framework, we make the following assumptions:

Assumption 1. The models of the environment under consideration are fully described by a random element Θ which depends solely on the environment. Moreover, the agent's initial knowledge about Θ is summarized by a prior density $p(\theta)$.

Assumption 2: The agent is equipped with a conditional predictor $p(s_{t+1}|\xi_t, a_t; \theta)$, i.e. the agent is capable of refining its prediction in the light of information about Θ.

We can derive the posterior distribution given a new state-action pair through Bayes' rule as

$$\begin{aligned} p(\theta|\xi_t, a_t, s_{t+1}) &= \frac{p(s_{t+1}, \theta|\xi_t, a_t)}{p(\xi_t, a_t, s_{t+1})} \\ &= \frac{p(\theta|\xi_t, a_t)p(s_{t+1}|\xi_t, a_t; \theta)}{\int_\theta p(s_{t+1}|\xi_t, a_t; \theta)p(\theta|\xi_t, a_t)}. \end{aligned} \tag{5}$$

Notice that knowing the action without subsequent observation cannot change the agent's state of knowledge about Θ, so we have $p(\theta|\xi_t, a_t) = p(\theta|\xi_t)$. The integral $\int_\theta p(s_{t+1}|\xi_t, a_t; \theta)p(\theta|\xi_t, a_t)$ tends to be intractable when using highly parametrized models (e.g., neural networks), which are often needed to accurately capture the environment model in high-dimensional continuous control.

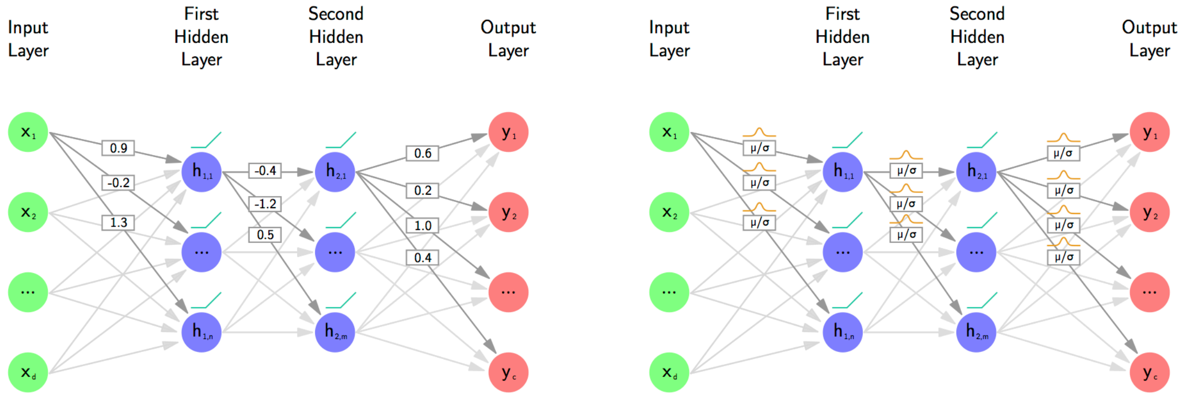
Herein, we embrace the fact that calculating the posterior $p(\theta|D)$ for a data set D is intractable. Instead we approximate it through an alternative distribution $q(\theta;\phi)$, parameterized by ϕ, and by minimizing $D_{\mathrm{KL}}[q(\theta;\phi)\|p(\theta|D)]$. We can derive the following equations:

$$\begin{aligned} &\underset{\phi}{\operatorname{argmin}} D_{\mathrm{KL}}[q(\theta;\phi)\|p(\theta|D)] \\ &\quad= \underset{\phi}{\operatorname{argmax}}\, \mathrm{ELBO} \\ &\quad= \underset{\phi}{\operatorname{argmax}} \int_\theta q(\theta;\phi) \log \frac{p(\theta, D)}{q(\theta;\phi)} \\ &\quad\overset{p(\theta,D)=p(\theta)p(D|\theta)}{=} \underset{\phi}{\operatorname{argmax}} \left\{ \int_\theta q(\theta;\phi)\log p(D|\theta) - \int_\theta q(\theta;\phi)\log\frac{q(\theta;\phi)}{p(\theta)} \right\} \\ &\quad= \underset{\phi}{\operatorname{argmax}} \left\{ \mathbb{E}_{\theta\sim q(\cdot;\phi)}[\log p(D|\theta)] - D_{KL}[q(\theta;\phi)\|p(\theta)] \right\}. \end{aligned} \tag{6}$$

Rather than computing information gain in Eq. (4) explicitly, we compute an approximation to it, leading to the following total reward:

$$r'(s_t, a_t, s_{t+1}) = r(s_t, a_t) + \eta D_{KL}[q(\theta;\phi_{t+1})\|q(\theta;\phi_t)] \tag{7}$$

with ϕ_{t+1} the updated and ϕ_t the old parameters representing the agent's belief. Natural candidates for parameterizing the agent's dynamics model are Bayesian neural networks (BNNs), as they maintain a distribution over their weights.



(a) Classic neural network.

(b) Bayesian neural network.

Figure 2: Classic NN v.s. BNN.

5 Bayesian neural networks

The goal of traditional neural is to find an optimal point estimate for the weights. This usually perform well in regions with lots of data but fail to express uncertainty in regions with little or no data, and leading to overconfident decisions. This drawback motivates the application of Bayesian learning to neural networks, introducing probability distributions over the weights. These distributions can be of various in theory. To make our lifes easier and to have an intuitive understanding of the distribution at each weight, we will use a Gaussian distribution. Fig. (2) shows the difference of this two kind of neural networks.

5.1 Build objective/loss

We will use variational inference in order to make the prediction of the posterior tractable. While we cannot model the posterior $p(w|D)$ directly, we try to find the parameters θ of a distribution on the weights $q(w|\theta)$ (commonly referred as the variational posterior) that minimizes the KL divergence with the true posterior. Formally, this can be expressed as:

$$\begin{aligned}\theta^* &= \underset{\theta}{\operatorname{argmin}} D_{\mathbb{KL}}[q(w|\theta)\|p(w|D)] \\ &= \underset{\theta}{\operatorname{argmin}} \int q(w|\theta)\log\frac{q(w|\theta)}{p(w)p(D|w)}\mathrm{d}w \\ &= \underset{\theta}{\operatorname{argmin}} D_{\mathbb{KL}}[q(w|\theta)\|p(w)] - \mathbb{E}_{q(w|\theta)}[\log p(D|w)] \end{aligned} \tag{8}$$

The resulting loss function, commonly referred to as either variational free energy or expected lower bound (ELBO), has to be minimized and is then given as follows:

$$F(D,\theta) = D_{\mathbb{KL}}[q(w|\theta)\|p(w)] - \mathbb{E}_{q(w|\theta)}[\log p(D|w)] \tag{9}$$

As one can easily see, the cost function tries to balance the complexity of the data $P(D|w)$ and the simplicity of the prior $P(w)$.

Unlike previous work, we do not use the closed form of the complexity cost: not requiring a closed form of the complexity cost allows many more combinations of prior and variational posterior families. Indeed this scheme is also simple to implement and allows prior/posterior combinations to be interchanged. We can approximate this exact cost through a Monte Carlo sampling procedure as follows

$$F(D,\theta) \simeq \sum_{i=1}^{n} \left\{ \log q(w^i|\theta) - \log P(w^i) - \log P(D|w^{(i)}) \right\} \tag{10}$$

where $w^{(i)}$ corresponds to the i-th Monte Carlo sample from the variational posterior $q(w^{(i)}|\theta)$. Note that every term of this approximate cost depends upon on the particular weights drawn from the variational posterior: this is an instance of a variational reduction technique known as common random numbers.

5.2 Gaussian variational posterior

Suppose that the variational posterior is a diagonal Gaussian distribution and the BNN weight distribution $q(\theta;\phi)$ is given by the fully factorized Gaussian distribution:

$$q(\theta;\phi) = \prod_{i=1}^{|\Theta|} \mathcal{N}(\theta_i|\mu_i;\delta_i^2) \tag{11}$$

then a sample of the weights w can be obtained by sampling a unit Gaussian, shifting it by a mean μ and scaling by a standard deviation σ. We parameterize the standard deviation pointwise as $\sigma = \log(1+\exp(\rho))$ and so ρ is always non-negative. The variational posterior parameters are $\theta = (\mu, \rho)$.

6 Implementation

6.1 KL Divergence for Gaussian distribution

Firstly, recall that the KL divergence between two distributions P and Q is definedas:

$$D_{\mathbb{KL}}(P\|Q) = \mathbb{E}_P\left[\log\frac{P}{Q}\right] \tag{12}$$

Also, the density function for a multivariate Gaussian distribution with mean μ and covariance matrix Σ is

$$P(x) = \frac{1}{(2\pi)^{n/2}\mathrm{det}(\Sigma)^{1/2}}\exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right) \tag{13}$$

Now, consider two multivariate Gaussians in $\mathbb{R}^n$, P_1 and P_2, we have

$$\begin{aligned}
&D_{\mathbb{KL}}(P_1\|P_2)\\
&\quad= \mathbb{E}_{P_1}[\mathrm{log}P_1 - \mathrm{log}_{P_2}]\\
&\quad= \frac{1}{2}\mathbb{E}_{P_1}[-\mathrm{logdet}\Sigma_1 - (x-\mu_1)^T\Sigma_1^{-1}(x-\mu_1) + \mathrm{logdet}\Sigma_2 + (x-\mu_2)^T\Sigma_2^{-1}(x-\mu_2)]\\
&\quad= \frac{1}{2}\mathrm{log}\frac{\mathrm{det}\Sigma_2}{\mathrm{det}\Sigma_1} + \frac{1}{2}\mathbb{E}_{P_1}[-(x-\mu_1)^T\Sigma_1^{-1}(x-\mu_1) + (x-\mu_2)^T\Sigma_2^{-1}(x-\mu_2)]\\
&\quad= \frac{1}{2}\mathrm{log}\frac{\mathrm{det}\Sigma_2}{\mathrm{det}\Sigma_1} + \frac{1}{2}\mathbb{E}_{P_1}\left[-\mathrm{tr}\left(\Sigma_1^{-1}(x-\mu_1)(x-\mu_1)^T\right) + \mathrm{tr}\left(\Sigma_2^{-1}(x-\mu_2)(x-\mu_2)^T\right)\right]\\
&\quad= \frac{1}{2}\mathrm{log}\frac{\mathrm{det}\Sigma_2}{\mathrm{det}\Sigma_1} + \frac{1}{2}\mathbb{E}_{P_1}\left[-\mathrm{tr}\left(\Sigma_1^{-1}\Sigma_1\right) + \mathrm{tr}\left(\Sigma_2^{-1}(xx^T - 2x\mu_2^T + \mu_2\mu_2^T)\right)\right]\\
&\quad= \frac{1}{2}\mathrm{log}\frac{\mathrm{det}\Sigma_2}{\mathrm{det}\Sigma_1} - \frac{1}{2}n + \frac{1}{2}\mathrm{tr}\left(\Sigma_2^{-1}(\Sigma_1 + \mu_1\mu_1^T - 2\mu_2\mu_1^T + \mu_2\mu_2^T)\right)\\
&\quad= \frac{1}{2}\left(\mathrm{log}\frac{\mathrm{det}\Sigma_2}{\mathrm{det}\Sigma_1} - n + \mathrm{tr}(\Sigma_2^{-1}\Sigma_1) + \mathrm{tr}(\mu_1^T\Sigma_2^{-1}\mu_1 - 2\mu_1^T\Sigma_2^{-1}\mu_2 + \mu_2^T\Sigma_2^{-1}\mu_2)\right)\\
&\quad= \frac{1}{2}\left(\mathrm{log}\frac{\mathrm{det}\Sigma_2}{\mathrm{det}\Sigma_1} - n + \mathrm{tr}(\Sigma_2^{-1}\Sigma_1) + (\mu_2-\mu_1)^T\Sigma_2^{-1}(\mu_2-\mu_1)\right) \qquad (14)
\end{aligned}$$

6.2 *2nd order Taylor expansion of KL divergence

$$\begin{aligned}
&D_{\mathbb{KL}}\left(p(x;\theta)\|p(x:\theta+\delta\theta)\right)\\
&= \int_{x\sim p(x;\theta)} p(x;\theta)\mathrm{log}\frac{p(x;\theta)}{p(x;\theta+\delta\theta)}\\
&\simeq \int_{x\sim p(x;\theta)} p(x;\theta)\left(\mathrm{log}\frac{p(x;\theta)}{p(x;\theta)} - \frac{d}{d\theta}\mathrm{log}p(x;\theta)^T\delta\theta - \frac{1}{2}\delta\theta^T\frac{d}{d\theta^2}\mathrm{log}p(x;\theta)\delta\theta\right)\\
&= \int_{x\sim p(x;\theta)} p(x;\theta)\left(-\left(\frac{\frac{\mathrm{d}}{\mathrm{d}\theta}p(x;\theta)}{p(x;\theta)}\right)^T\delta\theta - \frac{1}{2}\delta\theta^T\frac{\mathrm{d}}{\mathrm{d}\theta}\left(\frac{\frac{\mathrm{d}}{\mathrm{d}\theta}p(x;\theta)}{p(x;\theta)}\right)\delta\theta\right)\\
&= \int_{x\sim p(x;\theta)}\left\{p(x;\theta)\left(-\left(\frac{\frac{\mathrm{d}}{\mathrm{d}\theta}p(x;\theta)}{p(x;\theta)}\right)^T\delta\theta - \frac{1}{2}\delta\theta^T\left(\frac{p(x;\theta)\frac{\mathrm{d}^2}{\mathrm{d}\theta^2}p(x;\theta) - \left(\frac{\mathrm{d}p(x;\theta)}{\mathrm{d}\theta}\right)\left(\frac{\mathrm{d}p(x;\theta)}{\mathrm{d}\theta}\right)^T}{p(x;\theta)^2}\right)\delta\theta\right)\right\}\\
&= -\int_{x\sim p(x;\theta)}\frac{\mathrm{d}}{\mathrm{d}\theta}p(x;\theta)^T\delta\theta - \frac{1}{2}\delta\theta^T\int_{x\sim p(x;\theta)}\frac{\mathrm{d}^2}{\mathrm{d}\theta^2}p(x;\theta)\delta\theta + \frac{1}{2}\delta\theta^T\int_{x\sim p(x;\theta)}p(x;\theta)\left(\frac{\frac{\mathrm{d}p(x;\theta)}{\mathrm{d}\theta}}{p(x;\theta)}\right)\left(\frac{\frac{\mathrm{d}p(x;\theta)}{\mathrm{d}\theta}}{p(x;\theta)}\right)^T\delta\theta\\
&= -\left(\frac{\mathrm{d}}{\mathrm{d}\theta}\int_{x\sim p(x;\theta)}p(x;\theta)\right)^T\delta\theta - \frac{1}{2}\delta\theta^T\left(\frac{\mathrm{d}^2}{\mathrm{d}\theta^2}\int_{x\sim p(x;\theta)}p(x;\theta)\right)\delta\theta\\
&\qquad\qquad + \frac{1}{2}\delta\theta^T\left(\int_{x\sim p(x;\theta)}p(x;\theta)\left(\frac{\mathrm{d}}{\mathrm{d}\theta}\mathrm{log}p(x;\theta)\right)\right)\left(\frac{\mathrm{d}}{\mathrm{d}\theta}\mathrm{log}p(x;\theta)\right)^T\delta\theta\\
&= -\left(\frac{\mathrm{d}}{\mathrm{d}\theta}\mathbb{I}\right)^T\delta\theta - \frac{1}{2}\delta\theta^T\left(\frac{\mathrm{d}}{\mathrm{d}\theta^2}\mathbb{I}\right)\delta\theta + \frac{1}{2}\delta\theta^T\left(\int_{x\sim p(x;\theta)}p(x;\theta)\left(\frac{\mathrm{d}}{\mathrm{d}\theta}\mathrm{log}p(x;\theta)\right)\right)\left(\frac{\mathrm{d}}{\mathrm{d}\theta}\mathrm{log}p(x;\theta)\right)^T\delta\theta\\
&= -\mathbf{0} - \mathbf{0} + \frac{1}{2}\delta\theta^T\left(\int_{x\sim p(x;\theta)}p(x;\theta)\left(\frac{\mathrm{d}}{\mathrm{d}\theta}\mathrm{log}p(x;\theta)\right)\right)\left(\frac{\mathrm{d}}{\mathrm{d}\theta}\mathrm{log}p(x;\theta)\right)^T\delta\theta\\
&= \frac{1}{2}\delta\theta^T G(\theta)\delta\theta \qquad (15)
\end{aligned}$$

$G(\theta)$ = Fisher information matrix, which is independent of the choice of parameterization of the class of distribution.
Nature gradient g_N = the direction with highest increase in the objective per change in KL divergence.

$$\begin{aligned} g_N = & \underset{\delta\theta:\mathbb{KL}(p(\tau;\theta)\|p(\tau;\theta+\delta\theta))\leq\epsilon}{\text{argmax}} f(\theta+\delta\theta) \\ \simeq & \underset{\delta\theta:\mathbb{KL}(p(\tau;\theta)\|p(\tau;\theta+\delta\theta))\leq\epsilon}{\text{argmax}} f(\theta)+\nabla_\theta f(\theta)^T\delta\theta \\ = & \underset{\delta\theta:\mathbb{KL}(p(\tau;\theta)\|p(\tau;\theta+\delta\theta))\leq\epsilon}{\text{argmax}} \nabla_\theta f(\theta)^T\delta\theta \\ & = G(\theta)^{-1}\nabla_\theta f(\theta) \end{aligned} \tag{16}$$

It is worthy to say that we don't use the results of this section in the paper of VIME.

6.3 Newton's Method

In contrast to first-order methods, second-order methods make use of second order derivatives to improve optimization. The most widely used second-order methods is Newton's method.
Newton's method is an optimization scheme based on using a second-order Taylor series expansion to approximate $J(\theta)$ near some points θ_0 ignoring derivatives of higher order"

$$J(\theta) \simeq J(\theta_0) + (\theta-\theta_0)^T\nabla_\theta J(\theta_0) + \frac{1}{2}(\theta-\theta_0)^T H(\theta-\theta_0), \tag{17}$$

where H is the Hessian of J with respect to θ evaluated at θ_0. If we then solve for the critical point of this function, we obtain the Newton parameter update rule:

$$\theta^* = \theta_0 - H^{-1}\nabla_\theta J(\theta_0). \tag{18}$$

Thus for a locally quadratic function (with positive definite H), by rescaling the gradient by H^{-1}, Newton's method jumps directly to the minimum, If the objective function is convex but not quadratic (there are higher-order terms), this update can be iterated.

6.4 Optimization

The posterior distribution of the dynamics parameter, which is needed to compute the KL divergence in the total reward function r' of Eq. (4), can be compute through the following minimization

$$\phi' = \underset{\phi}{\text{argmin}} \left[\overbrace{\underbrace{D_{\mathbb{KL}}[q(\theta;\phi)\|q(\theta;\phi_{t-1})}_{l_{\mathbb{KL}(q(\theta;\phi))}]} - \mathbb{E}_{\theta\sim q(\cdot;\phi)}[\log p(s_t|\xi_{t-1},a_t;\theta)]}^{l(q(\theta;\phi),s_t)} \right], \tag{19}$$

where we replace the expectation over θ with samples $\theta \sim q(\cdot;\phi)$. Because we only update the model periodically based on samples drawn from the replay poll, this optimization can be performed in parallel for each s_t, keeping ϕ_{t-1} fixed. Once ϕ' has been obtained, we can use it to compute the intrinsic reward. To optimize Eq. (19) efficiently, we only take a single second-order step. This way, the gradient is rescaled according to the curvature of the KL divergence at the origin. As such, we compute $D_{\mathbb{KL}}[q(\theta;\phi+\lambda\Delta\phi)\|q(\theta;\phi)]$, with the update step $\Delta\phi$ defined as

$$\Delta\phi = H^{-1}(l)\nabla_\phi l(q(\theta;\phi),s_t), \tag{20}$$

in which $H(l)$ is the Hessian of $l(q(\theta;\phi),s_t)$. Since we assume that the variational approximation is a fully factorized Gaussian, the KL divergence from posterior to prior has a particularly simple form according to Eq. (14):

$$D_{\mathbb{KL}}[q(\theta;\phi)\|q(\theta;\phi')] = \frac{1}{2}\sum_{i=1}^{|\Theta|}\left(\left(\frac{\sigma_i}{\sigma_i^{'}}\right)^2 + 2\log\sigma_i^{'} - 2\log\sigma_i + \frac{(\mu_u^{'}-\mu_i)}{\sigma_i^{'2}}\right) - \frac{|\Theta|}{2}. \tag{21}$$

Because this KL divergence is approximately quadratic in tis parameters and the log-likelihood term can be seen as local linear compared to this highly curved KL term, we approximate H by only calculate it for the term

KL term $l_{\mathbb{KL}}(q(\theta;\phi))$. This can be computed very efficiently in case of a fully factorized Gaussian distribution, as this approximation becomes a diagonal matrix. Looking at Eq. (21), we can calculate the following Hessian at the origin. The μ and ρ entries are defined as

$$\frac{\partial^2 l_{\mathbb{KL}}}{\partial \mu_i^2} = \frac{1}{log^2(1+e^{\rho_i})} \qquad \text{and} \qquad \frac{\partial^2 l_{\mathbb{KL}}}{\partial \rho_i^2} = \frac{2e^{2\rho_i}}{(1+e^{\rho_i})^2} \frac{1}{\log^2(1+e^{\rho_i})}, \tag{22}$$

while all other entries are zero. Furthermore, it is also possible to approximate the KL divergence through a second-order Taylor expansion as $\frac{1}{2}\Delta\phi H \Delta\phi = \frac{1}{2}(H^{-1}\nabla)^T H(H^(- 1)\nabla)$, since both the value and gradient of the Kl divergence are zero at the origin. This gives us

$$D_{\mathbb{KL}}[q(\theta;\phi+\lambda\Delta\phi)\|q(\theta;\phi)] \simeq \frac{1}{2}\lambda^2 \nabla_\phi l^T H^{-1}(l_{\mathbb{KL}})\nabla_\phi l \tag{23}$$

Note that $H^{-1}(l_{\mathbb{KL}})$ is diagonal, so this expression can be computed efficiently.