University of Glasgow | School of Computing Science

Variational Inference for Inverse Reinforcement Learning with Gaussian Processes

Paulius Dilkas (2146879)

19th March 2019

ABSTRACT

We prove the validity of performing variational inference on the Gaussian process-based inverse reinforcement learning model that preserves and approximates the full joint probability distribution of rewards across states of the Markov decision process.

1. INTRODUCTION

Inverse reinforcement learning (IRL)—a problem proposed by Russell in 1998 [35]—asks us to find a reward function for a Markov decision process (MDP) that best explains a set of given demonstrations. IRL is important because reward functions can be hard to define manually [1, 2], and rewards are not entirely specific to a given environment, allowing one to reuse the same reward structure in previously unseen environments [2, 14, 19]. Moreover, IRL has seen a wide array of applications in autonomous vehicle control [15, 16] and learning to predict another agent's behaviour [5, 38, 42, 43, 44]. Most approaches in the literature make a convenient yet unjustified assumption that the reward function can be expressed as a linear combination of features. One proven way to abandon this assumption is by representing the reward function as a Gaussian process (GP) [14, 19, 28]. The original approach used maximum likelihood estimation [19], whereas we use variational inference (VI) instead, which learns approximate posterior probability distributions instead of point estimates. This approach can prove useful in three major ways:

1. Modelling full posterior distributions for various parameters can result in more precise reward predictions, as the model simply holds more information.
2. Having variance estimates for rewards can direct our choice in what data should be collected next.
3. An approximate Bayesian treatment of many parameters in the model guards against overfitting [14].

2. THE PROBLEM

Definition 2.1 (MDP). *A* Markov decision process *is a set $\mathcal{M} = \{\mathcal{S}, \mathcal{A}, \mathcal{T}, \gamma, \mathbf{r}\}$, where $\mathcal{S}$ and $\mathcal{A}$ are sets of states and actions, respectively; $\mathcal{T} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is a function defined so that $\mathcal{T}(s, a, s')$ is the probability of moving to state s' after taking action a in state s; $\gamma \in [0, 1)$ is the discount factor; and $\mathbf{r} \in \mathbb{R}^{|\mathcal{S}|}$ is the reward vector*[1].

[1]Depending on the situation, we will sometimes represent rewards as a function $r : \mathcal{S} \to \mathbb{R}$.

Definition 2.2 (IRL). *Given an MDP without rewards $\mathcal{M} \setminus \{\mathbf{r}\}$, an $|\mathcal{S}| \times d$ feature matrix $\mathbf{X}$ (where d is the number of features), and a set of expert demonstrations $\mathcal{D} = \{\zeta_i\}_{i=1}^{N}$, where each demonstration $\zeta_i = \{(s_{i,t}, a_{i,t})\}_{t=1}^{T}$ is a multiset of state-action pairs representing optimal actions executed by an expert, find the reward function that maximises the probability of observing the demonstrations, i.e.,*

$$\arg\max_{\mathbf{r}} p(\mathcal{D} \mid \mathbf{r}).$$

The optimal *policy* $\pi : \mathcal{S} \to \mathcal{A}$ (i.e., a choice of actions for each state that maximises reward over time) is usually constructed by defining a *value (utility) function* $V_{\mathbf{r}} : \mathcal{S} \to \mathbb{R}$ that measures how good a state is based on the reward $\mathbf{r}$ as well as the structure of the MDP. One can then find $V_{\mathbf{r}}$ by applying the Bellman backup operator until convergence to every $s \in \mathcal{S}$ (the technique is known as *value iteration*) [36]:

$$V_{\mathbf{r}}(s) \coloneqq r(s) + \gamma \max_{a \in \mathcal{A}} \sum_{s' \in \mathcal{S}} \mathcal{T}(s, a, s') V_{\mathbf{r}}(s').$$

However, we follow previous work on GP IRL [19, 14], and use a *linearly solvable* (or *maximum causal entropy*) MDP with stochastic policy that define probability distributions over actions (instead of suggesting a single action for each state) [42]. This type of MDP can be solved by applying the 'soft' version of the operator [19, 20]:

$$V_{\mathbf{r}}(s) \coloneqq \log \sum_{a \in \mathcal{A}} \exp\left(r(s) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{T}(s, a, s') V_{\mathbf{r}}(s') \right). \quad (1)$$

With this model, we can express the likelihood as [14, 19]

$$\begin{aligned} p(\mathcal{D} \mid \mathbf{r}) &= \prod_{i=1}^{N} \prod_{t=1}^{T} p(a_{i,t} \mid s_{i,t}) \\ &= \exp\left(\sum_{i=1}^{N} \sum_{t=1}^{T} Q_{\mathbf{r}}(s_{i,t}, a_{i,t}) - V_{\mathbf{r}}(s_{i,t}) \right), \end{aligned} \quad (2)$$

where

$$Q_{\mathbf{r}}(s, a) = r(s) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{T}(s, a, s') V_{\mathbf{r}}(s').$$

However, a reward function learned by maximising this likelihood is not transferable to new situations [14, 19]. One needs to model the reward structure in a way that would allow reward predictions for previously unseen states.

One way to model rewards without assumptions of linearity is with a *Gaussian process* (GP). A GP is a collection of random variables, any finite combination of which has a joint Gaussian distribution [32]. We write $r \sim \mathcal{GP}(0, k)$ to

say that r is a GP with mean 0 and covariance function k. *Covariance functions* (also known as *kernels*) take two state feature vectors as input and quantify how similar the two states are, in a sense that we would expect them to have similar rewards.

As training a GP with n data points has a time complexity of $\mathcal{O}(n^3)$ [32], numerous approximation methods have been suggested, many of which select a subset of data called *inducing points* and focus most of the training effort on them [21]. Let $\mathbf{X_u}$ be the matrix of features at inducing states, $\mathbf{u}$ the rewards at those states. Then the full joint probability distribution can be factorised as

$$p(\mathcal{D}, \mathbf{u}, \mathbf{r}) = p(\mathbf{u}) \times p(\mathbf{r} \mid \mathbf{u}) \times p(\mathcal{D} \mid \mathbf{r}), \tag{3}$$

where

$$\begin{aligned} p(\mathbf{u}) &= \mathcal{N}(\mathbf{u}; \mathbf{0}, \mathbf{K_{u,u}}) \\ &= \frac{1}{(2\pi)^{m/2}|\mathbf{K_{u,u}}|^{1/2}} \exp\left(-\frac{1}{2}\mathbf{u}^\intercal \mathbf{K_{u,u}^{-1}}\mathbf{u}\right) \\ &= \exp\left(-\frac{1}{2}\mathbf{u}^\intercal \mathbf{K_{u,u}^{-1}}\mathbf{u} - \frac{1}{2}\log|\mathbf{K_{u,u}}| - \frac{m}{2}\log 2\pi\right) \end{aligned}$$

is the GP prior [32], and $m \in \mathbb{N}$ is the number of inducing points. The GP posterior is then a multivariate Gaussian [19]

$$p(\mathbf{r} \mid \mathbf{u}) = \mathcal{N}(\mathbf{r}; \mathbf{K}_{\mathbf{r},\mathbf{u}}^\intercal \mathbf{K_{u,u}^{-1}}\mathbf{u}, \mathbf{K_{r,r}} - \mathbf{K}_{\mathbf{r},\mathbf{u}}^\intercal \mathbf{K_{u,u}^{-1}}\mathbf{K_{r,u}}), \tag{4}$$

and $p(\mathcal{D} \mid \mathbf{r})$ is as in (2). The matrices such as $\mathbf{K_{r,u}}$ are called *covariance matrices* and are defined as $[\mathbf{K_{r,u}}]_{i,j} = k(\mathbf{x}_{\mathbf{r},i}, \mathbf{x}_{\mathbf{u},j})$, where $\mathbf{x}_{\mathbf{r},i}$ and $\mathbf{x}_{\mathbf{u},j}$ denote feature vectors for the ith state in $\mathcal{S}$ and the jth state in $\mathbf{X_u}$, respectively [14].

Given this model, data $\mathcal{D}$, and inducing feature matrix $\mathbf{X_u}$, our goal is then to find optimal values of parameters $\boldsymbol{\lambda}$, inducing rewards $\mathbf{u}$, and the rewards for all relevant states $\mathbf{r}$. While the previous paper to consider this IRL model computed maximum likelihood (ML) estimates for $\boldsymbol{\lambda}$ and $\mathbf{u}$, and made an assumption that $\mathbf{r}$ in (4) has zero variance [19], we aim to avoid this assumption and use VI to approximate the full posterior distribution $p(\mathbf{u}, \mathbf{r} \mid \mathcal{D})$. *Variational inference* is an approximation technique for probability densities [4]. Let $q(\mathbf{u}, \mathbf{r})$ be our approximating family of probability distributions for $p(\mathbf{u}, \mathbf{r} \mid \mathcal{D})$. Then the job of VI is to optimise the parameters of the approximating distribution in order to minimise the *Kullback-Leibler* (KL) divergence between the original probability distribution and our approximation. KL divergence (asymmetrically) measures how different the two distributions are, and can be defined as [4]

$$\begin{aligned} D_{\mathrm{KL}}(q(\mathbf{u}, \mathbf{r}) \parallel p(\mathbf{u}, \mathbf{r} \mid \mathcal{D})) &= \mathbb{E}[\log q(\mathbf{u}, \mathbf{r}) - \log p(\mathbf{u}, \mathbf{r} \mid \mathcal{D})] \\ &= \mathbb{E}[\log q(\mathbf{u}, \mathbf{r}) - \log p(\mathcal{D}, \mathbf{u}, \mathbf{r})] \\ &\quad + \mathbb{E}[\log p(\mathcal{D})]. \end{aligned}$$

The last term is both hard to compute and constant w.r.t. $q(\mathbf{u}, \mathbf{r})$ [4], so we can remove it from our optimisation objective. The negation of what remains is often called the *evidence lower bound* (ELBO) and is defined as[2] [3, 4]

$$\begin{aligned} \mathcal{L} &= \mathbb{E}\left[\log \frac{p(\mathcal{D}, \mathbf{u}, \mathbf{r})}{q(\mathbf{u}, \mathbf{r})}\right] \\ &= \iiint \log \frac{p(\mathcal{D}, \mathbf{u}, \mathbf{r})}{q(\mathbf{u}, \mathbf{r})} q(\mathbf{u}, \mathbf{r})\, d\mathbf{r}\, d\mathbf{u}. \end{aligned} \tag{5}$$

[2]Throughout the proposal, all integrals should be interpreted as definite integrals over the entire sample space.

By considering full probability distributions instead of point estimates—as long as the approximations are able to capture important features of the posterior—our predictions are likely to be more accurate and rely on fewer assumptions. Moreover, we hope to make use of various recent advancements in VI for both time complexity and approximation distribution fit, making the resulting algorithm competitive both in terms of running time and model fit.

3. BACKGROUND

Here we introduce a few definitions and results from linear algebra, numerical analysis, and measure theory that will be used later in the paper. Namely, we will use several different vector and matrix norms, consider how an inverse of a matrix changes with a small perturbation, and use Lebesgue's dominated convergence theorem in order to justify the validity of our approach.

DEFINITION 3.1 (NORMS). *For any finite-dimensional vector* $\mathbf{x} = (x_1, \ldots, x_n)^\intercal$, *its* maximum norm *($\ell_\infty$-norm) is*

$$\|\mathbf{x}\|_\infty = \max_i |x_i|$$

whereas its ℓ_1-norm *is*

$$\|\mathbf{x}\|_1 = \sum_{i=1}^{n} |x_i|.$$

Let $\mathbf{A}$ *be a matrix. For any vector norm* $\|\cdot\|_p$, *we can also define its* induced norm *for matrices as*

$$\|\mathbf{A}\|_p = \sup_{\mathbf{x} \neq \mathbf{0}} \frac{\|\mathbf{A}\mathbf{x}\|_p}{\|\mathbf{x}\|_p}.$$

In particular, for $p = \infty$, *we have*

$$\|\mathbf{A}\|_\infty = \max_i \sum_j |A_{i,j}|.$$

LEMMA 3.2 (PERTURBATION LEMMA [18]). *Let* $\|\cdot\|$ *be any matrix norm, and let* $\mathbf{A}$ *and* $\mathbf{E}$ *be matrices such that* $\mathbf{A}$ *is invertible and* $\|\mathbf{A}^{-1}\|\|\mathbf{E}\| < 1$, *then* $\mathbf{A} + \mathbf{E}$ *is invertible, and*

$$\|(\mathbf{A} + \mathbf{E})^{-1}\| \leq \frac{\|\mathbf{A}^{-1}\|}{1 - \|\mathbf{A}^{-1}\|\|\mathbf{E}\|}.$$

THEOREM 3.3 (DOMINATED CONVERGENCE THEOREM [34]). *Let* $(X, \mathcal{M}, \mu)$ *be a measure space and* $\{f_n\}$ *a sequence of measurable functions on* X *for which* $\{f_n\} \to f$ *pointwise a.e. on* X *and the function* f *is measurable. Assume there is a non-negative function* g *that is integrable over* X *and dominates the sequence* $\{f_n\}$ *on* X *in the sense that*

$$|f_n| \leq g \text{ a.e. on } X \text{ for all } n.$$

Then f *is integrable over* X *and*

$$\lim_{n\to\infty} \int_X f_n\, d\mu = \int_X f\, d\mu.$$

4. THE MODEL

For any matrix $\mathbf{A}$, we will use either $A_{i,j}$ or $[\mathbf{A}]_{i,j}$ to denote the element of $\mathbf{A}$ in row i and column j. Moreover, we use $\mathrm{tr}(\mathbf{A})$ to denote its *trace* and $\mathrm{adj}(\mathbf{A})$ for its *adjugate* (or *classical adjoint*). For any vector $\mathbf{x}$, we write $\mathbb{R}_d[\mathbf{x}]$ to

denote a vector space of polynomials with degree at most d, where variables are elements of $\mathbf{x}$, and coefficients are in $\mathbb{R}$.

In this paper, all references to measurability are with respect to the Lebesgue measure. Similarly, whenever we consider the existence of an integral, we use the Lebesgue definition of integration.

We keep the covariance function the same as in the work by Levine et al. [19], which is a version of the automatic relevance detection kernel [19, 22]:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \lambda_0 \exp\Bigg(-\frac{1}{2}(\mathbf{x}_i - \mathbf{x}_j)^\intercal \mathbf{\Lambda}(\mathbf{x}_i - \mathbf{x}_j) - \mathbb{1}[i \neq j]\sigma^2 \operatorname{tr}(\mathbf{\Lambda}) \Bigg).$$

Here, λ_0 is the overall 'scale' factor for how similar or distant the states are, $\mathbf{\Lambda} = \operatorname{diag}(\lambda_1, \ldots, \lambda_d)$ is a diagonal matrix that determines the relevance of each feature (where d denotes the number of features), $\mathbb{1}$ is defined as

$$\mathbb{1}[b] = \begin{cases} 1 & \text{if } b \text{ is true} \\ 0 & \text{otherwise,} \end{cases}$$

and σ^2 is set to $10^{-2}/2$ (as the original paper noted that the value makes little difference to the performance of the algorithm [19]). We will write $\boldsymbol{\lambda} = (\lambda_0, \ldots, \lambda_d)^\intercal$ to refer to both λ_0 and $\mathbf{\Lambda}$ at the same time.

Ideally, we would like to model $\boldsymbol{\lambda}$ with an approximating distribution. However, due to how $p(\mathbf{u})$ has $\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}$ in its expression, and the ELBO is defined as an expectation, we are unable to show that the ELBO is well-defined. More generally, we pose the following problem, which is open to the best of our knowledge:

Open Problem 4.1. *Let $\mathbf{A}$ be a $n \times n$ matrix of coefficients, X be a random variable, and $\mathbf{M}$ be an $n \times n$ matrix such that $M_{i,j} = f(X, A_{i,j})$, where f is an arbitrary function. Under what circumstances does $\mathbb{E}[\mathbf{M}^{-1}]$ exist?*

While there are some obvious examples of when the required expected value exists (e.g., $f(X, A_{i,j}) = A_{i,j}X$ for an invertible $\mathbf{A}$ and many distributions of X), it would be particularly interesting to know whether the answer is 'always'. A proof of such a result would allow us to model $\boldsymbol{\lambda}$ instead of treating it as a variational parameter, and would thus guard against overfitting. For now, $\boldsymbol{\lambda}$ will have to be treated as a variational parameter.

It remains to decide on the model for $\mathbf{u}$ and $\mathbf{r}$. As is commonly done when applying VI to GPs, we set

$$q(\mathbf{u}, \mathbf{r}) = q(\mathbf{u})q(\mathbf{r} \mid \mathbf{u}), \tag{6}$$

where $q(\mathbf{r} \mid \mathbf{u}) = p(\mathbf{r} \mid \mathbf{u})$ and $q(\mathbf{u}) = \mathcal{N}(\mathbf{u}; \boldsymbol{\mu}, \mathbf{\Sigma})$ [7].

Ong et al. [25] have recently suggested that, in order to make variational approximation of a multivariate Gaussian more scalable, the covariance matrix should be decomposed as $\mathbf{\Sigma} = \mathbf{B}\mathbf{B}^\intercal + \mathbf{D}^2$, where $\mathbf{B}$ is a lower triangular $m \times p$ matrix with positive diagonal entries, and $\mathbf{D}$ is a diagonal matrix. Typically, we would set p so that $p \ll m$ to get an efficient approximation. However, as our goal is precision rather than scalability, we will set $p = m$ and $\mathbf{D} = \mathbf{O}_m$ in order to retain full covariance structure.

The resulting model is summarised in Figure 1. We rely on $p(\mathcal{D} \mid \mathbf{r})$ as the only link between data and the model. Since the expression for $q(\mathbf{r} \mid \mathbf{u})$ has both $\mathbf{u}$ and covariance

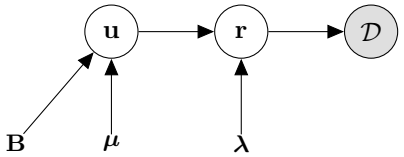

Figure 1: Our VI problem expressed as a (simplified) Bayesian network. The only observed variable (representing the demonstrations) is in a gray circle, modelled latent variables are in white circles, and the variational parameters are at the bottom.

matrices in it, $\mathbf{r}$ depends on both $\mathbf{u}$ and the parameters of the kernel, $\boldsymbol{\lambda}$. The two remaining dependencies stem from the fact that the approximating distribution for $\mathbf{u}$ is $\mathcal{N}(\boldsymbol{\lambda}, \mathbf{B}\mathbf{B}^\intercal)$.

As we want to restrict some parameters (namely, $\boldsymbol{\lambda}$ and the diagonal of $\mathbf{B}$) to positive values, we express them as exponentials and later adjust their derivatives accordingly. Specifically, we can set $\lambda_i = e^{\lambda_i'}$ and optimise λ_i' using the chain rule:

$$\frac{\partial \mathcal{L}}{\partial \lambda_i'} = e^{\lambda_i'} \frac{\partial \mathcal{L}}{\partial \lambda_i}.$$

This way, we restrict λ_i to positive values while allowing λ_i' to range over $\mathbb{R}$.

Finally, the parameters are initialised as follows:

$$\begin{aligned} \mu_i &\sim \mathcal{U}(0,1) && \text{for } i = 1, \ldots, m, \\ \lambda_0 &\sim \chi^2_5, \\ \lambda_i &\sim \chi^2_1 && \text{for } i = 1, \ldots, d, \\ \operatorname{diag}(\mathbf{B}) &\sim \chi^2_4, \\ \text{the rest of } \mathbf{B} &\sim \mathcal{N}(0,1). \end{aligned}$$

The initialisation of $\boldsymbol{\mu}$ mirrors the initialisation of $\mathbf{r}$ in previous work by Levine et al. [19]. While they have constant initial values for $\boldsymbol{\lambda}$, we sample from χ^2 distributions centred around those values (5 for λ_0 and 1 for any other λ_i). The distributions for initial values of $\mathbf{B}$ are simply set to provide a reasonable spread of positive values for the diagonal, and both positive and negative values for all other entries in the matrix.

4.1 Evidence Lower Bound

In this section, we derive and simplify the ELBO for this (now fully specified) model. Note that in order to keep the derivation simple, we drop all constant terms in the expression of $\mathcal{L}$, i.e., equality is taken to mean 'equality up to an additive constant'. Also note that all expected values are with respect to $(\mathbf{u}, \mathbf{r}) \sim q(\mathbf{u}, \mathbf{r})$.

In order to derive the ELBO, let us go back to (5) and write

$$\mathcal{L} = \mathbb{E}[\log p(\mathcal{D}, \mathbf{u}, \mathbf{r})] - \mathbb{E}[\log q(\mathbf{u}, \mathbf{r})].$$

By substituting in (3) and (6), we get

$$\begin{aligned} \mathcal{L} = \mathbb{E}[&\log p(\mathbf{u}) + \log p(\mathbf{r} \mid \mathbf{u}) + \log p(\mathcal{D} \mid \mathbf{r})] \\ &- \mathbb{E}[\log q(\mathbf{u}) + \log q(\mathbf{r} \mid \mathbf{u})]. \end{aligned}$$

Since $q(\mathbf{r} \mid \mathbf{u}) = p(\mathbf{r} \mid \mathbf{u})$, they cancel each other out. Also

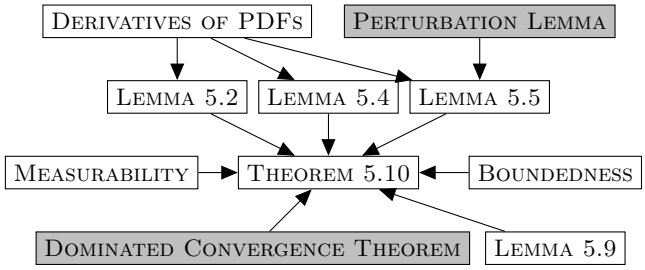

Figure 2: A graphical representation of dependencies between our theoretical results. An arrow from A to B means that A was used to prove B. Results from the literature are in gray.

notice that

$$\begin{aligned}\mathbb{E}[\log p(\mathbf{u}) - \log q(\mathbf{u})] &= -D_{\mathrm{KL}}(q(\mathbf{u}) \parallel p(\mathbf{u})) \\ &= -\frac{1}{2}(\operatorname{tr}(\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\boldsymbol{\Sigma}) + \boldsymbol{\mu}^\intercal \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\boldsymbol{\mu} - m \\ &\quad + \log|\mathbf{K}_{\mathbf{u},\mathbf{u}}| - \log|\boldsymbol{\Sigma}|),\end{aligned}$$

by the definition of KL divergence between two multivariate Gaussians [9]. Hence,

$$\begin{aligned}\mathcal{L} &= \mathbb{E}\left[\sum_{i=1}^{N}\sum_{t=1}^{T} Q_{\mathbf{r}}(s_{i,t}, a_{i,t}) - V_{\mathbf{r}}(s_{i,t})\right] \\ &\quad - \frac{1}{2}\left(\operatorname{tr}\left(\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\boldsymbol{\Sigma}\right) + \boldsymbol{\mu}^\intercal \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\boldsymbol{\mu} + \log|\mathbf{K}_{\mathbf{u},\mathbf{u}}| - \log|\boldsymbol{\Sigma}|\right).\end{aligned}$$

Using the expressions for $Q_{\mathbf{r}}$ we get

$$\begin{aligned}\mathcal{L} &= \mathbb{E}\left[\sum_{i=1}^{N}\sum_{t=1}^{T} r(s_{i,t}) - V_{\mathbf{r}}(s_{i,t}) + \gamma\sum_{s'\in\mathcal{S}}\mathcal{T}(s_{i,t}, a_{i,t}, s')V_{\mathbf{r}}(s')\right] \\ &\quad - \frac{1}{2}\left(\operatorname{tr}\left(\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\boldsymbol{\Sigma}\right) + \boldsymbol{\mu}^\intercal \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\boldsymbol{\mu} + \log|\mathbf{K}_{\mathbf{u},\mathbf{u}}| - \log|\boldsymbol{\Sigma}|\right).\end{aligned}$$

We can simplify $\sum_{i=1}^{N}\sum_{t=1}^{T} r(s_{i,t})$ by defining a new vector $\mathbf{t} = (t_1, \ldots, t_{|\mathcal{S}|})^\intercal$, where t_i is the number of times the state associated with the reward r_i has been visited across all demonstrations. Then

$$\begin{aligned}\mathbb{E}\left[\sum_{i=1}^{N}\sum_{t=1}^{T} r(s_{i,t})\right] &= \mathbb{E}[\mathbf{t}^\intercal\mathbf{r}] = \mathbf{t}^\intercal\mathbb{E}[\mathbf{r}] \\ &= \mathbf{t}^\intercal\mathbb{E}\left[\mathbf{K}_{\mathbf{r},\mathbf{u}}^\intercal\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{u}\right] = \mathbf{t}^\intercal\mathbf{K}_{\mathbf{r},\mathbf{u}}^\intercal\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\boldsymbol{\mu}.\end{aligned}$$

This allows us to simplify $\mathcal{L}$ to

$$\begin{aligned}\mathcal{L} &= \mathbf{t}^\intercal\mathbf{K}_{\mathbf{r},\mathbf{u}}^\intercal\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\boldsymbol{\mu} - \mathbb{E}[v] \\ &\quad - \frac{1}{2}\left(\operatorname{tr}\left(\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\boldsymbol{\Sigma}\right) + \boldsymbol{\mu}^\intercal\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\boldsymbol{\mu} + \log|\mathbf{K}_{\mathbf{u},\mathbf{u}}| - \log|\boldsymbol{\Sigma}|\right),\end{aligned}$$

where

$$v = \sum_{i=1}^{N}\sum_{t=1}^{T} V_{\mathbf{r}}(s_{i,t}) - \gamma\sum_{s'\in\mathcal{S}}\mathcal{T}(s_{i,t}, a_{i,t}, s')V_{\mathbf{r}}(s').$$

5. THEORETICAL JUSTIFICATION

The typical way to optimise a quantity (the ELBO, in this case) involves computing its gradient. Unfortunately, the term $\mathbb{E}[v]$ in $\mathcal{L}$ complicates the situation. The goal of this section is to show how Theorem 3.3 can be applied to our model in order to derive the gradient anyway[3]. After showing that the theorem applies to our situation, we can express $\nabla\mathbb{E}[v]$ as

$$\begin{aligned}\nabla\mathbb{E}[v] &= \nabla\iint q(\mathbf{r} \mid \mathbf{u})q(\mathbf{u})\,d\mathbf{r}\,d\mathbf{u} \\ &= \iint \nabla[vq(\mathbf{r} \mid \mathbf{u})q(\mathbf{u})]\,d\mathbf{r}\,d\mathbf{u} \\ &= \iint \frac{\nabla[vq(\mathbf{r} \mid \mathbf{u})q(\mathbf{u})]}{q(\mathbf{r} \mid \mathbf{u})q(\mathbf{u})} q(\mathbf{r} \mid \mathbf{u})q(\mathbf{u})\,d\mathbf{r}\,d\mathbf{u} \\ &\approx \frac{1}{S}\sum_{s=1}^{S}\frac{\nabla[vq(\mathbf{r} \mid \mathbf{u})q(\mathbf{u})]}{q(\mathbf{r} \mid \mathbf{u})q(\mathbf{u})},\end{aligned}$$

which can be computed by drawing S Monte Carlo samples $(\mathbf{u}_s, \mathbf{r}_s) \sim q(\mathbf{u}, \mathbf{r})$.

Our main goal is Theorem 5.10, which allows us to move differentiation inside the integral. In order to prove it, we use a number of intermediate results. We start by stating a few derivatives of probability density functions (PDFs) and covariance matrices, and bound their values with some easy-to-deal-with polynomials. We then provide a sketch proof of the measurability of MDP value functions, which is non-obvious due to their non-trivial definition. Afterwards, we establish bounds for the value functions, and, after another quick lemma, tackle the main proof of this paper. See Figure 2 for an overview of how these results fit together.

Before that, however, we define a few extra variables in order to simplify expressions of derivatives:

$$\begin{aligned}\mathbf{U} &= (\mathbf{u} - \boldsymbol{\mu})(\mathbf{u} - \boldsymbol{\mu})^\intercal, \\ \mathbf{S} &= \mathbf{K}_{\mathbf{r},\mathbf{u}}^\intercal\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}, \\ \boldsymbol{\Gamma} &= \mathbf{K}_{\mathbf{r},\mathbf{r}} - \mathbf{S}\mathbf{K}_{\mathbf{r},\mathbf{u}}, \\ \mathbf{R} &= \mathbf{S}\frac{\partial\mathbf{K}_{\mathbf{r},\mathbf{u}}}{\partial\lambda_i} - \frac{\partial\mathbf{K}_{\mathbf{r},\mathbf{r}}}{\partial\lambda_i} + \left(\frac{\partial\mathbf{K}_{\mathbf{r},\mathbf{u}}^\intercal}{\partial\lambda_i} - \mathbf{S}\frac{\partial\mathbf{K}_{\mathbf{u},\mathbf{u}}}{\partial\lambda_i}\right)\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{K}_{\mathbf{r},\mathbf{u}}, \\ Q &= (\mathbf{u} - \boldsymbol{\mu})^\intercal\boldsymbol{\Sigma}^{-1}(\mathbf{u} - \boldsymbol{\mu}).\end{aligned}$$

Also note that throughout this section the word 'constant' means 'constant with respect to $\mathbf{u}$ and $\mathbf{r}$'.

Lemma 5.1 (Derivatives of PDFs).

1. $\frac{\partial q(\mathbf{u})}{\partial\boldsymbol{\mu}} = \frac{1}{2}q(\mathbf{u})(\boldsymbol{\Sigma}^{-1} + \boldsymbol{\Sigma}^{-\intercal})(\mathbf{u} - \boldsymbol{\mu})$.
2. (a) $\frac{\partial q(\mathbf{u})}{\partial\boldsymbol{\Sigma}} = \frac{1}{2}q(\mathbf{u})(\boldsymbol{\Sigma}^{-1}\mathbf{U}\boldsymbol{\Sigma}^{-1} - \boldsymbol{\Sigma}^{-1})$.
   (b) $\frac{\partial q(\mathbf{u})}{\partial\mathbf{B}} = q(\mathbf{u})(\boldsymbol{\Sigma}^{-1}\mathbf{U}\boldsymbol{\Sigma}^{-1} - \boldsymbol{\Sigma}^{-1})\mathbf{B}$.
3. *For* $i = 0, \ldots, d$,
   (a)
   $$\begin{aligned}\frac{\partial q(\mathbf{r} \mid \mathbf{u})}{\partial\lambda_i} &= \frac{1}{2}q(\mathbf{r} \mid \mathbf{u})(|\boldsymbol{\Gamma}|^{-1}\operatorname{tr}(\mathbf{R}\operatorname{adj}(\boldsymbol{\Gamma})) \\ &\quad - (\mathbf{r} - \mathbf{S}\mathbf{u})^\intercal\boldsymbol{\Gamma}^{-1}\mathbf{R}\boldsymbol{\Gamma}^{-1}(\mathbf{r} - \mathbf{S}\mathbf{u})).\end{aligned}$$
   (b) *For any covariance matrix* $\mathbf{K}$,
   $$\frac{\partial\mathbf{K}}{\partial\lambda_i} = \begin{cases}\frac{1}{\lambda_i}\mathbf{K} & \text{if } i = 0, \\ \mathbf{L} & \text{otherwise,}\end{cases}$$

[3]This technique is inspired by black box VI [31], but takes a more detailed look at the problem and requires significantly more work to prove correctness.

where

$$L_{j,k} = k(\mathbf{x}_j, \mathbf{x}_k)\left(-\frac{1}{2}(x_{j,i} - x_{k,i})^2 - \mathbb{1}[j \neq k]\sigma^2\right).$$

LEMMA 5.2. *Let $i \in \{0, \ldots, d\}$ and $\epsilon > 0$ be arbitrary. Furthermore, let $c : \mathbb{R}^{|\mathcal{S}|} \times \mathbb{R}^m \to (\lambda_i - \epsilon, \lambda_i + \epsilon) \subset \mathbb{R}$ be a function with a codomain arbitrarily close to λ_i. Then*

$$\left.\frac{\partial q(\mathbf{r} \mid \mathbf{u})}{\partial \lambda_i}\right|_{\lambda_i = c(\mathbf{r},\mathbf{u})}$$

has upper and lower bounds of the form $q(\mathbf{r} \mid \mathbf{u})d(\mathbf{u})$, where $d(\mathbf{u}) \in \mathbb{R}_2[\mathbf{u}]$.

PROOF. Remember that

$$\frac{\partial q(\mathbf{r} \mid \mathbf{u})}{\partial \lambda_i} = \frac{1}{2}q(\mathbf{r} \mid \mathbf{u})(|\mathbf{\Gamma}|^{-1}\operatorname{tr}(\mathbf{R}\operatorname{adj}(\mathbf{\Gamma})) - (\mathbf{r} - \mathbf{S}\mathbf{u})^\intercal\mathbf{\Gamma}^{-1}\mathbf{R}\mathbf{\Gamma}^{-1}(\mathbf{r} - \mathbf{S}\mathbf{u})).$$

by Lemma 5.1. Let $\mathbf{K}$ be any covariance matrix and

$$\mathbf{A} = \frac{1}{\lambda_0}\mathbf{K}.$$

First, we will show that

$$\mathbf{K}|_{\lambda_i = c(\mathbf{r},\mathbf{u})} \to 0 \quad \text{and} \quad \left.\frac{\partial \mathbf{K}}{\partial \lambda_i}\right|_{\lambda_i = c(\mathbf{r},\mathbf{u})} \to 0$$

as $\epsilon \to 0$. We can easily establish constant upper and lower bounds on $\mathbf{K}|_{\lambda_i = c(\mathbf{r},\mathbf{u})}$ by the boundedness of c and continuity of the covariance function. Similar reasoning combined with the expressions for derivatives of covariance matrices in Lemma 5.1 gives constant upper and lower bounds on the elements of

$$\left.\frac{\partial \mathbf{K}}{\partial \lambda_i}\right|_{\lambda_i = c(\mathbf{r},\mathbf{u})}$$

as well.

Now, we will show that $\mathbf{K}^{-1}|_{\lambda_i = c(\mathbf{r},\mathbf{u})}$ exists and

$$\lim_{\epsilon \to 0} \mathbf{K}^{-1}|_{\lambda_i = c(\mathbf{r},\mathbf{u})} = \mathbf{K}.$$

If $i = 0$, then $\mathbf{K}|_{\lambda_i = c(\mathbf{r},\mathbf{u})} = c(\mathbf{r}, \mathbf{u})\mathbf{A}$. Therefore[4],

$$\mathbf{K}^{-1}|_{\lambda_i = c(\mathbf{r},\mathbf{u})} = \frac{1}{c(\mathbf{r}, \mathbf{u})}\mathbf{A}^{-1} \to \frac{1}{\lambda_0}\mathbf{A}^{-1} = \frac{1}{\lambda_0}\left(\frac{1}{\lambda_0}\mathbf{K}\right)^{-1} = \mathbf{K}^{-1}$$

as $\epsilon \to 0$.

We will now show the same result for $\lambda_i > 0$. Let

$$S = \sum_{n \in \{1,\ldots,d\} \setminus \{i\}} \frac{\lambda_n}{2}(x_{j,n} - x_{k,n})^2 + \mathbb{1}[j \neq k]\sigma^2\lambda_n$$

and $\delta = c(\mathbf{r}, \mathbf{u}) - \lambda_i$ so that $c(\mathbf{r}, \mathbf{u}) = \lambda_i + \delta$, and $\lim_{\epsilon \to 0} \delta = 0$.

[4]Note that since $\lambda_0 \neq 0$, $c(\mathbf{r}, \mathbf{u}) \neq 0$ for small enough ϵ.

Then,

$$\begin{aligned}
k(\mathbf{x}_j, \mathbf{x}_k)|_{\lambda_i = c(\mathbf{r},\mathbf{u})} &= \lambda_0 \exp\Big(-\frac{1}{2}c(\mathbf{r}, \mathbf{u})(x_{j,i} - x_{k,i})^2 \\
&\qquad - \mathbb{1}[j \neq k]\sigma^2 c(\mathbf{r}, \mathbf{u}) - S \Big) \\
&= \lambda_0 \exp\Big(-\frac{1}{2}(\lambda_i + \delta)(x_{j,i} - x_{k,i})^2 \\
&\qquad - \mathbb{1}[j \neq k]\sigma^2(\lambda_i + \delta) - S \Big) \\
&= \lambda_0 \exp\Big(-\frac{1}{2}(\mathbf{x}_j - \mathbf{x}_k)^\intercal\mathbf{\Lambda}(\mathbf{x}_j - \mathbf{x}_k) - \mathbb{1}[j \neq k]\sigma^2 \operatorname{tr}(\mathbf{\Lambda}) \\
&\qquad - \frac{\delta}{2}(x_{j,i} - x_{k,i})^2 - \mathbb{1}[j \neq k]\sigma^2\delta \Big) \\
&= k(\mathbf{x}_j, \mathbf{x}_k) \exp\Big(-\frac{\delta}{2}(x_{j,i} - x_{k,i})^2 - \mathbb{1}[j \neq k]\sigma^2\delta \Big) \\
&= k(\mathbf{x}_j, \mathbf{x}_k) + k(\mathbf{x}_j, \mathbf{x}_k)\Big(\exp\Big(-\frac{\delta}{2}(x_{j,i} - x_{k,i})^2 \\
&\qquad - \mathbb{1}[j \neq k]\sigma^2\delta \Big) - 1\Big)
\end{aligned}$$

Hence, we can express $\mathbf{K}|_{\lambda_i = c(\mathbf{r},\mathbf{u})}$ as $\mathbf{K}|_{\lambda_i = c(\mathbf{r},\mathbf{u})} = \mathbf{K} + \mathbf{E}$, where $\mathbf{E}$ is defined as

$$E_{j,k} = k(\mathbf{x}_j, \mathbf{x}_k)\left(\exp\left(-\frac{\delta}{2}(x_{j,i} - x_{k,i})^2 - \mathbb{1}[j \neq k]\sigma^2\delta\right) - 1\right).$$

By this definition,

$$\lim_{\epsilon \to 0} E_{j,k} = 0.$$

Then, since $\mathbf{K}$ is invertible, Lemma 3.2 shows the existence of $\mathbf{K}^{-1}_{\lambda_i = c(\mathbf{r},\mathbf{u})}$ and gives upper and lower bounds on all of its elements.

This is enough to prove constant upper and lower bounds on $\mathbf{S}$, $\mathbf{\Gamma}$, and $\mathbf{R}$ (all with λ_i replaced with $c(\mathbf{r}, \mathbf{u})$), which means that $(\mathbf{r} - \mathbf{S}\mathbf{u})^\intercal\mathbf{\Gamma}^{-1}\mathbf{R}\mathbf{\Gamma}^{-1}(\mathbf{r} - \mathbf{S}\mathbf{u})|_{\lambda_i = c(\mathbf{r},\mathbf{u})}$ has upper and lower bounds in $\mathbb{R}_2[\mathbf{u}]$.

Since

$$\lim_{\epsilon \to 0} \mathbf{\Gamma}|_{\lambda_i = c(\mathbf{r},\mathbf{u})} = \mathbf{\Gamma},$$

we also have that

$$\lim_{\epsilon \to 0} \det(\mathbf{\Gamma})|_{\lambda_i = c(\mathbf{r},\mathbf{u})} = \det(\mathbf{\Gamma}).$$

Assuming that $\mathbf{\Gamma}$ is invertible so that $q(\mathbf{r} \mid \mathbf{u})$ exists,

$$\det(\mathbf{\Gamma})|_{\lambda_i = c(\mathbf{r},\mathbf{u})} \neq 0$$

for small enough ϵ, and, thus, $\det(\mathbf{\Gamma})^{-1}|_{\lambda_i = c(\mathbf{r},\mathbf{u})}$ exists and has constant bounds.

Recall that

$$\mathbf{\Gamma} = \mathbf{K}_{\mathbf{r},\mathbf{r}} - \mathbf{S}\mathbf{K}_{\mathbf{r},\mathbf{u}} = \mathbf{K}_{\mathbf{r},\mathbf{r}} - \mathbf{K}_{\mathbf{r},\mathbf{u}}^\intercal\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{K}_{\mathbf{r},\mathbf{u}}.$$

We have already demonstrated that for $\mathbf{K} \in \{\mathbf{K}_{\mathbf{r},\mathbf{r}}, \mathbf{K}_{\mathbf{r},\mathbf{u}}, \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\}$,

$$\lim_{\epsilon \to 0} \mathbf{K}|_{\lambda_i = c(\mathbf{r},\mathbf{u})} = \mathbf{K}.$$

We can use a general fact that if $\lim_{x \to 0} f(x) = f$, then $f(x) = f + g(x)$ for some function g such that $\lim_{x \to 0} g(x) = 0$ to define matrices $\mathbf{E}_{\mathbf{r},\mathbf{r}}$, $\mathbf{E}_{\mathbf{r},\mathbf{u}}$, and $\mathbf{E}_{\mathbf{u},\mathbf{u}}$ such that

$$\begin{aligned}
\mathbf{K}_{\mathbf{r},\mathbf{r}}|_{\lambda_i = c(\mathbf{r},\mathbf{u})} &= \mathbf{K}_{\mathbf{r},\mathbf{r}} + \mathbf{E}_{\mathbf{r},\mathbf{r}}, & \mathbf{E}_{\mathbf{r},\mathbf{r}} &\to \mathbf{O}_{|\mathcal{S}|}, \\
\mathbf{K}_{\mathbf{r},\mathbf{u}}|_{\lambda_i = c(\mathbf{r},\mathbf{u})} &= \mathbf{K}_{\mathbf{r},\mathbf{u}} + \mathbf{E}_{\mathbf{r},\mathbf{u}}, \quad \text{and} & \mathbf{E}_{\mathbf{r},\mathbf{u}} &\to \mathbf{O}_{|\mathcal{S}|,m}, \\
\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}|_{\lambda_i = c(\mathbf{r},\mathbf{u})} &= \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} + \mathbf{E}_{\mathbf{u},\mathbf{u}}, & \mathbf{E}_{\mathbf{u},\mathbf{u}} &\to \mathbf{O}_m.
\end{aligned}$$

as $\epsilon \to 0$. Then, $\boldsymbol{\Gamma}|_{\lambda_i = c(\mathbf{r},\mathbf{u})} = \boldsymbol{\Gamma} + \mathbf{E}$, where

$$\begin{aligned}\mathbf{E} = \mathbf{E}_{\mathbf{r},\mathbf{r}} &- \mathbf{K}_{\mathbf{r},\mathbf{u}}^{\intercal}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{E}_{\mathbf{r},\mathbf{u}} - \mathbf{K}_{\mathbf{r},\mathbf{u}}^{\intercal}\mathbf{E}_{\mathbf{u},\mathbf{u}}(\mathbf{K}_{\mathbf{r},\mathbf{u}} + \mathbf{E}_{\mathbf{r},\mathbf{u}}) \\ &- \mathbf{E}_{\mathbf{r},\mathbf{u}}^{\intercal}(\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} + \mathbf{E}_{\mathbf{u},\mathbf{u}})(\mathbf{K}_{\mathbf{r},\mathbf{u}} + \mathbf{E}_{\mathbf{r},\mathbf{u}}) \\ &\to \mathbf{O}_{|\mathcal{S}|}\end{aligned}$$

as $\epsilon \to 0$. Thus, Lemma 3.2 shows that $\boldsymbol{\Gamma}^{-1}|_{\lambda_i = c(\mathbf{r},\mathbf{u})}$ exists and provides constant bounds on its elements.

Since we already know that $\boldsymbol{\Gamma}^{-1}|_{\lambda_i = c(\mathbf{r},\mathbf{u})}$ and $\det(\boldsymbol{\Gamma})|_{\lambda_i = c(\mathbf{r},\mathbf{u})}$ are bounded, so is $\operatorname{adj}(\boldsymbol{\Gamma}) = \det(\boldsymbol{\Gamma})\boldsymbol{\Gamma}^{-1}|_{\lambda_i = c(\mathbf{r},\mathbf{u})}$. Thus, we have constant bounds on $|\boldsymbol{\Gamma}|^{-1}\operatorname{tr}(\mathbf{R}\operatorname{adj}(\boldsymbol{\Gamma}))|_{\lambda_i = c(\mathbf{r},\mathbf{u})}$, which means that

$$\left.\frac{\partial q(\mathbf{r} \mid \mathbf{u})}{\partial \lambda_i}\right|_{\lambda_i = c(\mathbf{r},\mathbf{u})}$$

has the required bounds. □

Remark 5.3. *In order to find a derivative such as $\frac{\partial q(\mathbf{u})}{\partial \mu_i}$, we can find $\frac{\partial q(\mathbf{u})}{\partial \boldsymbol{\mu}}$ and simply take the ith element. A similar line of reasoning applies to matrices as well. Thus, we only need to consider derivatives with respect to $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$.*

Lemma 5.4. *Let $c : \mathbb{R}^{|\mathcal{S}|} \times \mathbb{R}^m \to (a, b) \subset \mathbb{R}$ be an arbitrary bounded function. Then, for $i = 1, \ldots, m$, every element of*

$$\left.\frac{\partial q(\mathbf{u})}{\partial \boldsymbol{\mu}}\right|_{\mu_i = c(\mathbf{r},\mathbf{u})}$$

has upper and lower bounds of the form $q(\mathbf{u})d(\mathbf{u})$, where $d(\mathbf{u}) \in \mathbb{R}_1[\mathbf{u}]$.

Proof. Using Lemma 5.1,

$$\left.\frac{\partial q(\mathbf{u})}{\partial \boldsymbol{\mu}}\right|_{\mu_i = c(\mathbf{r},\mathbf{u})} = \frac{1}{2}q(\mathbf{u})(\boldsymbol{\Sigma}^{-1} + \boldsymbol{\Sigma}^{-\intercal})(\mathbf{u} - \mathbf{c}(\mathbf{r},\mathbf{u})),$$

where $\mathbf{c}(\mathbf{r},\mathbf{u}) = (\mu_1, \ldots, \mu_{i-1}, c(\mathbf{r},\mathbf{u}), \mu_{i+1} \ldots, \mu_m)^{\intercal}$. Since $c(\mathbf{r},\mathbf{u})$ is bounded and $\boldsymbol{\Sigma}^{-1} + \boldsymbol{\Sigma}^{-\intercal}$ is a constant matrix, we can use the bounds on $c(\mathbf{r},\mathbf{u})$ to manufacture both upper and lower bounds on

$$\left.\frac{\partial q(\mathbf{u})}{\partial \boldsymbol{\mu}}\right|_{\mu_i = c(\mathbf{r},\mathbf{u})}$$

of the required form. □

Lemma 5.5. *Let $i, j = 1, \ldots, m$, and let $\epsilon > 0$ be arbitrary. Furthermore, let*

$$c : \mathbb{R}^{|\mathcal{S}|} \times \mathbb{R}^m \to (\Sigma_{i,j} - \epsilon, \Sigma_{i,j} + \epsilon) \subset \mathbb{R}$$

be a function with a codomain arbitrarily close to $\Sigma_{i,j}$. Then every element of

$$\left.\frac{\partial q(\mathbf{u})}{\partial \boldsymbol{\Sigma}}\right|_{\Sigma_{i,j} = c(\mathbf{r},\mathbf{u})}$$

has upper and lower bounds of the form $q(\mathbf{u})d(\mathbf{u})$, where $d(\mathbf{u}) \in \mathbb{R}_2[\mathbf{u}]$.

Proof. Using Lemma 5.1,

$$\left.\frac{\partial q(\mathbf{u})}{\partial \boldsymbol{\Sigma}}\right|_{\Sigma_{i,j} = c(\mathbf{r},\mathbf{u})} = \frac{1}{2}q(\mathbf{u})(\mathbf{C}(\mathbf{r},\mathbf{u})^{-\intercal}\mathbf{U}\mathbf{C}(\mathbf{r},\mathbf{u})^{-\intercal} - \mathbf{C}(\mathbf{r},\mathbf{u})^{-\intercal}),$$

where

$$[\mathbf{C}(\mathbf{r},\mathbf{u})]_{k,l} = \begin{cases} c(\mathbf{r},\mathbf{u}) & \text{if } (k,l) = (i,j), \\ \Sigma_{k,l} & \text{otherwise.} \end{cases}$$

We can also express $\mathbf{C}(\mathbf{r},\mathbf{u})$ as $\mathbf{C}(\mathbf{r},\mathbf{u}) = \boldsymbol{\Sigma} + \mathbf{E}(\mathbf{r},\mathbf{u})$, where

$$[\mathbf{E}(\mathbf{r},\mathbf{u})]_{k,l} = \begin{cases} c(\mathbf{r},\mathbf{u}) - \Sigma_{i,j} & \text{if } (k,l) = (i,j), \\ 0 & \text{otherwise.} \end{cases}$$

We begin by establishing upper and lower bounds on $\mathbf{C}(\mathbf{r},\mathbf{u})^{-1}$. For this, we use the maximum norm $\|\cdot\|_\infty$ on both vectors and matrices. We can apply Lemma 3.2 to $\boldsymbol{\Sigma}$ and $\mathbf{E}(\mathbf{r},\mathbf{u})$ since

$$\|\mathbf{E}(\mathbf{r},\mathbf{u})\|_\infty = \max_k \sum_l |[\mathbf{E}(\mathbf{r},\mathbf{u})]_{k,l}| = |c(\mathbf{r},\mathbf{u}) - \Sigma_{i,j}| < \epsilon$$

can be made arbitrarily small so that $\|\boldsymbol{\Sigma}^{-1}\|_\infty \|\mathbf{E}(\mathbf{r},\mathbf{u})\|_\infty < 1$. Then $\mathbf{C}(\mathbf{r},\mathbf{u})$ is invertible, and

$$\|\mathbf{C}(\mathbf{r},\mathbf{u})^{-1}\|_\infty \leq \frac{\|\boldsymbol{\Sigma}^{-1}\|_\infty}{1 - \|\boldsymbol{\Sigma}^{-1}\|_\infty \|\mathbf{E}(\mathbf{r},\mathbf{u})\|_\infty} < \frac{\|\boldsymbol{\Sigma}^{-1}\|_\infty}{1 - \|\boldsymbol{\Sigma}^{-1}\|_\infty \epsilon},$$

which means that

$$\max_k \sum_l \left|[\mathbf{C}(\mathbf{r},\mathbf{u})^{-1}]_{k,l}\right| < \frac{\|\boldsymbol{\Sigma}^{-1}\|_\infty}{1 - \|\boldsymbol{\Sigma}^{-1}\|_\infty \epsilon},$$

i.e., for any row k and column l,

$$\left|[\mathbf{C}(\mathbf{r},\mathbf{u})^{-1}]_{k,l}\right| < \frac{\|\boldsymbol{\Sigma}^{-1}\|_\infty}{1 - \|\boldsymbol{\Sigma}^{-1}\|_\infty \epsilon},$$

which bounds all elements of $\mathbf{C}(\mathbf{r},\mathbf{u})^{-1}$ as required. Since every element of $\mathbf{U} = (\mathbf{u} - \boldsymbol{\mu})(\mathbf{u} - \boldsymbol{\mu})^{\intercal}$ is in $\mathbb{R}_2[\mathbf{u}]$, and the elements of $\mathbf{C}(\mathbf{r},\mathbf{u})^{-1}$ are bounded, the desired result follows. □

Remark 5.6. *MDP values are characterised by both a state and a reward function/vector. In this section, we think of the value function as $V : \mathcal{S} \to \mathbb{R}^{|\mathcal{S}|} \to \mathbb{R}$, i.e., V takes a state $s \in \mathcal{S}$ and returns a function $V(s) : \mathbb{R}^{|\mathcal{S}|} \to \mathbb{R}$ that takes a reward vector $\mathbf{r} \in \mathbb{R}^{|\mathcal{S}|}$ and returns a value of the state s, $V_{\mathbf{r}}(s) \in \mathbb{R}$. Given a reward vector, the function $V(s)$ computes the values of all states and returns the value of state s.*

Proposition 5.7 (Measurability). *MDP value functions $V(s) : \mathbb{R}^{|\mathcal{S}|} \to \mathbb{R}$ (for $s \in \mathcal{S}$) are Lebesgue measurable.*

Proof. For any reward vector $\mathbf{r} \in \mathbb{R}^{|\mathcal{S}|}$, the collection of converged value functions $\{V_{\mathbf{r}}(s) \mid s \in \mathcal{S}\}$ satisfy

$$V_{\mathbf{r}}(s) = \log \sum_{a \in \mathcal{A}} \exp\left(r(s) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{T}(s,a,s')V_{\mathbf{r}}(s')\right) \quad (7)$$

for all $s \in \mathcal{S}$. Let $s_0 \in \mathcal{S}$ be an arbitrary state. In order to prove that $V(s_0)$ is measurable, it is enough to show that for any $\alpha \in \mathbb{R}$, the set

$$\left\{ \mathbf{r} \in \mathbb{R}^{|\mathcal{S}|} \;\middle|\; \begin{array}{l} V_{\mathbf{r}}(s_0) \in (-\infty, \alpha); \\ V_{\mathbf{r}}(s) \in \mathbb{R} \text{ for all } s \in \mathcal{S} \setminus \{s_0\}; \\ (7) \text{ is satisfied by all } s \in \mathcal{S} \end{array} \right\}$$

is measurable. Since this set can be constructed in Zermelo-Fraenkel set theory *without* the axiom of choice, it is measurable [13], which proves that $V(s)$ is a measurable function for any $s \in \mathcal{S}$. □

Proposition 5.8 (Boundedness). *If the initial values of the MDP value function satisfy the following bound, then the bound remains satisfied throughout value iteration:*

$$|V_{\mathbf{r}}(s)| \leq \frac{\|\mathbf{r}\|_\infty + \log|\mathcal{A}|}{1 - \gamma}. \quad (8)$$

PROOF. We begin by considering (8) without taking the absolute value of $V_{\mathbf{r}}(s)$, i.e.,

$$V_{\mathbf{r}}(s) \le \frac{\|\mathbf{r}\|_\infty + \log|\mathcal{A}|}{1-\gamma}, \tag{9}$$

and assuming that the initial values of $\{V_{\mathbf{r}}(s) \mid s \in \mathcal{S}\}$ already satisfy (9). Recall that for each $s \in \mathcal{S}$, the value of $V_{\mathbf{r}}(s)$ is updated by applying (1). Note that both log and exp are increasing functions, $\gamma > 0$, and the $\mathcal{T}$ function gives a probability (a non-negative number). Thus

$$\begin{aligned} V_{\mathbf{r}}(s) &\le \log \sum_{a\in\mathcal{A}} \exp\left(r(s) + \gamma \sum_{s'\in\mathcal{S}} \mathcal{T}(s,a,s') \frac{\|\mathbf{r}\|_\infty + \log|\mathcal{A}|}{1-\gamma}\right) \\ &= \log \sum_{a\in\mathcal{A}} \exp\left(r(s) + \frac{\gamma(\|\mathbf{r}\|_\infty + \log|\mathcal{A}|)}{1-\gamma} \sum_{s'\in\mathcal{S}} \mathcal{T}(s,a,s')\right) \\ &= \log \sum_{a\in\mathcal{A}} \exp\left(r(s) + \frac{\gamma(\|\mathbf{r}\|_\infty + \log|\mathcal{A}|)}{1-\gamma}\right) \end{aligned}$$

by the definition of $\mathcal{T}$. Then

$$\begin{aligned} V_{\mathbf{r}}(s) &\le \log\left(|\mathcal{A}| \exp\left(r(s) + \frac{\gamma(\|\mathbf{r}\|_\infty + \log|\mathcal{A}|)}{1-\gamma}\right)\right) \\ &= \log\left(\exp\left(\log|\mathcal{A}| + r(s) + \frac{\gamma(\|\mathbf{r}\|_\infty + \log|\mathcal{A}|)}{1-\gamma}\right)\right) \\ &= \log|\mathcal{A}| + r(s) + \frac{\gamma(\|\mathbf{r}\|_\infty + \log|\mathcal{A}|)}{1-\gamma} \\ &= \frac{\gamma(\|\mathbf{r}\|_\infty + \log|\mathcal{A}|) + (1-\gamma)(\log|\mathcal{A}| + r(s))}{1-\gamma} \\ &\le \frac{\gamma(\|\mathbf{r}\|_\infty + \log|\mathcal{A}|) + (1-\gamma)(\log|\mathcal{A}| + \|\mathbf{r}\|_\infty)}{1-\gamma} \\ &= \frac{\|\mathbf{r}\|_\infty + \log|\mathcal{A}|}{1-\gamma} \end{aligned}$$

by the definition of $\|\mathbf{r}\|_\infty$.

The proof for

$$V_{\mathbf{r}}(s) \ge \frac{\|\mathbf{r}\|_\infty + \log|\mathcal{A}|}{\gamma-1} \tag{10}$$

follows the same argument until we get to

$$\begin{aligned} V_{\mathbf{r}}(s) &\ge \frac{\gamma(\|\mathbf{r}\|_\infty + \log|\mathcal{A}|) + (\gamma-1)(\log|\mathcal{A}| + r(s))}{\gamma-1} \\ &\ge \frac{\gamma(\|\mathbf{r}\|_\infty + \log|\mathcal{A}|) + (\gamma-1)(-\log|\mathcal{A}| - \|\mathbf{r}\|_\infty)}{\gamma-1} \\ &= \frac{\|\mathbf{r}\|_\infty + \log|\mathcal{A}|}{\gamma-1}, \end{aligned}$$

where we use the fact that $r(s) \ge -\|\mathbf{r}\|_\infty - 2\log|\mathcal{A}|$. Combining (9) and (10) gives (8). □

LEMMA 5.9.

$$\int \|\mathbf{r}\|_\infty q(\mathbf{r} \mid \mathbf{u})\, d\mathbf{r} \le a + \|\mathbf{K}_{\mathbf{r},\mathbf{u}}^\intercal \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} \mathbf{u}\|_1,$$

where a is a constant independent of $\mathbf{u}$.

PROOF. Since $\|\mathbf{r}\|_\infty \le \|\mathbf{r}\|_1$,

$$\int \|\mathbf{r}\|_\infty q(\mathbf{r} \mid \mathbf{u})\, d\mathbf{r} \le \int \|\mathbf{r}\|_1 q(\mathbf{r} \mid \mathbf{u})\, d\mathbf{r} = \sum_{i=1}^{|\mathcal{S}|} \mathbb{E}[|r_i|].$$

As each $\mathbb{E}[|r_i|]$ is a mean of a folded Gaussian distribution,

$$\mathbb{E}[|r_i|] = \sigma_i \sqrt{\frac{2}{\pi}} \exp\left(-\frac{\xi_i^2}{2\sigma_i^2}\right) + \xi_i \left(1 - 2\Phi\left(-\frac{\xi_1}{\sigma_1}\right)\right),$$

where $\xi_i = \left[\mathbf{K}_{\mathbf{r},\mathbf{u}}^\intercal \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} \mathbf{u}\right]_i$, $\sigma_i = \sqrt{[\mathbf{K}_{\mathbf{r},\mathbf{r}} - \mathbf{K}_{\mathbf{r},\mathbf{u}}^\intercal \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} \mathbf{K}_{\mathbf{r},\mathbf{u}}]_{i,i}}$[5], and Φ is the cumulative distribution function of the standard Gaussian. Furthermore,

$$\mathbb{E}[|r_i|] \le \sigma_i \sqrt{\frac{2}{\pi}} + |\xi_i|,$$

as σ_i is non-negative, and $\Phi(x) \in [0,1]$ for all x. Since

$$\sum_{i=1}^{|\mathcal{S}|} |\xi_i| = \|\mathbf{K}_{\mathbf{r},\mathbf{u}}^\intercal \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} \mathbf{u}\|_1,$$

we can set

$$a = \sum_{i=1}^{|\mathcal{S}|} \sigma_i \sqrt{\frac{2}{\pi}}$$

to get the desired result. □

Our main theorem is a specialised version of an integral differentiation result by Chen [6].

THEOREM 5.10. *Whenever the derivative exists,*

$$\frac{\partial}{\partial t} \iint V_{\mathbf{r}}(s) q(\mathbf{r} \mid \mathbf{u}) q(\mathbf{u})\, d\mathbf{r}\, d\mathbf{u} = \iint \frac{\partial}{\partial t} [V_{\mathbf{r}}(s) q(\mathbf{r} \mid \mathbf{u}) q(\mathbf{u})]\, d\mathbf{r}\, d\mathbf{u},$$

where t is any scalar part of $\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$, or $\boldsymbol{\lambda}$.

PROOF. Let

$$\begin{aligned} f(\mathbf{r},\mathbf{u},t) &= V_{\mathbf{r}}(s) q(\mathbf{r} \mid \mathbf{u}) q(\mathbf{u}), \\ F(t) &= \iint f(\mathbf{r},\mathbf{u},t)\, d\mathbf{r}\, d\mathbf{u}, \end{aligned}$$

and fix the value of t. Let $(t_n)_{n=1}^\infty$ be any sequence such that $\lim_{n\to\infty} t_n = t$, but $t_n \ne t$ for all n. We want to show that

$$F'(t) = \lim_{n\to\infty} \frac{F(t_n) - F(t)}{t_n - t} = \iint \left.\frac{\partial f}{\partial t}\right|_{(\mathbf{r},\mathbf{u},t)} d\mathbf{r}\, d\mathbf{u}. \tag{11}$$

We have

$$\begin{aligned} \frac{F(t_n) - F(t)}{t_n - t} &= \iint \frac{f(\mathbf{r},\mathbf{u},t_n) - f(\mathbf{r},\mathbf{u},t)}{t_n - t}\, d\mathbf{r}\, d\mathbf{u} \\ &= \iint f_n(\mathbf{r},\mathbf{u})\, d\mathbf{r}\, d\mathbf{u}, \end{aligned}$$

where

$$f_n(\mathbf{r},\mathbf{u}) = \frac{f(\mathbf{r},\mathbf{u},t_n) - f(\mathbf{r},\mathbf{u},t)}{t_n - t}.$$

Since

$$\lim_{n\to\infty} f_n(\mathbf{r},\mathbf{u}) = \left.\frac{\partial f}{\partial t}\right|_{(\mathbf{r},\mathbf{u},t)},$$

(11) follows from Theorem 3.3 as soon as we show that both f and f_n are measurable and find a non-negative integrable function g such that for all n, $\mathbf{r}$, $\mathbf{u}$,

$$|f_n(\mathbf{r},\mathbf{u})| \le g(\mathbf{r},\mathbf{u}).$$

[5]The expression under the square root sign is non-negative because $\mathbf{K}_{\mathbf{r},\mathbf{r}} - \mathbf{K}_{\mathbf{r},\mathbf{u}}^\intercal \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} \mathbf{K}_{\mathbf{r},\mathbf{u}}$ is a covariance matrix of a Gaussian distribution, hence also positive semi-definite, which means that its diagonal entries are non-negative.

The MDP value function is measurable by Proposition 5.7. The result of multiplying or adding measurable functions (e.g., probability density functions) to a measurable function is still measurable. Thus, both f and f_n are measurable.

It remains to find g. For notational simplicity and without loss of generality, we will temporarily assume that t is a parameter of $q(\mathbf{r} \mid \mathbf{u})$. Then

$$|f_n(\mathbf{r}, \mathbf{u})| = |V_{\mathbf{r}}(s)| \left| \frac{q(\mathbf{r} \mid \mathbf{u})|_{t=t_n} - q(\mathbf{r} \mid \mathbf{u})}{t_n - t} \right| q(\mathbf{u})$$

since PDFs are non-negative. An upper bound for $|V_{\mathbf{r}}(s)|$ is given by Proposition 5.8, while

$$\frac{q(\mathbf{r} \mid \mathbf{u})|_{t=t_n} - q(\mathbf{r} \mid \mathbf{u})}{t_n - t} = \left. \frac{\partial q(\mathbf{r} \mid \mathbf{u})}{\partial t} \right|_{t=c(\mathbf{r},\mathbf{u})}$$

for some function $c : \mathbb{R}^{|\mathcal{S}|} \times \mathbb{R}^m \to (\min\{t, t_n\}, \max\{t, t_n\})$ due to the mean value theorem (since q is a continuous and differentiable function of t, regardless of the specific choices of q and t).

We then have that

$$|f_n(\mathbf{r}, \mathbf{u})| \leq \frac{\|\mathbf{r}\|_\infty + \log|\mathcal{A}|}{1-\gamma} \left| \left. \frac{\partial q(\mathbf{r} \mid \mathbf{u})}{\partial t} \right|_{t=c(\mathbf{r},\mathbf{u})} \right| q(\mathbf{u}).$$

The bound is clearly non-negative and measurable. It remains to show that it is also integrable. Depending on what t represents, we can use one of the Lemmas 5.2, 5.4, and 5.5, which gives us two polynomials $p_1(\mathbf{u}), p_2(\mathbf{u}) \in \mathbb{R}_2[\mathbf{u}]$ such that

$$p_1(\mathbf{u}) q(\mathbf{r} \mid \mathbf{u}) < \left. \frac{\partial q(\mathbf{r} \mid \mathbf{u})}{\partial t} \right|_{t=c(\mathbf{r},\mathbf{u})} < p_2(\mathbf{u}) q(\mathbf{r} \mid \mathbf{u}).$$

Then

$$\left| \left. \frac{\partial q(\mathbf{r} \mid \mathbf{u})}{\partial t} \right|_{t=c(\mathbf{r},\mathbf{u})} \right| < q(\mathbf{r} \mid \mathbf{u}) \max\{|p_1(\mathbf{u})|, |p_2(\mathbf{u})|\}.$$

We can now apply Lemma 5.9, which allows us to integrate out $\mathbf{r}$, and we are left with showing the existence of

$$\int \left(a + \|\mathbf{K}_{\mathbf{r},\mathbf{u}}^\intercal \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} \mathbf{u}\|_1\right) \max\{|p_1(\mathbf{u})|, |p_2(\mathbf{u})|\} q(\mathbf{u})\, d\mathbf{u}, \tag{12}$$

where a is a constant. The integral

$$\int \max \left\{ \begin{matrix} |p_1(\mathbf{u})|, \\ |p_2(\mathbf{u})| \end{matrix} \right\} q(\mathbf{u})\, d\mathbf{u} = \int \max \left\{ \begin{matrix} |p_1(\mathbf{u}) q(\mathbf{u})|, \\ |p_2(\mathbf{u}) q(\mathbf{u})| \end{matrix} \right\} d\mathbf{u}$$

exists because $p_1(\mathbf{u})q(\mathbf{u})$ and $p_2(\mathbf{u})q(\mathbf{u})$ are both integrable, hence their absolute values are integrable, and the maximum of two integrable functions is also integrable. Since $\|\mathbf{K}_{\mathbf{r},\mathbf{u}}^\intercal \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} \mathbf{u}\|_1 \in \mathbb{R}_1[\mathbf{u}]$, a similar argument can be applied to the rest of (12) as well. □

6. EVALUATION

In order to fully understand the model's behaviour, we focus on a three-state MDP where the agent can deterministically move from any state to any other state. More formally, we set $\mathcal{S} = \{s_1, s_2, s_3\}$, $\mathcal{A} = \{a_1, a_2\}$,

$$\begin{aligned} \mathcal{T}(s_1, a_1, s_2) &= 1, \quad \mathcal{T}(s_1, a_2, s_3) = 1, \\ \mathcal{T}(s_2, a_1, s_1) &= 1, \quad \mathcal{T}(s_1, a_2, s_3) = 1, \\ \mathcal{T}(s_3, a_1, s_1) &= 1, \quad \mathcal{T}(s_1, a_2, s_2) = 1, \end{aligned}$$

all other values of $\mathcal{T}$ to zero, and $\gamma = 0.9$. We also set the inducing points to be equal to the three states in $\mathcal{S}$, add a single feature $f : \mathcal{S} \to \mathbb{R}$ such that

$$f(s_1) = 1, \quad f(s_2) = 2, \quad f(s_3) = 3,$$

and create two demonstrations $\zeta_1 = \{(s_1, a_1)\}$ and $\zeta_2 = \{(s_3, a_2)\}$ that correspond to moving from s_1 and s_3 to s_2. Therefore, we would expect the reward of s_2 to be higher than the other two rewards in order to reflect this.

Unfortunately, we are not able to use $\frac{\partial \mathcal{L}}{\partial \mathbf{B}}$ in order to optimise $\mathbf{B}$. We illustrate the problem in Figure 3. On the left side of the figure, we plot how $q(\mathbf{u})$ behaves as a function of a diagonal and a non-diagonal element of $\mathbf{B}$. Both functions have maximum values that can be attained by following the corresponding derivatives. However, when these derivatives are used to estimate $\frac{\partial \mathbb{E}[v]}{\partial \mathbf{B}}$, the resulting derivatives no longer match their corresponding functions, although the functions themselves still have optimal values: at or below 1 for $B_{1,1}$ and at 0 for $B_{2,1}$. This leads us to consider two restrictions of the model in order to investigate its convergence behaviour:

- In **Scenario 1**, we remove $\frac{\partial \mathbb{E}[v]}{\partial \mathbf{B}}$ from $\frac{\partial \mathcal{L}}{\partial \mathbf{B}}$. This essentially optimises $\mathbf{B}$ to match $\mathbf{K}_{\mathbf{u},\mathbf{u}}$, because then $\frac{\partial \mathcal{L}}{\partial \mathbf{B}}$ becomes $-\frac{\partial}{\partial \mathbf{B}} D_{\mathrm{KL}}(q(\mathbf{u}) \parallel p(\mathbf{u}))$, i.e., we are optimising $\mathbf{B}$ to minimise the difference between the prior and the posterior of $\mathbf{u}$.
- In **Scenario 2**, we set $\mathbf{B} = \mathbf{I}_m$, and do not optimise it at all.

We plot how $\mathcal{L}$ as well as policies $\pi(a_1 \mid s_1)$, $\pi(a_2 \mid s_3)$, and $\pi(a_1 \mid s_2)$ converge over a number of iterations in Figure 4. The first two policies correspond to actions taken in the set of demonstrations $\mathcal{D}$, so we would expect to see their probabilities converge to values above 0.5. Note that due to the stochastic nature of our MDP model, we do not expect to see any probability reach exactly 1. The third policy, however, has no relevant data in $\mathcal{D}$, so the maximum causal entropy framework would put the probability at around 0.5.

As we ran the algorithm with two stopping conditions—300 iterations and the ℓ_1-norm of the change in parameter values being below 0.01—note that the algorithm terminated early in Scenario 2 but not in Scenario 1, although there is clear convergence in both cases. An important difference is that $\pi(a_1 \mid s_1)$ and $\pi(a_2 \mid s_3)$ converge closer and closer to 1 with the slightly-restricted model of Scenario 1, but stabilise at 0.6 in Scenario 2 due to an important part of the model being completely fixed. In both cases, $\pi(a_1 \mid s_2)$ converges to 0.5 as expected.

Figure 5 shows how the parameters of the model converge in both scenarios. The algorithm seems to converge fine in Scenario 2, but most of the parameters fail to stabilise in Scenario 1. Most importantly, the diagonal values of the $\mathbf{B}$ matrix diverge to positive infinity, leading to higher variance in ELBO estimates seen in Figure 4. The non-diagonal values, however, converge just fine. Both λ_0 and λ_1 also diverge to positive infinity, while the elements of $\boldsymbol{\mu}$ diverge, but in 'the right' direction: as μ_2 increases while μ_1 and μ_3 decrease, the policies in Figure 4 converge to their optimal values.

This leaves us with two models: one that converges to reasonable-but-suboptimal values, and one that diverges to infinite variance but also provides correct policies.

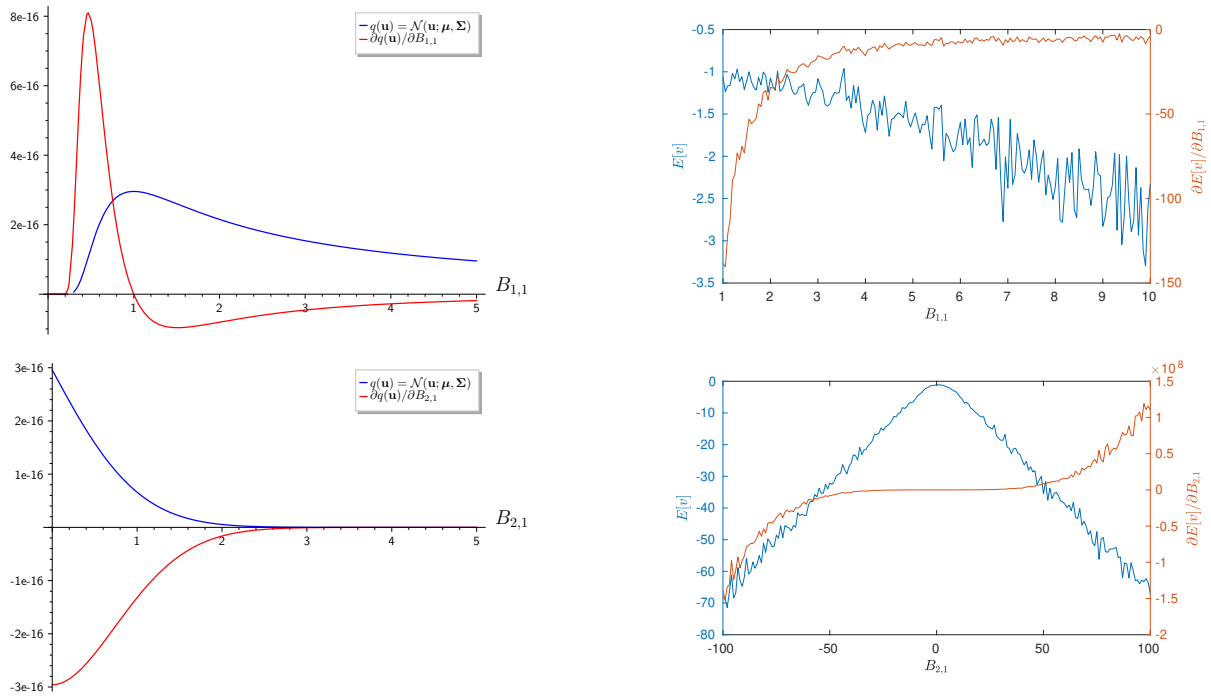

Figure 3: How a well-defined problem with a correct solutions becomes a well-defined problem with an incorrect solution. In each plot, the function we are trying to optimise is in blue, and its derivative is in red. The plots on the left are for $q(\mathbf{u})$ and its derivatives with respect to $B_{1,1}$ (at the top) and $B_{2,1}$ (at the bottom), whereas the plots on the right are for $\mathbb{E}[v]$ and its corresponding derivatives.

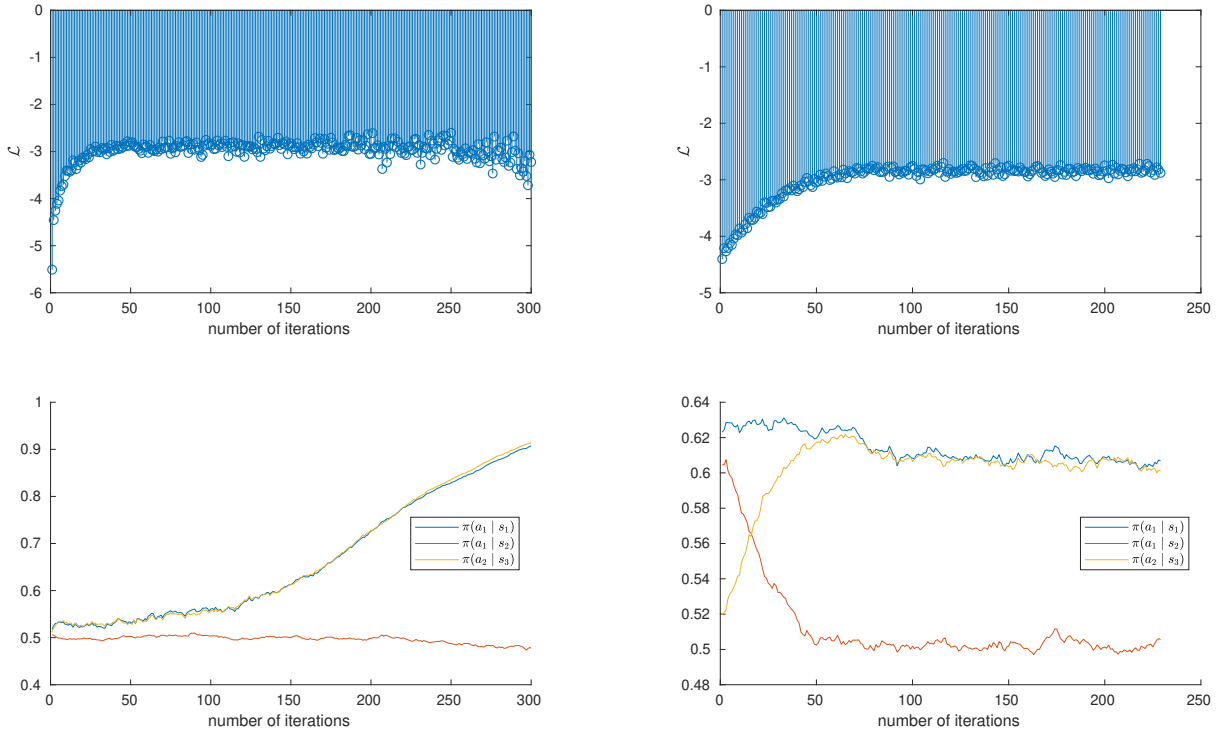

Figure 4: The convergence of $\mathcal{L}$ (at the top) and several example policies (at the bottom) over a number of iterations for Scenario 1 on the left and Scenario 2 on the right.

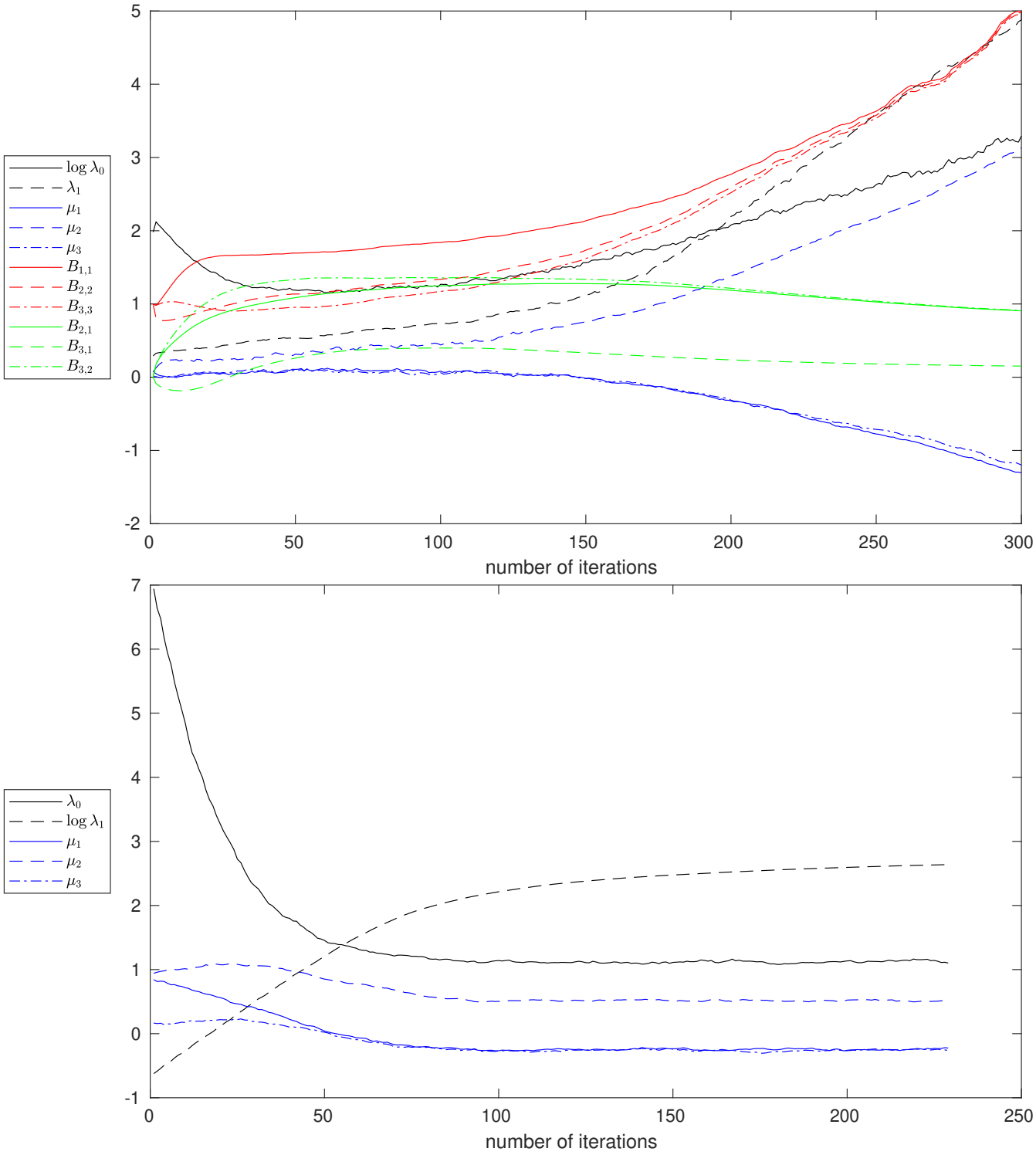

Figure 5: Convergence of all optimised parameters for Scenario 1 at the top and Scenario 2 at the bottom. In order to represent different variables on the same scale, some variables have been log-transformed. Colours denote which vector or matrix each scalar comes from: black for $\boldsymbol{\lambda}$, blue for $\boldsymbol{\mu}$, red for diagonal elements of B, and green for its non-diagonal elements.

7. RELATED WORK

As mentioned in the introduction, most IRL algorithms assume that the reward function can be represented as a linear combination of features. This assumption originated in one of the earliest papers on the topic by Ng and Russell [24], which introduced several linear programming approaches to the problem. The authors also noticed that often multiple reward functions can explain the same behaviour, and suggested heuristics for reward functions that are 'far away' from reward functions that do not fit the data.

A few years later, Abbeel and Ng [1] developed an algorithm for the same formulation of the problem with a guarantee to converge quickly. Neu and Szepesvári [23] identified a weakness in Abbeel and Ng's approach: the algorithm requires features to be 'appropriately' scaled, and optimal scaling may not be known. Instead, they suggest a way to combine IRL with *apprenticeship learning*, i.e., a supervised learning task for optimal *policy* recovery (whereas IRL focuses on recovering the reward function).

Ramachandran and Amir [30] were the first to formulate IRL in terms of Bayesian learning. While the model is easily interpretable and able to handle experts that make mistakes, the algorithm can only handle small state spaces and requires Monte Carlo Markov Chain (MCMC) sampling for inference.

Ziebart et al. [42] keep the linearity assumption, but introduce an influential idea: resolving the ambiguity when multiple reward functions explain the data by appealing to the maximum entropy principle.

Choi and Kim [8] extend the Bayesian model to learn good features as well as the reward function, trying to overcome the limitation of linearity. However, the approach is quite limiting: all features are assumed to have Boolean values, and the algorithm simply learns their conjunctions.

Levine et al. [19] are the first to suggest a way to learn nonlinear reward functions without harsh restrictions on the problem domain by using GPs. We base our work primarily on their paper, and the weaknesses we hope to address have already been covered in the previous section. A recent extension to their work by Jin et al. [14] aims to harness the power of deep learning by using several layers of GPs, making the model less dependent on being provided good features. They also use VI, but with a few simplifying assumptions: deterministic training conditional for the reward vector, and fully independent training conditional for the latent state.

Finally, instead of using GPs to model nonlinear reward functions, one can use a neural network (NN), as demonstrated by Wulfmeier et al. [40]. Their approach benefits from constant time inference and the ability to learn complex features either from already-given features or even from raw data. The only disadvantage (as demonstrated in the paper) is that NNs converge slower than GPs (as measured by the number of demonstrations).

7.1 Variational Inference

Variational inference has seen a recent increase in interest among academics, with different approaches focusing on different goals: better time complexity, handling a wider variety of models, making approximations more accurate, and using more complex function approximation techniques (such as NNs) to infer local latent variable values without having to calculate them individually for each data point [41]. As our IRL model is based on a GP, we will begin by reviewing some of the VI approaches applied specifically to GP regression. Based on a recent review of scalable GPs [41], we will concentrate on *stochastic variational sparse approximations*, as they have achieved modelling accuracy close to that of the full GP with no approximations, with many methods providing a time complexity of $\mathcal{O}(m^3)$. Below we provide a short overview of various assumptions that have been used in approximating *sparse* GPs (i.e., GPs that use inducing points), following on a paper by Quiñonero-Candela and Rasmussen [29].

Table ?? further summarises some of the recent and/or influential GP approximation approaches that might be relevant to our situation. For many of the variables in our model (including the inducing points and the hyperparameters of the covariance function), we have three possible ways of handling them:

- Provide a full Bayesian treatment by defining a prior probability distribution. While this option can easily prevent overfitting, the choice of prior can be difficult to justify.
- Fix the value of the variable. We can then optimise the value outside of the main algorithm, e.g., by running the algorithm with a set of possible values and keeping the value that produces the best performance. This option can be slow if we are interested in finding the optimal value, but could be appropriate if the variable has little impact on the algorithm's performance.
- Treat the variable as a variational parameter. This method will efficiently optimise the value of the variable, but is vulnerable to overfitting. Furthermore, this requires us to find the derivative of the ELBO w.r.t. the variable, which could be difficult.

Titsias [37] is the first to suggest a variational approximation where inducing points are treated as variational parameters. The paper has influenced many of the later works covered in this section, and has been reformulated by Gal et al. [10] into a distributed algorithm. We will fix the inducing points for the baseline version of our algorithm, but keep the idea in mind for possible future work.

Hensman et al. [12] reduce the complexity from $\mathcal{O}(nm^2)$ to $\mathcal{O}(m^3)$ using a stochastic variational inference (SVI) approach. While results such as efficiently computable natural gradients and analytically tractable optimal solutions for variational parameters are unlikely to transfer to a different problem domain, the overall SVI framework and an approximating Gaussian distribution for $\mathbf{u}$ will play an important role in our proposal.

Cheng and Boots [7] suggest using different bases in the reproducing kernel Hilbert space for the mean and covariance functions. The paper makes few assumptions and mostly relies on the work of Titsias [37], making it seem transferable to a new domain. However, applying the idea to IRL is likely to be beyond the scope of the project.

The variational Fourier features (VFF) algorithm by Hensman et al. [11] is only defined for Matérn kernels, which is likely to be too restrictive for our situation (the original paper on using GPs for IRL [19] used the automatic relevance detection kernel that has weights controlling how important each feature is). While extending VFF to support a flexible class of kernels defined by Wilson and Adams [39] is an

interesting and promising avenue of work, it is likely to be beyond the scope of this project as well.

While Peng et al. [26] provide a highly efficient distributed implementation, the derivation of the ELBO relies primarily on the fact that the evidence for a GP is a Gaussian, while in our case the evidence is anything but. Hence, it is unlikely that the ideas from this paper could be applicable to our IRL model.

Finally, since we expect $p(\mathbf{u}, \mathbf{r} \mid \mathcal{D})$ to be highly irregular, we would like our approximation to be capable of representing a wide range of possible probability distributions. The primary way of representing complex posteriors in VI is by using *normalising flows*, i.e., a collection of invertible functions—parametrised by additional variational parameters—that are applied to latent variables [33]. Normalising flows, along with some of the ideas mentioned previously, could provide great benefits to the GP IRL VI model. However, we are forced to keep any non-essential features as possible future work in order to make the project feasible in a given time frame. Furthermore, note that due to various differences between regression and IRL, we cannot simply apply a GP VI model as a whole. Instead, smaller tricks and ideas from various papers can (and will) be used throughout the project.

8. CONCLUSIONS

Reasonable results in convergence. More variables/information. Fewer assumptions.

We show how to avoid the deterministic training conditional assumption.

8.1 Further Work

An interesting extension to our work would be to consider IRL in the context of a reinforcement learning (RL) agent. Suppose we have an agent whose purpose is to learn optimal behaviour from observing other agents using IRL. It could then take reward variance estimates into account when choosing what states to visit next. It would have to handle the balance between exploration and exploitation similarly to many RL agents, but the information about rewards would come from observing (presumably near-optimal) behaviour exhibited by other agents rather than directly from the environment.

It is also worth noting the approach presented in this paper requires solving S MDPs for every iteration of optimising the parameters (where S is the number of samples drawn from $q(\mathbf{u}, \mathbf{r})$). There are at least two ways to reduce or eliminate this performance bottleneck:

- The MDP value function could be approximated, allowing for some minor mistakes in the resulting policy.
- Perhaps there is a good way to use information about previously computed value functions for similar rewards to hasten the current computation. One simple way to do this would be by initialising the current values to the optimal values of the previous MDP value function computation.

9. REFERENCES

[1] P. Abbeel and A. Y. Ng. Apprenticeship learning via inverse reinforcement learning. In C. E. Brodley, editor, *Machine Learning, Proceedings of the Twenty-first International Conference (ICML 2004), Banff, Alberta, Canada, July 4-8, 2004*, volume 69 of *ACM International Conference Proceeding Series*. ACM, 2004.

[2] S. Arora and P. Doshi. A survey of inverse reinforcement learning: Challenges, methods and progress. *CoRR*, abs/1806.06877, 2018.

[3] C. M. Bishop. *Pattern recognition and machine learning, 5th Edition*. Information science and statistics. Springer, 2007.

[4] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017.

[5] K. D. Bogert and P. Doshi. Multi-robot inverse reinforcement learning under occlusion with estimation of state transitions. *Artif. Intell.*, 263:46–73, 2018.

[6] R. Chen. The dominated convergence theorem and applications. National Cheng Kung University, 2016.

[7] C. Cheng and B. Boots. Variational inference for Gaussian process models with linear complexity. In I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 5190–5200, 2017.

[8] J. Choi and K. Kim. Bayesian nonparametric feature construction for inverse reinforcement learning. In F. Rossi, editor, *IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence, Beijing, China, August 3-9, 2013*, pages 1287–1293. IJCAI/AAAI, 2013.

[9] J. Duchi. Derivations for linear algebra and optimization. Stanford University.

[10] Y. Gal, M. van der Wilk, and C. E. Rasmussen. Distributed variational inference in sparse Gaussian process regression and latent variable models. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3257–3265, 2014.

[11] J. Hensman, N. Durrande, and A. Solin. Variational Fourier features for Gaussian processes. *Journal of Machine Learning Research*, 18:151:1–151:52, 2017.

[12] J. Hensman, N. Fusi, and N. D. Lawrence. Gaussian processes for big data. In A. Nicholson and P. Smyth, editors, *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence, UAI 2013, Bellevue, WA, USA, August 11-15, 2013*. AUAI Press, 2013.

[13] H. Herrlich. *Axiom of choice*. Springer, 2006.

[14] M. Jin, A. C. Damianou, P. Abbeel, and C. J. Spanos. Inverse reinforcement learning via deep Gaussian process. In G. Elidan, K. Kersting, and A. T. Ihler, editors, *Proceedings of the Thirty-Third Conference on Uncertainty in Artificial Intelligence, UAI 2017, Sydney, Australia, August 11-15, 2017*. AUAI Press, 2017.

[15] B. Kim and J. Pineau. Socially adaptive path planning

in human environments using inverse reinforcement learning. *I. J. Social Robotics*, 8(1):51–66, 2016.

[16] H. Kretzschmar, M. Spies, C. Sprunk, and W. Burgard. Socially compliant mobile robot navigation via inverse reinforcement learning. *I. J. Robotics Res.*, 35(11):1289–1307, 2016.

[17] S. Laue, M. Mitterreiter, and J. Giesen. Computing higher order derivatives of matrix and tensor expressions. In S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada.*, pages 2755–2764, 2018.

[18] W. Layton and M. Sussman. *Numerical linear algebra*. Lulu.com, 2014.

[19] S. Levine, Z. Popovic, and V. Koltun. Nonlinear inverse reinforcement learning with Gaussian processes. In J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. C. N. Pereira, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011. Proceedings of a meeting held 12-14 December 2011, Granada, Spain.*, pages 19–27, 2011.

[20] S. Levine, Z. Popovic, and V. Koltun. Supplementary material: Nonlinear inverse reinforcement learning with Gaussian processes. `http://graphics.stanford.edu/projects/gpirl/gpirl_supplement.pdf`, December 2011.

[21] H. Liu, Y. Ong, X. Shen, and J. Cai. When Gaussian process meets big data: A review of scalable GPs. *CoRR*, abs/1807.01065, 2018.

[22] R. Neal. *Bayesian Learning for Neural Networks*. Lecture Notes in Statistics. Springer New York, 2012.

[23] G. Neu and C. Szepesvári. Apprenticeship learning using inverse reinforcement learning and gradient methods. In R. Parr and L. C. van der Gaag, editors, *UAI 2007, Proceedings of the Twenty-Third Conference on Uncertainty in Artificial Intelligence, Vancouver, BC, Canada, July 19-22, 2007*, pages 295–302. AUAI Press, 2007.

[24] A. Y. Ng and S. J. Russell. Algorithms for inverse reinforcement learning. In P. Langley, editor, *Proceedings of the Seventeenth International Conference on Machine Learning (ICML 2000), Stanford University, Stanford, CA, USA, June 29 - July 2, 2000*, pages 663–670. Morgan Kaufmann, 2000.

[25] V. M.-H. Ong, D. J. Nott, and M. S. Smith. Gaussian variational approximation with a factor covariance structure. *Journal of Computational and Graphical Statistics*, 27(3):465–478, 2018.

[26] H. Peng, S. Zhe, X. Zhang, and Y. Qi. Asynchronous distributed variational Gaussian process for regression. In D. Precup and Y. W. Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 2788–2797. PMLR, 2017.

[27] K. B. Petersen, M. S. Pedersen, et al. The matrix cookbook. *Technical University of Denmark*, 7(15):510, 2008.

[28] Q. Qiao and P. A. Beling. Inverse reinforcement learning with Gaussian process. *CoRR*, abs/1208.2112, 2012.

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

[30] D. Ramachandran and E. Amir. Bayesian inverse reinforcement learning. In M. M. Veloso, editor, *IJCAI 2007, Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, January 6-12, 2007*, pages 2586–2591, 2007.

[31] R. Ranganath, S. Gerrish, and D. M. Blei. Black box variational inference. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics, AISTATS 2014, Reykjavik, Iceland, April 22-25, 2014*, volume 33 of *JMLR Workshop and Conference Proceedings*, pages 814–822. JMLR.org, 2014.

[32] C. E. Rasmussen and C. K. I. Williams. *Gaussian processes for machine learning*. Adaptive computation and machine learning. MIT Press, 2006.

[33] D. J. Rezende and S. Mohamed. Variational inference with normalizing flows. In F. R. Bach and D. M. Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 1530–1538. JMLR.org, 2015.

[34] H. Royden and P. Fitzpatrick. *Real Analysis*. Prentice Hall, 2010.

[35] S. J. Russell. Learning agents for uncertain environments (extended abstract). In P. L. Bartlett and Y. Mansour, editors, *Proceedings of the Eleventh Annual Conference on Computational Learning Theory, COLT 1998, Madison, Wisconsin, USA, July 24-26, 1998.*, pages 101–103. ACM, 1998.

[36] S. J. Russell and P. Norvig. *Artificial Intelligence - A Modern Approach (3. internat. ed.)*. Pearson Education, 2010.

[37] M. K. Titsias. Variational learning of inducing variables in sparse Gaussian processes. In D. A. V. Dyk and M. Welling, editors, *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics, AISTATS 2009, Clearwater Beach, Florida, USA, April 16-18, 2009*, volume 5 of *JMLR Proceedings*, pages 567–574. JMLR.org, 2009.

[38] A. Vogel, D. Ramachandran, R. Gupta, and A. Raux. Improving hybrid vehicle fuel efficiency using inverse reinforcement learning. In J. Hoffmann and B. Selman, editors, *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, July 22-26, 2012, Toronto, Ontario, Canada.* AAAI Press, 2012.

[39] A. Wilson and R. Adams. Gaussian process kernels for pattern discovery and extrapolation. In S. Dasgupta and D. McAllester, editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 1067–1075, Atlanta, Georgia, USA, 17–19 Jun 2013. PMLR.

[40] M. Wulfmeier, P. Ondruska, and I. Posner. Maximum

entropy deep inverse reinforcement learning. *arXiv preprint arXiv:1507.04888*, 2015.

[41] C. Zhang, J. Bütepage, H. Kjellström, and S. Mandt. Advances in variational inference. *CoRR*, abs/1711.05597, 2017.

[42] B. D. Ziebart, A. L. Maas, J. A. Bagnell, and A. K. Dey. Maximum entropy inverse reinforcement learning. In *AAAI*, volume 8, pages 1433–1438. Chicago, IL, USA, 2008.

[43] B. D. Ziebart, A. L. Maas, A. K. Dey, and J. A. Bagnell. Navigate like a cabbie: probabilistic reasoning from observed context-aware behavior. In H. Y. Youn and W. Cho, editors, *UbiComp 2008: Ubiquitous Computing, 10th International Conference, UbiComp 2008, Seoul, Korea, September 21-24, 2008, Proceedings*, volume 344 of *ACM International Conference Proceeding Series*, pages 322–331. ACM, 2008.

[44] B. D. Ziebart, N. D. Ratliff, G. Gallagher, C. Mertz, K. M. Peterson, J. A. Bagnell, M. Hebert, A. K. Dey, and S. S. Srinivasa. Planning-based prediction for pedestrians. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, October 11-15, 2009, St. Louis, MO, USA*, pages 3931–3936. IEEE, 2009.

APPENDIX

A. PROOFS

LEMMA 5.1 (DERIVATIVES OF PDFS).

1. $\frac{\partial q(\mathbf{u})}{\partial \boldsymbol{\mu}} = \frac{1}{2} q(\mathbf{u})(\boldsymbol{\Sigma}^{-1} + \boldsymbol{\Sigma}^{-\intercal})(\mathbf{u} - \boldsymbol{\mu})$.
2. (a) $\frac{\partial q(\mathbf{u})}{\partial \boldsymbol{\Sigma}} = \frac{1}{2} q(\mathbf{u})(\boldsymbol{\Sigma}^{-1}\mathbf{U}\boldsymbol{\Sigma}^{-1} - \boldsymbol{\Sigma}^{-1})$.
   (b) $\frac{\partial q(\mathbf{u})}{\partial \mathbf{B}} = q(\mathbf{u})(\boldsymbol{\Sigma}^{-1}\mathbf{U}\boldsymbol{\Sigma}^{-1} - \boldsymbol{\Sigma}^{-1})\mathbf{B}$.
3. *For* $i = 0, \ldots, d$,
   (a)
$$\frac{\partial q(\mathbf{r} \mid \mathbf{u})}{\partial \lambda_i} = \frac{1}{2} q(\mathbf{r} \mid \mathbf{u})(|\boldsymbol{\Gamma}|^{-1} \operatorname{tr}(\mathbf{R} \operatorname{adj}(\boldsymbol{\Gamma})) - (\mathbf{r} - \mathbf{S}\mathbf{u})^\intercal \boldsymbol{\Gamma}^{-1} \mathbf{R} \boldsymbol{\Gamma}^{-1} (\mathbf{r} - \mathbf{S}\mathbf{u})).$$
   (b) *For any covariance matrix* $\mathbf{K}$,
$$\frac{\partial \mathbf{K}}{\partial \lambda_i} = \begin{cases} \frac{1}{\lambda_i}\mathbf{K} & \text{if } i = 0, \\ \mathbf{L} & \text{otherwise,} \end{cases}$$
   where
$$L_{j,k} = k(\mathbf{x}_j, \mathbf{x}_k) \left(-\frac{1}{2}(x_{j,i} - x_{k,i})^2 - \mathbb{1}[j \neq k]\sigma^2 \right).$$

PROOF.

1.
$$\begin{aligned} \frac{\partial q(\mathbf{u})}{\partial m} &= q(\mathbf{u}) \frac{\partial}{\partial \boldsymbol{\mu}} \left[-\frac{Q}{2} \right] \\ &= -\frac{1}{2} q(\mathbf{u})(\boldsymbol{\Sigma}^{-1} + \boldsymbol{\Sigma}^{-\intercal})(\mathbf{u} - \boldsymbol{\mu}) \frac{\partial}{\partial \boldsymbol{\mu}} [\mathbf{u} - \boldsymbol{\mu}] \\ &= \frac{1}{2} q(\mathbf{u})(\boldsymbol{\Sigma}^{-1} + \boldsymbol{\Sigma}^{-\intercal})(\mathbf{u} - \boldsymbol{\mu}). \end{aligned}$$

2. An online tool by Laue et al.[6] [17] can be used to find both derivatives.

[6] `http://www.matrixcalculus.org/`

3. (a) Since
$$\begin{aligned} q(\mathbf{r} \mid \mathbf{u}) &= \mathcal{N}(\mathbf{r}; \mathbf{K}_{\mathbf{r},\mathbf{u}}^\intercal \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} \mathbf{u}, \mathbf{K}_{\mathbf{r},\mathbf{r}} - \mathbf{K}_{\mathbf{r},\mathbf{u}}^\intercal \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} \mathbf{K}_{\mathbf{r},\mathbf{u}}) \\ &= \mathcal{N}(\mathbf{r}; \mathbf{S}\mathbf{u}, \boldsymbol{\Gamma}), \end{aligned}$$
   we have
$$\frac{\partial q(\mathbf{r} \mid \mathbf{u})}{\partial \lambda_i} = -\frac{1}{2} q(\mathbf{r} \mid \mathbf{u}) \frac{\partial}{\partial \lambda_i} [(\mathbf{r} - \mathbf{S}\mathbf{u})^\intercal \boldsymbol{\Gamma}^{-1} (\mathbf{r} - \mathbf{S}\mathbf{u}) + \log |\boldsymbol{\Gamma}|].$$
   The same online tool can be used to show that
$$\frac{\partial}{\partial \lambda_i} \log |\boldsymbol{\Gamma}| = -|\boldsymbol{\Gamma}|^{-1} \operatorname{tr}(\mathbf{R} \operatorname{adj}(\boldsymbol{\Gamma})),$$
   and
$$\frac{\partial}{\partial \lambda_i} \boldsymbol{\Gamma}^{-1} = \boldsymbol{\Gamma}^{-1} \mathbf{R} \boldsymbol{\Gamma}^{-1}.$$
   (b) If $i = 0$, then
$$\frac{\partial \mathbf{K}}{\partial \lambda_i} = \frac{1}{\lambda_i} \mathbf{K}$$
   by the structure of each element of $\mathbf{K}$. If $i \neq 0$, then each element of $\frac{\partial \mathbf{K}}{\partial \lambda_i}$ is
$$\begin{aligned} L_{j,k} &= \frac{\partial k(\mathbf{x}_j, \mathbf{x}_k)}{\partial \lambda_i} \\ &= k(\mathbf{x}_j, \mathbf{x}_k) \frac{\partial}{\partial \lambda_i} \left[-\frac{1}{2}(\mathbf{x}_j - \mathbf{x}_k)^\intercal \boldsymbol{\Lambda} (\mathbf{x}_j - \mathbf{x}_k) - \mathbb{1}[j \neq k]\sigma^2 \operatorname{tr}(\boldsymbol{\Lambda}) \right] \\ &= k(\mathbf{x}_j, \mathbf{x}_k) \frac{\partial}{\partial \lambda_i} \left[-\frac{1}{2} \sum_{l=1}^{d} \lambda_l (x_{j,l} - x_{k,l})^2 - \mathbb{1}[j \neq k]\sigma^2 \sum_{l=1}^{d} \lambda_l \right] \\ &= k(\mathbf{x}_j, \mathbf{x}_k) \left(-\frac{1}{2}(x_{j,i} - x_{k,i})^2 - \mathbb{1}[j \neq k]\sigma^2 \right). \end{aligned}$$

□

B. DERIVATIVES OF THE ELBO

B.1 $\partial / \partial \boldsymbol{\mu}$

We begin by removing terms independent of $\boldsymbol{\mu}$:
$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\mu}} = \frac{\partial}{\partial \boldsymbol{\mu}} [\mathbf{t}^\intercal \mathbf{K}_{\mathbf{r},\mathbf{u}}^\intercal \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} \boldsymbol{\mu}] - \frac{1}{2} \frac{\partial}{\partial \boldsymbol{\mu}} \left[\boldsymbol{\mu}^\intercal \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} \boldsymbol{\mu} \right] - \frac{\partial}{\partial \boldsymbol{\mu}} \mathbb{E}[v].$$
Here
$$\frac{\partial}{\partial \boldsymbol{\mu}} \left[\boldsymbol{\mu}^\intercal \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} \boldsymbol{\mu} \right] = (\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} + \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-\intercal}) \boldsymbol{\mu}$$
by Petersen and Pedersen [27], and
$$\begin{aligned} \frac{\partial}{\partial \boldsymbol{\mu}} \mathbb{E}[V_{\mathbf{r}}(s)] &= \frac{\partial}{\partial \boldsymbol{\mu}} \iint V_{\mathbf{r}}(s) q(\mathbf{r} \mid \mathbf{u}) q(\mathbf{u}) \, d\mathbf{r} \, d\mathbf{u} \\ &= \iint V_{\mathbf{r}}(s) q(\mathbf{r} \mid \mathbf{u}) \frac{\partial q(\mathbf{u})}{\partial \boldsymbol{\mu}} \, d\mathbf{r} \, d\mathbf{u} \\ &= \frac{1}{2} \mathbb{E}[V_{\mathbf{r}}(s)(\boldsymbol{\Sigma}^{-1} + \boldsymbol{\Sigma}^{-\intercal})(\mathbf{u} - \boldsymbol{\mu})] \end{aligned}$$
by Theorem 5.10 and Lemma 5.1. Hence,
$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \boldsymbol{\mu}} = \mathbf{t}^\intercal \mathbf{K}_{\mathbf{r},\mathbf{u}}^\intercal \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} - \frac{1}{2}(\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} + \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-\intercal}) \boldsymbol{\mu} \\ - \frac{1}{2} \mathbb{E} \left[(\boldsymbol{\Sigma}^{-1} + \boldsymbol{\Sigma}^{-\intercal})(\mathbf{u} - \boldsymbol{\mu}) v \right]. \end{aligned}$$

B.2 $\partial/\partial\mathbf{B}$

$$\frac{\partial\mathcal{L}}{\partial\mathbf{B}} = \frac{1}{2}\left(\frac{\partial}{\partial\mathbf{B}}\log|\boldsymbol{\Sigma}| - \frac{\partial}{\partial\mathbf{B}}\operatorname{tr}\left(\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\boldsymbol{\Sigma}\right)\right) - \frac{\partial}{\partial\mathbf{B}}\mathbb{E}[v].$$

By Theorem 5.10,

$$\frac{\partial}{\partial\mathbf{B}}\mathbb{E}[V_{\mathbf{r}}(s)] = \iint V_{\mathbf{r}}(s)q(\mathbf{r}\mid\mathbf{u})\frac{\partial q(\mathbf{u})}{\partial\mathbf{B}}\,d\mathbf{r}\,d\mathbf{u}.$$

Then, using the aforementioned tool by Laue et al. [17], we get

$$\frac{\partial}{\partial\mathbf{B}}\log|\boldsymbol{\Sigma}| = 2\boldsymbol{\Sigma}^{-1}\mathbf{B}, \quad \frac{\partial}{\partial\mathbf{B}}\operatorname{tr}\left(\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\boldsymbol{\Sigma}\right) = 2\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{B},$$

and Lemma 5.1 gives

$$\frac{\partial q(\mathbf{u})}{\partial\mathbf{B}} = q(\mathbf{u})(\boldsymbol{\Sigma}^{-1}\mathbf{U}\boldsymbol{\Sigma}^{-1} - |\boldsymbol{\Sigma}|^{-1}\operatorname{adj}(\boldsymbol{\Sigma}))\mathbf{B}.$$

Therefore,

$$\frac{\partial\mathcal{L}}{\partial\mathbf{B}} = \left(\boldsymbol{\Sigma}^{-1} - \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\right)\mathbf{B} - \mathbb{E}[(\boldsymbol{\Sigma}^{-1}\mathbf{U}\boldsymbol{\Sigma}^{-1} - |\boldsymbol{\Sigma}|^{-1}\operatorname{adj}(\boldsymbol{\Sigma}))\mathbf{B}v].$$

B.3 $\partial/\partial\lambda_j$

For $j = 0, \ldots, d$,

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial\lambda_j} &= \mathbf{t}^{\intercal}\frac{\partial}{\partial\lambda_j}\left[\mathbf{K}_{\mathbf{r},\mathbf{u}}^{\intercal}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\right]\boldsymbol{\mu} - \frac{\partial}{\partial\lambda_j}\mathbb{E}[v] \\ &\quad - \frac{1}{2}\left(\frac{\partial}{\partial\lambda_j}\operatorname{tr}\left(\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\boldsymbol{\Sigma}\right) + \boldsymbol{\mu}^{\intercal}\frac{\partial\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}}{\partial\lambda_j}\boldsymbol{\mu} + \frac{\partial}{\partial\lambda_j}\log|\mathbf{K}_{\mathbf{u},\mathbf{u}}|\right),\end{aligned}$$

where

$$\begin{aligned}\frac{\partial\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}}{\partial\lambda_j} &= -\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\frac{\partial\mathbf{K}_{\mathbf{u},\mathbf{u}}}{\partial\lambda_j}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}, \\ \frac{\partial}{\partial\lambda_j}\left[\mathbf{K}_{\mathbf{r},\mathbf{u}}^{\intercal}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\right] &= \frac{\partial\mathbf{K}_{\mathbf{r},\mathbf{u}}^{\intercal}}{\partial\lambda_j}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} + \mathbf{K}_{\mathbf{r},\mathbf{u}}^{\intercal}\frac{\partial\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}}{\partial\lambda_j} \\ &= \left(\frac{\partial\mathbf{K}_{\mathbf{r},\mathbf{u}}^{\intercal}}{\partial\lambda_j} - \mathbf{K}_{\mathbf{r},\mathbf{u}}^{\intercal}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\frac{\partial\mathbf{K}_{\mathbf{u},\mathbf{u}}}{\partial\lambda_j}\right)\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}, \\ \frac{\partial}{\partial\lambda_j}\operatorname{tr}(\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\boldsymbol{\Sigma}) &= \operatorname{tr}\left(\frac{\partial}{\partial\lambda_j}\left[\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\boldsymbol{\Sigma}\right]\right) = \operatorname{tr}\left(\frac{\partial\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}}{\partial\lambda_j}\boldsymbol{\Sigma}\right) \\ &= -\operatorname{tr}\left(\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\frac{\partial\mathbf{K}_{\mathbf{u},\mathbf{u}}}{\partial\lambda_j}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\boldsymbol{\Sigma}\right), \\ \frac{\partial}{\partial\lambda_j}\log|\mathbf{K}_{\mathbf{u},\mathbf{u}}| &= \operatorname{tr}\left(\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\frac{\partial\mathbf{K}_{\mathbf{u},\mathbf{u}}}{\partial\lambda_j}\right)\end{aligned}$$

by Theorem 5.10 and Lemma 5.1. Thus,

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial\lambda_j} &= \mathbf{t}^{\intercal}\left(\frac{\partial\mathbf{K}_{\mathbf{r},\mathbf{u}}^{\intercal}}{\partial\lambda_j} - \mathbf{K}_{\mathbf{r},\mathbf{u}}^{\intercal}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\frac{\partial\mathbf{K}_{\mathbf{u},\mathbf{u}}}{\partial\lambda_j}\right)\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\boldsymbol{\mu} \\ &\quad + \frac{1}{2}\left[\operatorname{tr}\left(\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\frac{\partial\mathbf{K}_{\mathbf{u},\mathbf{u}}}{\partial\lambda_j}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\boldsymbol{\Sigma}\right) + \boldsymbol{\mu}^{\intercal}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\frac{\partial\mathbf{K}_{\mathbf{u},\mathbf{u}}}{\partial\lambda_j}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\boldsymbol{\mu}\right. \\ &\qquad \left. - \operatorname{tr}\left(\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\frac{\partial\mathbf{K}_{\mathbf{u},\mathbf{u}}}{\partial\lambda_j}\right)\right] \\ &\quad - \frac{1}{2}\mathbb{E}[(|\boldsymbol{\Gamma}|^{-1}\operatorname{tr}(\mathbf{R}\operatorname{adj}(\boldsymbol{\Gamma})) \\ &\qquad - (\mathbf{r}-\mathbf{S}\mathbf{u})^{\intercal}\boldsymbol{\Gamma}^{-1}\mathbf{R}\boldsymbol{\Gamma}^{-1}(\mathbf{r}-\mathbf{S}\mathbf{u}))v],\end{aligned}$$

by Petersen and Pedersen [27], and

$$\begin{aligned}\frac{\partial}{\partial\lambda_j}\mathbb{E}[V_{\mathbf{r}}(s)] &= \iint V_{\mathbf{r}}(s)\frac{\partial q(\mathbf{r}\mid\mathbf{u})}{\partial\lambda_j}q(\mathbf{u})\,d\mathbf{r}\,d\mathbf{u} \\ &= \frac{1}{2}\mathbb{E}[V_{\mathbf{r}}(s)(|\boldsymbol{\Gamma}|^{-1}\operatorname{tr}(\mathbf{R}\operatorname{adj}(\boldsymbol{\Gamma})) \\ &\qquad - (\mathbf{r}-\mathbf{S}\mathbf{u})^{\intercal}\boldsymbol{\Gamma}^{-1}\mathbf{R}\boldsymbol{\Gamma}^{-1}(\mathbf{r}-\mathbf{S}\mathbf{u}))]\end{aligned}$$

where the remaining derivatives can be found in Lemma 5.1.