Appendix of:
Inverse Reinforcement Learning via Deep Gaussian Process

1 Background: Inverse Reinforcement Learning and DGP-IRL

The Markov Decision Process (MDP) is characterized by $\{\mathcal{S}, \mathcal{A}, \mathcal{T}, \gamma, \boldsymbol{r}\}$, which represents the state space, action space, transition model, discount factor, and reward function, respectively.

The IRL task is to find the reward function r^* such that the induced optimal policy matches the demonstrations, given $\{\mathcal{S}, \mathcal{A}, \mathcal{T}, \gamma\}$ and $\mathcal{M} = \{\zeta_1, ..., \zeta_h\}$, where $\zeta_i = \{(s_{i,1}, a_{i,1}), ..., (s_{i,T}, a_{i,T})\}$ is the demonstration trajectory, consisting of state-action pairs.

Deep Gaussian process for inverse reinforcement learning (DGP-IRL) extends the deep Gaussian process (deep GP) framework to the IRL domain, as shown in Fig. 1. DGP-IRL learns an abstract representation that reveals the reward structure by warping the original feature space through the latent layers, $\mathbf{D}, \mathbf{B}$.

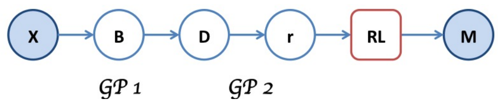

Figure 1: The proposed deep GP model for IRL, where latent Gaussian processes are introduced to learn a representation of the world for the latent reward $\boldsymbol{r}$. The rewards are provided to the reinforcement learning (RL) engine to generate a set of observable trajectories $\mathcal{M}$.

For a set of observed trajectories $\mathcal{M}$, our objective is to optimize the corresponding marginalized log-likelihood given the states in the world as represented by $\mathbf{X}$:

$$\log p(\mathcal{M}|\mathbf{X}) = \log \int p(\mathcal{M}|\boldsymbol{r}) p(\boldsymbol{r}|\mathbf{D}) p(\mathbf{D}|\mathbf{B}) p(\mathbf{B}|\mathbf{X}) d(\boldsymbol{r}, \mathbf{D}, \mathbf{B}) \tag{1}$$

where the integration is with respect to the latent layers, including the reward vector $\boldsymbol{r}$. As introduced in the main paper, $\mathbf{d}^m \in \mathbb{R}^n$ is the m-th column of the latent layer $\mathbf{D} = \begin{bmatrix}\mathbf{d}^1 & \cdots & \mathbf{d}^{m_1}\end{bmatrix}$, and similarly for $\mathbf{B} = \begin{bmatrix}\mathbf{b}^1 & \cdots & \mathbf{b}^{m_1}\end{bmatrix}$:

$$p(\mathcal{M}|\boldsymbol{r}) = \sum_{i=1}^{h} \sum_{t=1}^{T} \big(Q(s_{i,t}, a_{i,t}; \boldsymbol{r}) - V(s_{i,t}; \boldsymbol{r})\big) \tag{2}$$

$$p(\boldsymbol{r}|\mathbf{D}) = \mathcal{N}(\boldsymbol{r}|\mathbf{0}, K_{\mathbf{DD}}) \tag{3}$$

$$p(\mathbf{D}|\mathbf{B}) = \prod_{m=1}^{m_1} \mathcal{N}(\mathbf{d}^m|\mathbf{b}^m, \lambda^{-1}\mathbf{I}) \tag{4}$$

$$p(\mathbf{B}|\mathbf{X}) = \prod_{m=1}^{m_1} \mathcal{N}(\mathbf{b}^m|\mathbf{0}, K_{\mathbf{XX}}) \tag{5}$$

where $p(\mathcal{M}|\boldsymbol{r})$ represents the reinforcement learning term, given by:

$$\log p(\mathcal{M}|\boldsymbol{r}) = \sum_i \sum_t (Q(s_{i,t}, a_{i,t}; \boldsymbol{r}) - V(s_{i,t}; \boldsymbol{r})) \tag{6}$$

$$= \sum_t \sum_t \left(\boldsymbol{r}_{s_{i,t}, a_{i,t}} - V(s_{i,t}; \boldsymbol{r}) + \sum_{s'} \gamma \mathcal{T}_{s'}^{s_{i,t}, a_{i,t}} V(s'; \boldsymbol{r}) \right) \tag{7}$$

The Q-value $Q(s_{i,t}, a_{i,t}; \boldsymbol{r})$ used above is a measure of how desirable is the corresponding state-action pair $(s_{i,t}, a_{i,t})$ under rewards $\boldsymbol{r}$ for all the world states, and is defined by:

$$Q(s_{i,t}, a_{i,t}; \boldsymbol{r}) = \boldsymbol{r}_{s_{i,t}, a_{i,t}} + \sum_{s'} \gamma \mathcal{T}_{s'}^{s_{i,t}, a_{i,t}} V(s'; \boldsymbol{r})$$

where $\boldsymbol{r}_{s_{i,t},a_{i,t}} = r(s_{i,t}, a_{i,t}) \in \mathbb{R}$ is the reward for $(s_{i,t}, a_{i,t})$, γ is the discount factor, $\mathcal{T}_{s'}^{s_{i,t},a_{i,t}} = P(s'|s_{i,t}, a_{i,t})$ is the transition probability by the transition model, and $V(s_{i,t}; \boldsymbol{r})$ is the value associated with state $s_{i,t}$, obtained by the modified Bellman backup operator:

$$V(s_{i,t}; \boldsymbol{r}) = \log \sum_{a \in \mathcal{A}} \exp \left(\boldsymbol{r}_{s_{i,t},a_{i,t}} + \sum_{s'} \gamma \mathcal{T}_{s'}^{s_{i,t},a} V(s'; \boldsymbol{r}) \right)$$

where we apply a **soft-max function** $V(s_{i,t}; \boldsymbol{r}) = \log \sum_{a \in \mathcal{A}} \exp\left(Q(s_{i,t}, a; \boldsymbol{r})\right)$ for the Q-values with all possible actions $a \in \mathcal{A}$. The value function $V(s; \boldsymbol{r})$ for state s can be obtained by repeatedly applying the above Bellman backup operator. For simplicity of notations, we use $V(s_{i,t}; \boldsymbol{r}), Q(s_{i,t}, a_{i,t}; \boldsymbol{r})$ to denote the solution after Bellman backup operators, unlike some literature that uses $V^*(s_{i,t}; \boldsymbol{r}), Q^*(s_{i,t}, a_{i,t}; \boldsymbol{r})$ to denote the difference. Detailed derivation of the above relationships can be found in [4].

2 Variational Lower Bound for DGP-IRL

It is intractable to perform the integration as in (1) for the marginal log-likelihood. In addition to $p(\mathcal{M}|\boldsymbol{r})$, which involves the latent variable $\boldsymbol{r}$ in a way which requires Q-value iterations, the term $p(\boldsymbol{r}|\mathbf{D}) = \mathcal{N}(\boldsymbol{r}|\mathbf{0}, K_{\mathbf{DD}})$ has a nonlinear dependency on $\mathbf{D}$ in the kernel matrix.

To tackle this issue, we introduce inducing outputs $\boldsymbol{f}, \mathbf{V}$ and their corresponding inputs $\mathbf{Z}, \mathbf{W}$, as shown in Fig. 2. The resulting model follows the main paper:

$$\begin{aligned}
p(\mathcal{M}|\boldsymbol{r}) &= \sum_{i=1}^{h} \sum_{t=1}^{T} \big(Q(s_{i,t}, a_{i,t}; \boldsymbol{r}) - V(s_{i,t}; \boldsymbol{r}) \big) && (8)\\
p(\boldsymbol{r}|\boldsymbol{f}, \mathbf{D}, \mathbf{Z}) &= \mathcal{N}(\boldsymbol{r}|K_{\mathbf{DZ}} K_{\mathbf{ZZ}}^{-1} \boldsymbol{f}, \mathbf{0}) && (9)\\
p(\boldsymbol{f}|\mathbf{Z}) &= \mathcal{N}(\boldsymbol{f}|\mathbf{0}, K_{\mathbf{ZZ}}) && (10)\\
p(\mathbf{D}|\mathbf{B}) &= \prod_{m=1}^{m_1} \mathcal{N}(\mathbf{d}^m|\mathbf{b}^m, \lambda^{-1}\mathbf{I}) && (11)\\
p(\mathbf{B}|\mathbf{V}, \mathbf{X}, \mathbf{W}) &= \prod_{m=1}^{m_1} \mathcal{N}(\mathbf{b}^m|K_{\mathbf{XW}} K_{\mathbf{WW}}^{-1} \mathbf{v}^m, \boldsymbol{\Sigma}_B) && (12)
\end{aligned}$$

We also design the variation distribution as illustrated in the main paper:

$$\begin{aligned}
\mathcal{Q} &= q(\boldsymbol{f}) q(\mathbf{D}) p(\mathbf{B}|\mathbf{V}, \mathbf{X}) q(\mathbf{V}), \text{ with :}\\
q(\boldsymbol{f}) &= \delta(\boldsymbol{f} - \tilde{\boldsymbol{f}})\\
q(\mathbf{D}) &= \prod_{m=1}^{m_1} \delta\left(\mathbf{d}^m - K_{\mathbf{XW}} K_{\mathbf{WW}}^{-1} \tilde{\mathbf{v}}^m\right)\\
q(\mathbf{V}) &= \prod_{m=1}^{m_1} \mathcal{N}\left(\mathbf{v}^m|\tilde{\mathbf{v}}^m, \mathbf{G}^m\right),
\end{aligned}$$

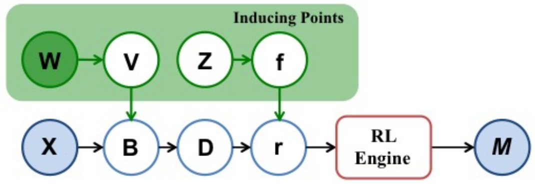

Figure 2: Illustration of DGP-IRL with the inducing outputs $\boldsymbol{f}, \mathbf{V}$ and the corresponding inputs $\mathbf{Z}, \mathbf{W}$.

where the variational distribution $\mathcal{Q}$ is to not be confused with the notation for Q-values, Q. Using the above distribution $\mathcal{Q}$, we can derive the variational lower bound as follows:

$$\log p(\mathcal{M}|\mathbf{X}, \mathbf{Z}, \mathbf{W}) = \log \int p(\mathcal{M}, \boldsymbol{r}, \boldsymbol{f}, \mathbf{V}, \mathbf{D}, \mathbf{B}|\mathbf{Z}, \mathbf{W}, \mathbf{X}) d(\boldsymbol{r}, \boldsymbol{f}, \mathbf{V}, \mathbf{D}, \mathbf{B}) \tag{13}$$

$$= \log \int \underbrace{p(\mathcal{M}|\boldsymbol{r}) p(\boldsymbol{r}|\boldsymbol{f}, \mathbf{D}, \mathbf{Z})}_{p(\mathcal{M}|K_{\mathbf{DZ}} K_{\mathbf{ZZ}}^{-1} \boldsymbol{f})} p(\boldsymbol{f}|\mathbf{Z}) p(\mathbf{D}|\mathbf{B}) p(\mathbf{B}|\mathbf{V}, \mathbf{W}, \mathbf{X}) p(\mathbf{V}|\mathbf{W}) d(\boldsymbol{r}, \boldsymbol{f}, \mathbf{V}, \mathbf{D}, \mathbf{B}) \tag{14}$$

$$\geq \int q(\boldsymbol{f}) q(\mathbf{D}) p(\mathbf{B}|\mathbf{V}, \mathbf{W}, \mathbf{X}) q(\mathbf{V}) \log \frac{p(\mathcal{M}|K_{\mathbf{DZ}} K_{\mathbf{ZZ}}^{-1} \boldsymbol{f}) p(\boldsymbol{f}|\mathbf{Z}) p(\mathbf{D}|\mathbf{B}) p(\mathbf{V}|\mathbf{W})}{q(\boldsymbol{f}) q(\mathbf{D}) q(\mathbf{V})} \tag{15}$$

$$\begin{aligned} &= \log p(\mathcal{M}|K_{\tilde{\mathbf{D}}\mathbf{Z}} K_{\mathbf{ZZ}}^{-1} \tilde{\boldsymbol{f}}) + \log p(\boldsymbol{f} = \tilde{\boldsymbol{f}}|\mathbf{Z}) \\ &\quad + \int q(\mathbf{V}) q(\mathbf{D}) p(\mathbf{B}|\mathbf{V}, \mathbf{W}, \mathbf{X}) \log \frac{p(\mathbf{D}|\mathbf{B}) p(\mathbf{V}|\mathbf{W})}{q(\mathbf{V})} d(\mathbf{D}, \mathbf{B}, \mathbf{V}) \end{aligned} \tag{16}$$

In the above derivation, the combination of $p(\mathcal{M}|\boldsymbol{r}) p(\boldsymbol{r}|\boldsymbol{f}, \mathbf{D}, \mathbf{Z})$ in (14) uses the deterministic training conditional (DTC) assumption [2], i.e., $p(\boldsymbol{r}|\boldsymbol{f}, \mathbf{D}, \mathbf{Z}) = \delta(\boldsymbol{r} - K_{\mathbf{DZ}} K_{\mathbf{ZZ}}^{-1} \boldsymbol{f})$, (15) applies Jensen's inequality with the variational distribution Q, (16) is a direct consequence of the choice of Q, and $\tilde{\mathbf{D}} = \begin{bmatrix} \tilde{\mathbf{d}}^1 & \cdots & \tilde{\mathbf{d}}^{m_1} \end{bmatrix}$, with $\tilde{\mathbf{d}}^m = K_{\mathbf{XW}} K_{\mathbf{WW}}^{-1} \tilde{\mathbf{v}}^m$.

Utility 1 (Gaussian identities) *If the marginal and conditional Gaussian distributions for* $\boldsymbol{f}$ *and* $\mathbf{v}$ *are in the form:*

$$p(\boldsymbol{f}|\mathbf{v}) = \mathcal{N}(\boldsymbol{f}|\mathbf{M}\mathbf{v} + \mathbf{m}, \boldsymbol{\Sigma}_{\boldsymbol{f}})$$
$$p(\mathbf{v}) = \mathcal{N}(\mathbf{v}|\boldsymbol{\mu}_{\mathbf{v}}, \boldsymbol{\Sigma}_{\mathbf{v}})$$

Then the marginal distribution of $\boldsymbol{f}$ *is:*

$$p(\boldsymbol{f}) = \mathcal{N}(\boldsymbol{f}|\mathbf{M}\boldsymbol{\mu}_{\mathbf{v}} + \mathbf{m}, \boldsymbol{\Sigma}_{\boldsymbol{f}} + \mathbf{M}\boldsymbol{\Sigma}_{\mathbf{v}}\mathbf{M}^{\top}) \tag{17}$$

Using the Gaussian identities, the derivation of $\int q(\mathbf{V}) p(\mathbf{B}|\mathbf{V}, \mathbf{W}, \mathbf{X}) d\mathbf{V}$ is as follows:

$$\begin{aligned} \int q(\mathbf{V}) p(\mathbf{B}|\mathbf{V}, \mathbf{W}, \mathbf{X}) d\mathbf{V} &= \int \prod_{m=1}^{m_1} \mathcal{N}(\mathbf{v}^m|\tilde{\mathbf{v}}^m, \mathbf{G}^m) \mathcal{N}(\mathbf{b}^m|K_{\mathbf{XW}} K_{\mathbf{WW}}^{-1} \mathbf{v}^m, \boldsymbol{\Sigma}_{\mathbf{B}}) d\mathbf{V} \\ &= \prod_{m=1}^{m_1} \mathcal{N}(\mathbf{b}^m | \underbrace{K_{\mathbf{XW}} K_{\mathbf{WW}}^{-1} \tilde{\mathbf{v}}^m}_{\tilde{\mathbf{b}}^m}, \underbrace{\boldsymbol{\Sigma}_{\mathbf{B}} + K_{\mathbf{XW}} K_{\mathbf{WW}}^{-1} \mathbf{G}^m K_{\mathbf{WW}}^{-1} K_{\mathbf{WX}}}_{\tilde{\boldsymbol{\Sigma}}_{\mathbf{B}}^m}) \end{aligned}$$

Therefore, we can obtained a closed form integration for the last term in (16) as follows:

$$\begin{aligned}
&\int q(\mathbf{V})q(\mathbf{D})p(\mathbf{B}|\mathbf{V},\mathbf{W},\mathbf{X})\log p(\mathbf{D}|\mathbf{B})d(\mathbf{D},\mathbf{B},\mathbf{V}) \\
&= \int \left(\int q(\mathbf{V})p(\mathbf{B}|\mathbf{V},\mathbf{W},\mathbf{X})d\mathbf{V} \right) q(\mathbf{D})\log p(\mathbf{D}|\mathbf{B})d(\mathbf{D},\mathbf{B}) \\
&= \int \prod_{m=1}^{m_1} \mathcal{N}(\mathbf{b}^m|\tilde{\mathbf{b}}^m, \tilde{\mathbf{\Sigma}}_{\mathbf{B}}^m) \log \prod_{m=1}^{m_1} \mathcal{N}(\mathbf{d}^m = \tilde{\mathbf{d}}^m|\mathbf{b}^m, \lambda^{-1}\mathbf{I})d\mathbf{B} \\
&= \int \prod_{m=1}^{m_1} \mathcal{N}(\mathbf{b}^m|\tilde{\mathbf{b}}^m, \tilde{\mathbf{\Sigma}}_{\mathbf{B}}^m) \log \prod_{m=1}^{m_1} \left((2\pi)^{-n/2}|\lambda^{-1}\mathbf{I}|^{-1/2} e^{-\frac{\lambda}{2}(\tilde{\mathbf{d}}^m-\mathbf{b}^m)^\top(\tilde{\mathbf{d}}^m-\mathbf{b}^m)} \right) d\mathbf{B} \\
&= \int \prod_{m=1}^{m_1} \mathcal{N}(\mathbf{b}^m|\tilde{\mathbf{b}}^m, \tilde{\mathbf{\Sigma}}_{\mathbf{B}}^m) \left(-\frac{nm_1}{2}\log(2\pi\lambda^{-1}) - \frac{\lambda}{2}\sum_{m=1}^{m_1} (\tilde{\mathbf{d}}^m - \mathbf{b}^m)^\top(\tilde{\mathbf{d}}^m - \mathbf{b}^m) \right) d\mathbf{B} \\
&= -\frac{nm_1}{2}\log(2\pi\lambda^{-1}) - \frac{\lambda}{2}\sum_{m=1}^{m_1} \left(Tr(\tilde{\mathbf{\Sigma}}_{\mathbf{B}}^m) + (\tilde{\mathbf{d}}^m - \tilde{\mathbf{b}}^m)^\top(\tilde{\mathbf{d}}^m - \tilde{\mathbf{b}}^m) \right)
\end{aligned}$$

where $\tilde{\mathbf{\Sigma}}_B^m = \mathbf{\Sigma}_B + K_{\mathbf{XW}}K_{\mathbf{WW}}^{-1}\mathbf{G}^m K_{\mathbf{WW}}^{-1}K_{\mathbf{WX}}$, $\tilde{\mathbf{b}}^m = K_{\mathbf{XW}}K_{\mathbf{WW}}^{-1}\tilde{\mathbf{v}}^m$, and $\tilde{\mathbf{d}}^m = K_{\mathbf{XW}}K_{\mathbf{WW}}^{-1}\tilde{\mathbf{v}}^m$, according to the variational distribution Q.

We now express the variational lower bound of the log likelihood as follow:

$$\mathcal{L} = \mathcal{L}_M + \mathcal{L}_G - \mathcal{L}_{KL} + \mathcal{L}_B - \frac{nm_1}{2}\log(2\pi\lambda^{-1}) \tag{18}$$

where

$$\mathcal{L}_M = \log p(\mathcal{M}|K_{\tilde{\mathbf{D}}\mathbf{Z}}K_{\mathbf{ZZ}}^{-1}\tilde{\boldsymbol{f}}) \tag{19}$$

$$\mathcal{L}_G = \log p(\boldsymbol{f} = \tilde{\boldsymbol{f}}|\mathbf{Z}) = \log\mathcal{N}(\boldsymbol{f} = \tilde{\boldsymbol{f}}|0, K_{\mathbf{ZZ}}) \tag{20}$$

$$= -\frac{1}{2}\tilde{\boldsymbol{f}}^\top K_{\mathbf{ZZ}}^{-1}\tilde{\boldsymbol{f}} - \frac{n_{inducing}}{2}\log(2\pi) - \frac{1}{2}\log|K_{\mathbf{ZZ}}| \tag{21}$$

$$\mathcal{L}_{KL} = KL(q(\mathbf{V})||p(\mathbf{V}|\mathbf{W})) = \sum_{m=1}^{m_1} KL(\mathcal{N}(\mathbf{v}^m|\tilde{\mathbf{v}}^m, \mathbf{G}^m)||\mathcal{N}(\mathbf{v}^m|0, K_{\mathbf{WW}})) \tag{22}$$

$$= \sum_{m=1}^{m_1} \frac{1}{2}\left(Tr(K_{\mathbf{WW}}^{-1}(\mathbf{G}^m + \tilde{\mathbf{v}}^m\tilde{\mathbf{v}}^{m\top}) - n_{inducing} + \log\left(\frac{|K_{\mathbf{WW}}|}{|\mathbf{G}^m|}\right)\right) \tag{23}$$

$$\mathcal{L}_B = -\frac{\lambda}{2}\sum_{m=1}^{m_1} Tr(\mathbf{\Sigma}_{\mathbf{B}} + K_{\mathbf{XW}}K_{\mathbf{WW}}^{-1}\mathbf{G}^m K_{\mathbf{WW}}^{-1}K_{\mathbf{WX}}) \tag{24}$$

which is also described in the main paper. The learning of the model involves optimizing over the variational parameters, including $\tilde{\boldsymbol{f}}, \tilde{\mathbf{v}}^m, \mathbf{G}^m$, inducing inputs $\mathbf{Z}$, as well as hyperparameters for the kernel functions, which is performed through backpropagation based on the gradients of the variational lower bound (18) with respect to these parameters.

3 Optimizing the Variational Distribution $q(\mathbf{V})$

As can be seen, the variational lower bound (18) depends on the parameters of the variational distribution $q(\mathbf{V}) = \prod_{m=1}^{m_1} \mathcal{N}(\mathbf{v}^m|\tilde{\mathbf{v}}^m, \mathbf{G}^m)$, which can be optimized to improve the lower bound further. For the last term in (16), we have

$$
\begin{aligned}
&\int q(\mathbf{V})q(\mathbf{D})p(\mathbf{B}|\mathbf{V},\mathbf{W},\mathbf{X})\log\frac{p(\mathbf{D}|\mathbf{B})p(\mathbf{V}|\mathbf{W})}{q(\mathbf{V})}d(\mathbf{D},\mathbf{B},\mathbf{V})\\
&=\int q(\mathbf{V})\left(\int q(\mathbf{D})p(\mathbf{B}|\mathbf{V},\mathbf{W},\mathbf{X})\log\frac{p(\mathbf{D}|\mathbf{B})p(\mathbf{V}|\mathbf{W})}{q(\mathbf{V})}d(\mathbf{D},\mathbf{B})\right)d\mathbf{V}\\
&=\int q(\mathbf{V})\left(\int p(\mathbf{B}|\mathbf{V},\mathbf{W},\mathbf{X})\log\frac{p(\mathbf{D}=\tilde{\mathbf{D}}|\mathbf{B})p(\mathbf{V}|\mathbf{W})}{q(\mathbf{V})}d\mathbf{B}\right)d\mathbf{B}\\
&=\int q(\mathbf{V})\log\frac{e^{\langle\log p(\mathbf{D}=\tilde{\mathbf{D}}|\mathbf{B})\rangle_{p(\mathbf{B}|\mathbf{V},\mathbf{W},\mathbf{X})}}p(\mathbf{V}|\mathbf{W})}{q(\mathbf{V})}d\mathbf{V}
\end{aligned}
$$

where we have $\tilde{\mathbf{D}}=\begin{bmatrix}\tilde{\mathbf{d}}^1 & \cdots & \tilde{\mathbf{d}}^{m_1}\end{bmatrix}$, with $\tilde{\mathbf{d}}^m=K_{\mathbf{XW}}K_{\mathbf{WW}}^{-1}\tilde{\mathbf{e}}^m$, and $\tilde{\mathbf{e}}^m$ for $m=1,...,m_1$ are variational parameters to optimize.

To maximize the above quantity, we can reverse the Jensen's inequality to obtin the condition that:

$$\log q(\mathbf{V})=\mathit{const}+\langle\log p(\mathbf{D}=\tilde{\mathbf{D}}|\mathbf{B})\rangle_{p(\mathbf{B}|\mathbf{V},\mathbf{W},\mathbf{X})}+\log p(\mathbf{V}|\mathbf{W})$$

Now for the term $\langle\log p(\mathbf{D}=\tilde{\mathbf{D}}|\mathbf{B})\rangle_{p(\mathbf{B}|\mathbf{V},\mathbf{W},\mathbf{X})}$, we have:

$$
\begin{aligned}
&\langle\log p(\mathbf{D}=\tilde{\mathbf{D}}|B)\rangle_{p(\mathbf{B}|\mathbf{V},\mathbf{W},\mathbf{X})}=\sum_{m=1}^{m_1}\langle\log\mathcal{N}(\mathbf{d}^m=\tilde{\mathbf{d}}^m|\mathbf{b}^m,\lambda^{-1}I)\rangle_{p(\mathbf{B}|\mathbf{V},\mathbf{W},\mathbf{X})}\\
&=\mathit{const}+\sum_{m=1}^{m_1}\left\langle-\frac{\lambda}{2}Tr\left(\tilde{\mathbf{d}}^m\tilde{\mathbf{d}}^{m\top}+\mathbf{b}^m\mathbf{b}^{m\top}-2\tilde{\mathbf{d}}^m\mathbf{b}^{m\top}\right)\right\rangle_{\mathcal{N}(\mathbf{b}^m|K_{\mathbf{XW}}K_{\mathbf{WW}}^{-1}\mathbf{v}^m,\mathbf{\Sigma_B})}\\
&=\mathit{const}+\sum_{m=1}^{m_1}\left(-\frac{\lambda}{2}Tr\left(\tilde{\mathbf{d}}^m\tilde{\mathbf{d}}^{m\top}+\mathbf{\Sigma_B}+\mathbf{v}^{m\top}K_{\mathbf{WW}}^{-1}K_{\mathbf{WX}}K_{\mathbf{XW}}K_{\mathbf{WW}}^{-1}\mathbf{v}^m-2\mathbf{v}^{m\top}K_{\mathbf{WW}}^{-1}K_{\mathbf{WX}}\tilde{\mathbf{d}}^m\right)\right)
\end{aligned}
$$

Therefore, we have:

$$\log q(\mathbf{v}^m)=\mathit{const}-\frac{1}{2}\left(\lambda\mathbf{v}^{m\top}K_{\mathbf{WW}}^{-1}K_{\mathbf{WX}}K_{\mathbf{XW}}K_{\mathbf{WW}}^{-1}\mathbf{v}^m-2\lambda\mathbf{v}^{m\top}K_{\mathbf{WW}}^{-1}K_{\mathbf{WX}}\tilde{\mathbf{d}}^m+\mathbf{v}^{m\top}K_{\mathbf{WW}}^{-1}\mathbf{v}^m\right)$$

Therefore by completing the squares we have $q(\mathbf{v}^m)=\mathcal{N}(\mathbf{v}^m|\tilde{\mathbf{v}}_*^m,\mathbf{\Sigma}_{\mathbf{v}*}^m)$:

$$
\begin{aligned}
\mathbf{\Sigma}_{\mathbf{v}*}^m&=(K_{\mathbf{WW}}^{-1}+\lambda K_{\mathbf{WW}}^{-1}K_{\mathbf{WX}}K_{\mathbf{XW}}K_{\mathbf{WW}}^{-1})^{-1}\\
&=\lambda^{-1}K_{\mathbf{WW}}(\lambda^{-1}K_{\mathbf{WW}}+K_{\mathbf{WX}}K_{\mathbf{XW}})^{-1}K_{\mathbf{WW}}\\
\tilde{\mathbf{v}}_*^m&=\lambda\mathbf{\Sigma}_{\mathbf{v}*}^mK_{\mathbf{WW}}^{-1}K_{\mathbf{WX}}\tilde{\mathbf{d}}^m\\
&=K_{\mathbf{WW}}\underbrace{(\lambda^{-1}K_{\mathbf{WW}}+K_{\mathbf{WX}}K_{\mathbf{XW}})^{-1}}_{\mathbf{\Gamma}}K_{\mathbf{WX}}\tilde{\mathbf{d}}^m
\end{aligned}
$$

With the above optimized variational parameters for $q(\mathbf{v}^m)$, we first obtain:

$$
\begin{aligned}
&\int q(\mathbf{v}^m)\langle\log p(\mathbf{d}^m=\tilde{\mathbf{d}}^m|\mathbf{b}^m)\rangle_{p(\mathbf{b}^m|\mathbf{v}^m,\mathbf{W},\mathbf{X})}d\mathbf{v}^m=\\
&-\frac{n}{2}\log(2\pi\lambda^{-1})-\frac{\lambda}{2}Tr\left(\tilde{\mathbf{d}}^m\tilde{\mathbf{d}}^{m\top}+\mathbf{\Sigma_B}+K_{\mathbf{WW}}^{-1}K_{\mathbf{WX}}K_{\mathbf{XW}}K_{\mathbf{WW}}^{-1}(\mathbf{\Sigma}_{\mathbf{v}*}^m+\tilde{\mathbf{v}}_*^m\tilde{\mathbf{v}}_*^{m\top})-2\tilde{\mathbf{v}}_*^{m\top}K_{\mathbf{WW}}^{-1}K_{\mathbf{WX}}\tilde{\mathbf{d}}^m\right)
\end{aligned}
$$

Next, we calculate $\int q(\mathbf{v}^m)\log p(\mathbf{v}^m|\mathbf{W})d\mathbf{v}^m$:

$$\int q(\mathbf{v}^m)\log p(\mathbf{v}^m|\mathbf{W})=-\frac{n}{2}\log(2\pi)-\frac{1}{2}\log|K_{\mathbf{WW}}|-\frac{1}{2}Tr(K_{\mathbf{WW}}^{-1}(\mathbf{\Sigma}_{\mathbf{v}*}^m+\tilde{\mathbf{v}}_*^m\tilde{\mathbf{v}}_*^{m\top}))$$

Finally we have:

$$H(q(\mathbf{v}^m))=q(\mathbf{v}^m)\log\frac{1}{q(\mathbf{v}^m)}=\frac{n}{2}\log(2\pi)+\frac{1}{2}\log|\mathbf{\Sigma}_{\mathbf{v}*}^m| \tag{25}$$

Summarizing, we have:

$$
\begin{aligned}
&\int q(\mathbf{V})q(\mathbf{D})p(\mathbf{B}|\mathbf{V},\mathbf{W},\mathbf{X})\log\frac{p(\mathbf{D}|\mathbf{B})p(\mathbf{V}|\mathbf{W})}{q(\mathbf{V})}d(\mathbf{D},\mathbf{B},\mathbf{V})\\
&\le \sum_{m=1}^{m_1}\Bigg[-\frac{n}{2}\log(2\pi\lambda^{-1})-\frac{1}{2}\log|K_{\mathbf{WW}}|-\frac{1}{2}Tr(K_{\mathbf{WW}}^{-1}(\mathbf{\Sigma}_{\mathbf{v}*}^{m}+\tilde{\mathbf{v}}_*^{m}\tilde{\mathbf{v}}_*^{m\top}))+\frac{1}{2}\log|\mathbf{\Sigma}_{\mathbf{v}*}^{m}|\\
&-\frac{\lambda}{2}Tr\Big(\tilde{\mathbf{d}}^{m}\tilde{\mathbf{d}}^{m\top}+\mathbf{\Sigma_B}+K_{\mathbf{WW}}^{-1}K_{\mathbf{WX}}K_{\mathbf{XW}}K_{\mathbf{WW}}^{-1}(\mathbf{\Sigma}_{\mathbf{v}*}^{m}+\tilde{\mathbf{v}}_*^{m}\tilde{\mathbf{v}}_*^{m\top})-2\tilde{\mathbf{v}}_*^{m\top}K_{\mathbf{WW}}^{-1}K_{\mathbf{WX}}\tilde{\mathbf{d}}^{m}\Big)\Bigg]
\end{aligned}
$$

We now express the variational lower bound of the log likelihood as follow:

$$\mathcal{L}=\mathcal{L}_M+\mathcal{L}_G+\mathcal{L}_{DBV} \tag{26}$$

where

$$
\begin{aligned}
&\mathcal{L}_M=\log p(\mathcal{M}|K_{\tilde{\mathbf{D}}\mathbf{Z}}K_{\mathbf{ZZ}}^{-1}\tilde{\boldsymbol{f}}) && (27)\\
&\mathcal{L}_G=\log p(u=\tilde{u}|Z)=\log\mathcal{N}(u=\tilde{u}|0,K_{ZZ}) && (28)\\
&=-\frac{1}{2}\tilde{u}^{\top}K_{ZZ}^{-1}\tilde{u}-\frac{K}{2}\log(2\pi)-\frac{1}{2}\log|K_{ZZ}| && (29)\\
&\mathcal{L}_{DBV}=\sum_{m=1}^{m_1}\Bigg[-\frac{n}{2}\log(2\pi\lambda^{-1})-\frac{1}{2}\log|K_{\mathbf{WW}}|-\frac{1}{2}Tr(K_{\mathbf{WW}}^{-1}(\mathbf{\Sigma}_{\mathbf{v}*}^{m}+\tilde{\mathbf{v}}_*^{m}\tilde{\mathbf{v}}_*^{m\top}))+\frac{1}{2}\log|\mathbf{\Sigma}_{\mathbf{v}*}^{m}|\\
&-\frac{\lambda}{2}Tr\Big(\tilde{\mathbf{d}}^{m}\tilde{\mathbf{d}}^{m\top}+\mathbf{\Sigma_B}+K_{\mathbf{WW}}^{-1}K_{\mathbf{WX}}K_{\mathbf{XW}}K_{\mathbf{WW}}^{-1}(\mathbf{\Sigma}_{\mathbf{v}*}^{m}+\tilde{\mathbf{v}}_*^{m}\tilde{\mathbf{v}}_*^{m\top})-2\tilde{\mathbf{v}}_*^{m\top}K_{\mathbf{WW}}^{-1}K_{\mathbf{WX}}\tilde{\mathbf{d}}^{m}\Big)\Bigg] && (30)
\end{aligned}
$$

where $\tilde{\mathbf{d}}^m=K_{\mathbf{XW}}K_{\mathbf{WW}}^{-1}\tilde{\mathbf{e}}^m$, $\mathbf{\Gamma}=(\lambda^{-1}K_{\mathbf{WW}}+K_{\mathbf{WX}}K_{\mathbf{XW}})^{-1}$,$\Sigma_{\mathbf{v}*}^m=\lambda^{-1}K_{\mathbf{WW}}\mathbf{\Gamma}K_{\mathbf{WW}}$.

The parameters we need to learn in this case include the variational parameters $\tilde{\boldsymbol{f}}$, and $\tilde{\mathbf{e}}^m$ for $m=1,...,m_1$, inducing inputs $\mathbf{Z}$, as well as hyperparameters for kernel functions.

4 Parameters Learning by Derivatives

In this section, we will obtain the derivatives of the marginal log likelihood $\mathcal{L}$ in (26) with respect to the variational parameters $\tilde{\boldsymbol{f}}$, $\tilde{\mathbf{e}}^m$ and inducing inputs $\mathbf{Z}$. The derivative of the reinforcement learning term, $p(\mathcal{M}|\boldsymbol{r})$ in (7), with respect to the reward vectors $\boldsymbol{r}$, is given by:

$$\frac{\partial}{\partial\boldsymbol{r}}\log p(\mathcal{M}|\boldsymbol{r})=\sum_i\sum_t\left(\frac{\partial}{\partial\boldsymbol{r}}\boldsymbol{r}_{s_{i,t},a_{i,t}}-\frac{\partial}{\partial\boldsymbol{r}}V^{\boldsymbol{r}}_{s_{i,t}}+\sum_{s'}\gamma\mathcal{T}^{s_{i,t},a_{i,t}}_{s'}\frac{\partial}{\partial\boldsymbol{r}}V^{\boldsymbol{r}}_{s'}\right) \tag{31}$$

The first term, $\sum_i\sum_t\frac{\partial}{\partial\boldsymbol{r}}\boldsymbol{r}_{s_{i,t},a_{i,t}}$, is simply a vector that counts the number of state-action pairs in the demonstrations $\hat{\mu}$, whose entry corresponding to (s,a) is given by: $\hat{\mu}_{s,a}=\sum_i\sum_t 1_{s_{i,t}=s\wedge a_{i,t}=a}$. The second term involves the derivative of the value function at state s with respect to rewards, as indicated in [4], equal to the expected visitation count of each state-action pair when starting from state s and following the optimal stochastic policy, i.e., $\frac{\partial}{\partial\boldsymbol{r}}V^{\boldsymbol{r}}_s=E[\mu|s]$, where μ is a vector with each entry $\mu_{s,a}$ corresponding to the expected visitation count for (s,a). Therefore, (31) can be written as:

$$
\begin{aligned}
\frac{\partial}{\partial\boldsymbol{r}}\log p(\mathcal{M}|\boldsymbol{r})&=\hat{\mu}-\sum_i\sum_t E[\mu|s_{i,t}]+\sum_i\sum_t\sum_{s'}\gamma\mathcal{T}^{s_{i,t},a_{i,t}}_{s'}E[\mu|s_{i,t}]\\
&=\hat{\mu}-\sum_s\hat{\nu}_sE[\mu|s]
\end{aligned}
$$

where $\hat{\nu}_s=\sum_a\hat{\mu}_{s,a}-\sum_i\sum_t\gamma\mathcal{T}^{s_{i,t},a_{i,t}}_{s'}$. The term $\sum_s\hat{\nu}_sE[\mu|s]$ can be computed efficiently by a simple iterative algorithm described in [4], which we do not recount here. Note that the above derivation follows from [1].

For the variational parameters $\tilde{\boldsymbol{f}}$, we need to consider only two terms that involve it, i.e., $\mathcal{L}_{\mathcal{M}}, \mathcal{L}_G$:

$$\frac{\partial \mathcal{L}_M}{\partial \tilde{\boldsymbol{f}}} = \frac{\partial \boldsymbol{r}}{\partial \tilde{\boldsymbol{f}}} \frac{\partial \mathcal{L}_{\mathcal{M}}}{\partial \boldsymbol{r}} = K_{\tilde{\mathbf{D}}\mathbf{Z}} K_{\mathbf{ZZ}}^{-1} \frac{\partial \log p(\mathcal{M}|\boldsymbol{r})}{\partial \boldsymbol{r}}$$
$$\frac{\partial \mathcal{L}_G}{\partial \tilde{\boldsymbol{f}}} = -K_{\mathbf{ZZ}}^{-1} \tilde{\boldsymbol{f}}$$

where $\boldsymbol{r} = K_{\tilde{\mathbf{D}}\mathbf{Z}} K_{\mathbf{ZZ}}^{-1} \tilde{\boldsymbol{f}}$ is the reward vector that we use for reinforcement learning.

For the variational parameters $\tilde{\mathbf{e}}^m$, let $\tilde{\mathbf{D}} = \left[K_{\mathbf{XW}} K_{\mathbf{WW}}^{-1} \tilde{\mathbf{e}}^1, ..., K_{\mathbf{XW}} K_{\mathbf{WW}}^{-1} \tilde{\mathbf{e}}^{m_1}\right] \in \mathbb{R}^{n \times m_1}$, and $\mathbf{E} = [\tilde{\mathbf{e}}^1, ..., \tilde{\mathbf{e}}^{m_1}] \in \mathbb{R}^{K \times m_1}$:

$$\frac{\partial \mathcal{L}_{\mathcal{M}}}{\partial \mathbf{E}} = \frac{\partial \tilde{\mathbf{D}}}{\partial \mathbf{E}} \frac{\partial K_{\tilde{\mathbf{D}}\mathbf{Z}}}{\partial \tilde{\mathbf{D}}} \frac{\partial \boldsymbol{r}}{\partial K_{\tilde{\mathbf{D}}\mathbf{Z}}} \frac{\partial \mathcal{L}_{\mathcal{M}}}{\partial \boldsymbol{r}}$$

In addition, by applying matrix derivatives,

$$\frac{\partial \mathcal{L}_{DBV}}{\partial \mathbf{e}^m} = -\frac{\lambda}{2} \Big(2K_{\mathbf{WW}}^{-1} K_{\mathbf{WX}} K_{\mathbf{XW}} K_{\mathbf{WW}}^{-1} + 2K_{\mathbf{WW}}^{-1} K_{\mathbf{WX}} K_{\mathbf{XW}} \mathbf{\Gamma} K_{\mathbf{WX}} K_{\mathbf{XW}} \Gamma K_{\mathbf{WX}} K_{\mathbf{XW}} K_{\mathbf{WW}}^{-1}$$
$$-4K_{\mathbf{WW}}^{-1} K_{\mathbf{WX}} K_{\mathbf{XW}} \mathbf{\Gamma} K_{\mathbf{WX}} K_{\mathbf{XW}} K_{\mathbf{WW}}^{-1} \Big) \mathbf{e}^m - K_{\mathbf{WW}}^{-1} K_{\mathbf{WX}} K_{\mathbf{XW}} \mathbf{\Gamma} K_{\mathbf{WW}} \mathbf{\Gamma} K_{\mathbf{WX}} K_{\mathbf{XW}} K_{\mathbf{WW}}^{-1} \mathbf{e}^m$$

The gradients are provided to minFunc [3], which calls a quasi-Newton strategy, where limited-memory BFGS updates with Shanno-Phua scaling are used in computing the step direction, and a bracketing line-search for a point satisfying the strong Wolfe conditions is used to compute the step direction.

References

[1] S. Levine, Z. Popovic, and V. Koltun. Nonlinear inverse reinforcement learning with Gaussian processes. In *Advances in Neural Information Processing Systems*, pages 19–27, 2011.

[2] J. Quiñonero-Candela and C. E. Rasmussen. A unifying view of sparse approximate Gaussian process regression. *The Journal of Machine Learning Research*, 6:1939–1959, 2005.

[3] M. Schmidt. minFunc: unconstrained differentiable multivariate optimization in matlab. *URL http://www. di. ens. fr/mschmidt/Software/minFunc. html*, 2012.

[4] B. D. Ziebart, J. A. Bagnell, and A. K. Dey. Modeling interaction via the principle of maximum causal entropy. In *International Conference on Machine Learning (ICML)*, pages 1247–1254, 2010.