# A variational approximate posterior for the deep Wishart process

**Anonymous Author(s)**
Affiliation
Address
email

## Abstract

Recent work introduced deep kernel processes as an entirely kernel-based alternative to NNs (Aitchison et al. 2020). Deep kernel processes flexibly learn good top-layer representations by alternately sampling the kernel from a distribution over positive semi-definite matrices and performing nonlinear transformations. A particular deep kernel process, the deep Wishart process (DWP), is of particular interest because its prior is equivalent to deep Gaussian process (DGP) priors. However, inference in DWPs has not yet been possible due to the lack of sufficiently flexible distributions over positive semi-definite matrices. Here, we give a novel approach to obtaining flexible distributions over positive semi-definite matrices by generalising the Bartlett decomposition of the Wishart probability density. We use this new distribution to develop an approximate posterior for the DWP that includes dependency across layers. We develop a doubly-stochastic inducing-point inference scheme for the DWP and show experimentally that inference in the DWP gives improved performance over doing inference in a DGP with the equivalent prior.

## 1 Introduction

The successes of modern deep learning have highlighted that good performance on tasks such as image classification (Krizhevsky et al., 2012) requires deep models with lower layers that have the flexibility to learn good representations. Up until very recently, this was only possible in feature-based methods such as neural networks (NNs). Kernel methods did not have this flexibility because the kernel could be modified only using a few kernel hyperparameters. However, with the advent of deep kernel processes (DKPs; Aitchison et al., 2020), we now have deep kernel methods that offer neural-network like flexibility in the kernel / top-layer representation. DKPs introduce this flexibility by taking the kernel from the previous layer, then sampling from a Wishart or inverse Wishart centered on that kernel, followed by a nonlinear transformation. The sampling and nonlinear transformation steps are repeated multiple times to form a deep architecture. Remarkably, deep Gaussian processes (DGPs; Damianou & Lawrence, 2013; Salimbeni & Deisenroth, 2017), standard Bayesian NNs, infinite-width Bayesian NNs (neural network Gaussian processes or NNGPs; Lee et al., 2017; Matthews et al., 2018; Novak et al., 2018; Garriga-Alonso et al., 2018) and infinite NNs with finite width bottlenecks (Aitchison, 2019) can be written as DKPs (Aitchison et al., 2020). In e.g. DGPs the random variables used in variational inference are ultimately features, even though a kernel is computed as a function of the features. By contrast, in a DKP, there are no features *at all*: the only random variables are the positive semi-definite kernel matrices themselves, which are sampled directly from distributions over positive semi-definite matrices such as the (inverse) Wishart.

Aitchison et al. (2020) argued that DKPs should have considerable advantages over related feature-based models, because feature-based models have pervasive symmetries in the true posterior, which

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

are difficult to capture in standard variational approximate posteriors. For instance, in a neural network, it is possible to permute rows and columns of weight matrices, such that the activations at a given layer are permuted, but the network's overall input-output function remains the same (MacKay, 1992; Sussmann, 1992; Bishop et al., 1995). These permutations result in networks with exactly the same output probability density under the true posterior, but with very different probability densities under standard variational approximate posteriors, which are generally unimodal. However, these issues do not arise with DKPs, because all permutations of the hidden units correspond to the same kernel (see Appendix D in Aitchison et al. (2020) for more details).

Deep Wishart processes (DWPs) are the most important DKP because their prior is equivalent to the DGP prior. While Aitchison et al. (2020) showed this equivalence, they were not able to do inference in DWPs because they were not able to find a sufficiently flexible distribution over positive semi-definite matrices to form the basis of an approximate posterior. Instead, they were forced to work with a different DKP: the deep *inverse* Wishart processes (DIWPs), which was easier because the inverse Wishart itself forms a suitable approximate posterior. We show how to create a sufficiently flexible approximate posterior for DWPs, thereby enabling us to compare directly to the equivalent DGPs. In particular, our contributions are:

- We develop a new family of flexible distributions over positive semi-definite matrices by generalising the Bartlett decomposition (Sec. 3.2).
- We use this distribution to develop an effective approximate posterior for the deep Wishart process which incorporates dependency across layers (Sec. 3.3).
- We develop a doubly stochastic inducing-point inference scheme for the DWP. While the derivation mostly follows that for deep inverse Wishart processes (Aitchison et al., 2020), we need to give a novel scheme for sampling the test/training points conditioned on the inducing points, as this is very different in the DWP compared to the previous DIWP (Sec. 3.4).
- We empirically compare DGP and DWP inference with the *exact same prior*. This was not possible in Aitchison et al. (2020) as they only derived an inference scheme for deep inverse Wishart processes, whose prior is not equivalent to a DGP prior. As expected, DWPs show strong benefits over DGPs.

# 2 Background

## 2.1 Wishart distribution

The Wishart, $\mathcal{W}(\mathbf{\Sigma}, \nu)$, is a distribution over positive semi-definite $P \times P$ matrices, $\mathbf{G}$, with positive definite scale parameter $\mathbf{\Sigma} \in \mathbb{R}^{P \times P}$ and a positive, integer-valued degrees-of-freedom parameter, $\nu$. The Wishart distribution is defined by taking $\nu$ vectors $\mathbf{n}_\lambda \in \mathbf{R}^P$ sampled from a zero-mean Gaussian. These vectors can be generated from standard Gaussian vectors, $\boldsymbol{\xi}_\lambda$, by transforming them with the Cholesky, $\mathbf{L}$ of the scale parameter, $\mathbf{\Sigma} = \mathbf{L}\mathbf{L}^T$,

$$\mathbf{L}\boldsymbol{\xi}_\lambda = \mathbf{n}_\lambda \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}) \qquad \text{where } \boldsymbol{\xi}_\lambda \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \tag{1}$$

Both $\mathbf{n}_\lambda$ and $\boldsymbol{\xi}_\lambda$ can be stacked to form $P \times \nu$ matrices, $\mathbf{N}$ and $\mathbf{\Xi}$,

$$\mathbf{N} = \begin{pmatrix} \mathbf{n}_1 & \mathbf{n}_2 & \cdots & \mathbf{n}_\nu \end{pmatrix} \qquad \mathbf{\Xi} = \begin{pmatrix} \boldsymbol{\xi}_1 & \boldsymbol{\xi}_2 & \cdots & \boldsymbol{\xi}_\nu \end{pmatrix}. \tag{2}$$

Wishart samples are defined by taking the sum of the outer products of the $\mathbf{n}_\lambda$'s, which can be written as a matrix-multiplication,

$$\sum_{\lambda=1}^{\nu} \mathbf{n}_\lambda \mathbf{n}_\lambda^T = \mathbf{N}\mathbf{N}^T = \mathbf{L}\mathbf{\Xi}\mathbf{\Xi}^T\mathbf{L} = \mathbf{L}\mathbf{Z}\mathbf{L}^T = \mathbf{G} \sim \mathcal{W}(\mathbf{\Sigma}, \nu) \tag{3}$$

where $\mathbf{Z} = \mathbf{\Xi}\mathbf{\Xi}^T$ is a sample from a standard Wishart (i.e. one with an identity scale parameter,)

$$\sum_{\lambda=1}^{\nu} \boldsymbol{\xi}_\lambda \boldsymbol{\xi}_\lambda^T = \mathbf{\Xi}\mathbf{\Xi}^T = \mathbf{Z} \sim \mathcal{W}(\mathbf{I}, \nu). \tag{4}$$

Note that therefore the Wishart has mean,

$$\mathbb{E}[\mathbf{G}] = \nu\, \mathbb{E}\left[\mathbf{n}_\lambda \mathbf{n}_\lambda^T\right] = \nu \mathbf{\Sigma} \tag{5}$$

### 2.2 Bartlett decomposition

However, sampling $\mathbf{\Xi}$ can be computationally expensive for very large values of $\nu$. Instead, it is possible to sample a Wishart by writing down the distribution over the Cholesky of $\mathbf{Z}$, denoted $\mathbf{A}$ (Bartlett, 1934). Taking $\mathbf{Z} = \mathbf{A}\mathbf{A}^T$, the distribution over $\mathbf{A}$ is,

$$\mathrm{P}\left(A_{jj}^2\right) = \text{Gamma}\left(A_{jj}^2; \alpha{=}\tfrac{\nu-j+1}{2}, \beta{=}\tfrac{1}{2}\right), \tag{6a}$$

$$\mathrm{P}\left(A_{j>k}\right) = \mathcal{N}\left(A_{jk}; 0, 1\right). \tag{6b}$$

i.e. the square of the on-diagonal elements are Gamma distributed and the off-diagonal elements are IID standard Gaussian.

### 2.3 Deep Gaussian proceses (DGPs)

In a DGP, we progressively sample features, $\mathbf{F}_\ell$, from a Gaussian process, conditioned on features from the previous layer,

$$\mathrm{P}\left(\mathbf{F}_\ell | \mathbf{F}_{\ell-1}\right) = \textstyle\prod_{\lambda=1}^{\nu_\ell} \mathcal{N}\left(\mathbf{f}_\lambda^\ell; \mathbf{0}, \mathbf{K}_{\text{features}}^\ell\left(\mathbf{F}_{\ell-1}\right)\right) \qquad \text{with } \mathbf{F}_0 = \mathbf{X}, \tag{7a}$$

$$\mathrm{P}\left(\mathbf{Y} | \mathbf{F}_{L+1}\right) = \textstyle\prod_{\lambda=1}^{\nu_{L+1}} \mathcal{N}\left(\mathbf{y}_\lambda; \mathbf{f}_\lambda^{L+1}, \sigma^2\mathbf{I}\right) \tag{7b}$$

where $\mathbf{X} \in \mathbb{R}^{P\times\nu_0}$ is the input and $\mathbf{F}_\ell \in \mathbb{R}^{P\times\nu_\ell}$ are the features. We use $P$ for the number of input points and $\nu_\ell$ for the width of layer $\ell$; thus $\nu_0$ is the number of inputs and $\nu_{L+1}$ is the number of outputs. In addition, the features and targets can be written as a stack of vectors, $\mathbf{f}_\lambda^\ell \in \mathbb{R}^P$ and $\mathbf{y}_\lambda \in \mathbb{R}^P$, i.e.

$$\mathbf{F}_\ell = \begin{pmatrix}\mathbf{f}_1^\ell & \mathbf{f}_2^\ell & \cdots & \mathbf{f}_{\nu_\ell}^\ell\end{pmatrix} \qquad \mathbf{Y} = \begin{pmatrix}\mathbf{y}_1 & \mathbf{y}_2 & \cdots & \mathbf{y}_{\nu_{L+1}}\end{pmatrix}. \tag{8}$$

The function $\mathbf{K}_{\text{features}}^\ell\left(\mathbf{F}_{\ell-1}\right)$ takes the features at the previous layer and returns the corresponding $P \times P$ kernel matrix. We mainly consider isotropic kernels such as the squared exponential, which can be written as a function of $R_{ij}^{\ell-1}$, the distance between input features $i$ and $j$,

$$K_{\text{features};ij}^\ell = k_\ell(R_{ij}^{\ell-1}), \tag{9}$$

$$R_{ij}^{\ell-1} = \tfrac{1}{N_\ell}\textstyle\sum_{\lambda=1}^{N_\ell}\left(F_{i\lambda}^{\ell-1} - F_{j\lambda}^{\ell-1}\right)^2. \tag{10}$$

### 2.4 Deriving equivalent deep Wishart processes

Following Aitchison et al. (2020), we show how the DGP model of Eq. (7) can be expressed as a deep Wishart process. We first consider the $P \times P$ Gram matrices defined as

$$\mathbf{G}_\ell = \tfrac{1}{\nu_\ell}\mathbf{F}_\ell\mathbf{F}_\ell^T = \tfrac{1}{\nu_\ell}\sum_{\lambda=1}^{\nu_\ell}\mathbf{f}_\lambda^\ell(\mathbf{f}_\lambda^\ell)^T, \tag{11}$$

where $\mathbf{f}_\lambda^\ell$ are IID and multivariate-Gaussian distributed conditioned on the features at the previous layer (Eq. 7a). Thus, $\mathbf{G}_\ell$ follows the definition of the Wishart (Eq. 3), and we can sample $\mathbf{G}_\ell$ directly,

$$\mathrm{P}\left(\mathbf{G}_\ell | \mathbf{F}_{\ell-1}\right) = \mathcal{W}\left(\mathbf{G}_\ell; \tfrac{1}{\nu_\ell}\mathbf{K}_{\text{features}}^\ell(\mathbf{F}_{\ell-1}), \nu_\ell\right). \tag{12}$$

To work entirely with Gram matrices rather than features, we need to be able to compute the kernel, $\mathbf{K}_{\text{features}}^\ell(\mathbf{F}_{\ell-1})$ as a function of the Gram matrix at the previous layer, $\mathbf{G}_{\ell-1}$. Remarkably, this is possible for a large family of practically relevant kernels (Aitchison et al., 2020). In particular, note that it is possible to recover the distance from the Gram matrix,

$$R_{ij}^\ell = \tfrac{1}{N_\ell}\textstyle\sum_{\lambda=1}^{N_\ell}\left(\left(F_{i\lambda}^\ell\right)^2 - 2F_{i\lambda}^\ell F_{j\lambda}^\ell + \left(F_{j\lambda}^\ell\right)^2\right) = G_{ii}^\ell - 2G_{ij}^\ell + G_{jj}^\ell. \tag{13}$$

Thus, at least for isotropic kernels, which depend only on the distance, it is possible to obtain $\mathbf{K}_\ell(\cdot)$, which takes the Gram matrix from the previous layer and returns the same kernel matrix as that returned by applying $\mathbf{K}_{\text{features}}$ to the features from the previous layer:

$$\mathbf{K}_{\text{features}}^\ell(\mathbf{F}_{\ell-1}) = \mathbf{K}_\ell(\mathbf{G}_{\ell-1}) = \mathbf{K}_\ell(\tfrac{1}{\nu_\ell}\mathbf{F}_{\ell-1}\mathbf{F}_{\ell-1}^T). \tag{14}$$

By using the equivalent kernel written as a function of the Gram matrix at the previous layer, we can entirely eliminate intermediate layer features, resulting in a deep Wishart process,

$$\mathrm{P}\left(\mathbf{G}_\ell|\mathbf{G}_{\ell-1}\right) = \mathcal{W}\left(\mathbf{G}_\ell; \tfrac{1}{\nu_\ell}\mathbf{K}_\ell(\mathbf{G}_{\ell-1}), \nu_\ell\right) \qquad \text{with } \mathbf{G}_0 = \tfrac{1}{\nu_0}\mathbf{X}\mathbf{X}^T, \tag{15a}$$

$$\mathrm{P}\left(\mathbf{F}_{L+1}|\mathbf{G}_L\right) = \textstyle\prod_{\lambda=1}^{\nu_{L+1}} \mathcal{N}\left(\mathbf{f}_\lambda^{L+1}; \mathbf{0}, \mathbf{K}_\ell\left(\mathbf{G}_L\right)\right), \tag{15b}$$

$$\mathrm{P}\left(\mathbf{Y}|\mathbf{F}_{L+1}\right) = \textstyle\prod_{\lambda=1}^{\nu_{L+1}} \mathcal{N}\left(\mathbf{y}_\lambda; \mathbf{0}, \sigma^2\mathbf{I}\right). \tag{15c}$$

### 2.5 DWP formulation captures true-posterior symmetries while DGP does not

We now have two equivalent generative models: one phrased in terms of features, $\mathbf{F}_\ell$ and another phrased in terms of Gram matrices, $\mathbf{G}_\ell$. Is there any reason to prefer one over the other? It turns out that there is. In particular, consider a transformation of the features, $\mathbf{F}'_\ell = \mathbf{U}\mathbf{F}_\ell$ where $\mathbf{U}$ is a unitary matrix, such that $\mathbf{U}\mathbf{U}^T = \mathbf{I}$. Remarkably, the true posterior is symmetric under these transformations, in the sense that all unitary transformations of the underlying features have the exact same true-posterior probability density (see Aitchison et al., 2020, Appendix D.2),

$$\mathrm{P}\left(\mathbf{F}'_1, \ldots, \mathbf{F}'_L, \mathbf{F}_{L+1}|\mathbf{X}, \mathbf{Y}\right) = \mathrm{P}\left(\mathbf{F}_1, \ldots, \mathbf{F}_L, \mathbf{F}_{L+1}|\mathbf{X}, \mathbf{Y}\right). \tag{16}$$

It would be desirable for variational approximate posteriors to capture these true posterior symmetries. However, the usual family of Gaussian approximate posteriors over features fails to capture these symmetries because they use non-zero means. Worryingly, the failure to capture these symmetries can bias variational inference to focus on low-mass areas of the true posterior (Moore, 2016; Pourzanjani et al., 2017).

In contrast, the deep Wishart process sidesteps this issue by phrasing posteriors entirely in terms of Gram matrices, $\mathbf{G}_\ell = \frac{1}{\nu_\ell}\mathbf{F}_\ell\mathbf{F}_\ell^T$. Critically, the Gram matrix is invariant to unitary transformations of the features,

$$\mathbf{G}_\ell = \tfrac{1}{N_\ell}\mathbf{F}_\ell\mathbf{F}_\ell^T = \tfrac{1}{N_\ell}\mathbf{F}_\ell\mathbf{U}_\ell\mathbf{U}_\ell^T\mathbf{F}_\ell^T = \tfrac{1}{N_\ell}\mathbf{F}'_\ell\mathbf{F}'^T_\ell. \tag{17}$$

As such, DWP approximate posteriors written in terms of $\mathbf{G}_\ell$ implicitly respect this unitary symmetry over the features.

## 3 Methods

As detailed in Aitchison et al. (2020), the key difficulty in obtaining a variational inference scheme for DWPs is the difficulty of providing a sufficiently flexible approximate posterior. In particular, as we are working with a probabilistic process, the number of input points, $P$, can be arbitrarily large, and thus there is always the possibility that $\nu < P$ and hence that our sampled Gram matrices are low-rank. We therefore need to form flexible variational approximate posteriors over rank $\nu$ Gram matrices. An obvious first choice is the Wishart distribution itself with degrees of freedom $\nu$, so as to match the rank of matrices sampled from the prior. However, for fixed degrees of freedom the Wishart variance,

$$\mathbb{V}\left[G_{ij}\right] = \nu\left(\Sigma_{ij}^2 + \Sigma_{ii}\Sigma_{jj}\right) \tag{18}$$

cannot be specified independently of the mean (Eq. 5), which is essential for a variational approximate posterior that can flexibly capture potentially narrow true posteriors. An alternative approach would be to work with a non-central Wishart, which is defined by taking $\mathbf{\Xi}$, which is IID standard Gaussian in the case of the Wishart, to have non-zero mean. However, the non-central Wishart has a probability density function that is too difficult to evaluate in the inner loop of a deep learning algorithm (Koev & Edelman, 2006). Instead, we develop a new Generalised Singular Wishart distribution, based on the Bartlett decomposition, which modifies the Wishart to give independent control over the mean and variance of sampled matrices.

### 3.1 Singular Bartlett decomposition

To define the Generalised Singular Wishart distribution, we first need to generalise the Wishart construction to potentially singular matrices (i.e. those for which $\nu < P$). Remembering that

$\mathbf{Z} = \mathbf{A}\mathbf{A}^T$, in the singular case $\mathbf{A}$ is given by

$$\mathbf{A} = \begin{pmatrix} A_{11} & \dots & 0 \\ \vdots & \ddots & \vdots \\ A_{\nu 1} & \dots & A_{\nu\nu} \\ \vdots & \vdots & \vdots \\ A_{P1} & \dots & A_{P\nu} \end{pmatrix}, \tag{19}$$

$$\mathrm{P}\left(A_{jj}^2\right) = \text{Gamma}\left(A_{jj}^2; \tfrac{\nu-j+1}{2}, \tfrac{1}{2}\right) \tag{20}$$

$$\mathrm{P}\left(A_{i>j}\right) = \mathcal{N}\left(A_{ij}; 0, 1\right). \tag{21}$$

The full probability density is

$$\mathrm{P}\left(\mathbf{A}\right) = \prod_{j=1}^{\min(p,\nu)} 2A_{jj}\text{Gamma}\left(A_{jj}^2; \tfrac{\nu-j+1}{2}, \tfrac{1}{2}\right) \prod_{i=j+1}^{p} \mathcal{N}\left(A_{ij}; 0, 1\right). \tag{22}$$

where the $2A_{jj}$ accounts for the Jacobian for the transformation from $A_{jj}^2$ to $A_{jj}$. Now we transform from $\mathbf{A}$ to $\mathbf{G}$ using the Jacobians in Appendix C and Appendix B (i.e. by first transforming $\mathbf{A} \to \mathbf{L}\mathbf{A}$ then transforming $\mathbf{L}\mathbf{A} \to \mathbf{L}\mathbf{A}\mathbf{A}^T\mathbf{L}^T$.)

$$\mathrm{P}\left(\mathbf{G}\right) = \left(\prod_{j=1}^{p} \frac{1}{L_{jj}^{\min(j,\nu)}}\right) \prod_{j=1}^{\min(p,\nu)} \frac{\text{Gamma}\left(A_{jj}^2; \frac{\nu-j+1}{2}, \frac{1}{2}\right)}{A_{jj}^{p-j} L_{jj}^{p-j+1}} \prod_{i=j+1}^{p} \mathcal{N}\left(A_{ij}; 0, 1\right). \tag{23}$$

In Appendix D we prove that this corresponds to the known full rank and singular Wishart distribution.

### 3.2 Generalised singular Wishart distributions

Our goal is to develop a generalisation of the Wishart distribution, $\mathcal{W}\left(\boldsymbol{\Sigma}, \nu, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\mu}, \boldsymbol{\sigma}\right)$ based on the Bartlett decomposition, which will turn out to have additional parameters, $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ for the on-diagonal elements of $\mathbf{A}$ and $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ for the off-diagonal elements. As we will be using this distribution for the approximate posterior, we write densities under this distribution as $\mathrm{Q}\left(\cdot\right)$,

$$\mathrm{Q}\left(\mathbf{G}\right) = \mathcal{W}\left(\mathbf{G}; \boldsymbol{\Sigma}, \nu, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\mu}, \boldsymbol{\sigma}\right). \tag{24}$$

To specify this distribution, we generalise the Bartlett decomposition using

$$\mathrm{Q}\left(A_{jj}^2\right) = \text{Gamma}\left(A_{jj}^2; \alpha_j, \beta_j\right) \qquad \text{for } j \leq \nu \tag{25a}$$

$$\mathrm{Q}\left(A_{i>j}\right) = \mathcal{N}\left(A_{ij}; \mu_{ij}, \sigma_{ij}^2\right). \tag{25b}$$

Thus, the full probability density for $\mathbf{A}$ is,

$$\mathrm{Q}\left(\mathbf{A}\right) = \prod_{j=1}^{\nu} 2A_{jj}\text{Gamma}\left(A_{jj}^2; \alpha_j, \beta_j\right) \prod_{i=j+1}^{p} \mathcal{N}\left(A_{ij}; \mu_{ij}, \sigma_{ij}^2\right), \tag{26}$$

applying the same transformations and Jacobians as in the previous section, this implies a distribution over $\mathbf{G}$ of,

$$\mathrm{Q}\left(\mathbf{G}\right) = \left(\prod_{j=1}^{p} \frac{1}{L_{jj}^{\min(j,\nu)}}\right) \prod_{j=1}^{\min(p,\nu)} \frac{1}{A_{jj}^{p-j} L_{jj}^{p-j+1}} \text{Gamma}\left(A_{jj}^2; \alpha_j, \beta_j\right) \prod_{i=j+1}^{p} \mathcal{N}\left(A_{ij}; \mu_{ij}, \sigma_{ij}^2\right). \tag{27}$$

### 3.3 Full approximate posterior distribution

Inspired by the across-layer dependencies in Aitchison et al. (2020), we use an Generalised Wishart approximate posterior for $\mathbf{G}_\ell$ with dependencies across layers,

$$\mathrm{Q}\left(\mathbf{G}_\ell | \mathbf{G}_{\ell-1}\right) = \mathcal{W}\left(\mathbf{G}_\ell; (1-p_\ell)\tfrac{1}{\nu_\ell}\mathbf{K}(\mathbf{G}_{\ell-1}) + p_\ell \mathbf{V}_\ell \mathbf{V}_\ell^T,\, \nu_\ell,\, \boldsymbol{\alpha}_\ell,\, \boldsymbol{\beta}_\ell,\, \boldsymbol{\mu}_\ell,\, \boldsymbol{\sigma}_\ell\right), \tag{28a}$$

where the approximate posterior parameters are $\{\mathbf{V}_\ell, \boldsymbol{\alpha}_\ell, \boldsymbol{\beta}_\ell, \boldsymbol{\mu}_\ell, \boldsymbol{\sigma}_\ell, p_\ell\}_{\ell=1}^{L}$, where $0 < p_\ell < 1$ is a scalar, and $\mathbf{V}_\ell \in \mathbb{R}^{p \times p}$. Note that the exact form for the across layer dependencies in our approximate posterior is inspired by the approximate posterior in (Aitchison et al., 2020).

### 3.4 Doubly stochastic inducing-point variational inference in deep inverse Wishart processes

For efficient inference in high-dimensional problems, we take inspiration from the DGP literature (Salimbeni & Deisenroth, 2017) by considering doubly-stochastic inducing-point deep Wishart processes. We begin by decomposing all variables into inducing and training (or test) points $\mathbf{X}_\mathrm{i} \in \mathbb{R}^{P_\mathrm{i} \times N_0}$ and $\mathbf{X}_\mathrm{t} \in \mathbb{R}^{P_\mathrm{t} \times N_0}$ where $P_\mathrm{i}$ is the number of inducing points, and $P_\mathrm{t}$ is the number of testing/training points,

$$\mathbf{X} = \begin{pmatrix} \mathbf{X}_\mathrm{i} \\ \mathbf{X}_\mathrm{t} \end{pmatrix}, \qquad \mathbf{F}_{L+1} = \begin{pmatrix} \mathbf{F}_\mathrm{i}^{L+1} \\ \mathbf{F}_\mathrm{t}^{L+1} \end{pmatrix}, \qquad \mathbf{G}_\ell = \begin{pmatrix} \mathbf{G}_\mathrm{ii}^\ell & \mathbf{G}_\mathrm{it}^\ell \\ \mathbf{G}_\mathrm{ti}^\ell & \mathbf{G}_\mathrm{tt}^\ell \end{pmatrix}, \tag{29}$$

where e.g. $\mathbf{G}_\mathrm{ii}^\ell$ is $P_\mathrm{i} \times P_\mathrm{i}$ and $\mathbf{G}_\mathrm{it}^\ell$ is $P_\mathrm{i} \times P_\mathrm{t}$. The full ELBO including latent variables for all the inducing and training points is

$$\mathcal{L} = \mathbb{E}\left[\log \mathrm{P}\left(\mathbf{Y}|\mathbf{F}_{L+1}\right) + \log \frac{\mathrm{P}\left(\{\mathbf{G}_\ell\}_{\ell=1}^L, \mathbf{F}_{L+1}|\mathbf{X}\right)}{\mathrm{Q}\left(\{\mathbf{G}_\ell\}_{\ell=1}^L, \mathbf{F}_{L+1}|\mathbf{X}\right)}\right], \tag{30}$$

where the expectation is taken over $\mathrm{Q}\left(\{\mathbf{G}_\ell\}_{\ell=1}^L, \mathbf{F}_{L+1}|\mathbf{X}\right)$. The prior is given by combining all terms in Eq. (15) for both inducing and test/train inputs,

$$\mathrm{P}\left(\{\mathbf{G}_\ell\}_{\ell=1}^L, \mathbf{F}_{L+1}|\mathbf{X}\right) = \left[\prod_{\ell=1}^L \mathrm{P}\left(\mathbf{G}_\ell|\mathbf{G}_{\ell-1}\right)\right] \mathrm{P}\left(\mathbf{F}_{L+1}|\mathbf{G}_L\right), \tag{31}$$

where the $\mathbf{X}$-dependence enters on the right because $\mathbf{G}_0 = \frac{1}{\nu_0}\mathbf{X}\mathbf{X}^T$. Taking inspiration from Salimbeni & Deisenroth (2017), the full approximate posterior is the product of an approximate posterior over inducing points and the conditional prior for train/test points,

$$\begin{aligned} &\mathrm{Q}\left(\{\mathbf{G}_\ell\}_{\ell=1}^L, \mathbf{F}_{L+1}|\mathbf{X}\right) = \\ &\qquad \mathrm{Q}\left(\{\mathbf{G}_\mathrm{ii}^\ell\}_{\ell=1}^L, \mathbf{F}_\mathrm{i}^{L+1}|\mathbf{X}_\mathrm{i}\right) \mathrm{P}\left(\{\mathbf{G}_\mathrm{it}^\ell\}_{\ell=1}^L, \{\mathbf{G}_\mathrm{tt}^\ell\}_{\ell=1}^L, \mathbf{F}_\mathrm{t}^{L+1}|\{\mathbf{G}_\mathrm{ii}^\ell\}_{\ell=1}^L, \mathbf{F}_\mathrm{i}^{L+1}, \mathbf{X}\right). \end{aligned} \tag{32}$$

And the prior can be written in the same form,

$$\begin{aligned} &\mathrm{P}\left(\{\mathbf{G}_\ell\}_{\ell=1}^L, \mathbf{F}_{L+1}|\mathbf{X}\right) = \\ &\qquad \mathrm{P}\left(\{\mathbf{G}_\mathrm{ii}^\ell\}_{\ell=1}^L, \mathbf{F}_\mathrm{i}^{L+1}|\mathbf{X}_\mathrm{i}\right) \mathrm{P}\left(\{\mathbf{G}_\mathrm{it}^\ell\}_{\ell=1}^L, \{\mathbf{G}_\mathrm{tt}^\ell\}_{\ell=1}^L, \mathbf{F}_\mathrm{t}^{L+1}|\{\mathbf{G}_\mathrm{ii}^\ell\}_{\ell=1}^L, \mathbf{F}_\mathrm{i}^{L+1}, \mathbf{X}\right). \end{aligned} \tag{33}$$

We discuss the second terms (the conditional prior) in Eq. (37). The first terms (the prior and approximate posteriors over inducing points), are given by combining terms in Eq. (15) and Eq. (28),

$$\mathrm{P}\left(\{\mathbf{G}_\mathrm{ii}^\ell\}_{\ell=1}^L, \mathbf{F}_\mathrm{i}^{L+1}|\mathbf{X}_\mathrm{i}\right) = \left[\prod_{\ell=1}^L \mathrm{P}\left(\mathbf{G}_\mathrm{ii}^\ell|\mathbf{G}_\mathrm{ii}^{\ell-1}\right)\right] \mathrm{P}\left(\mathbf{F}_\mathrm{i}^{L+1}|\mathbf{G}_\mathrm{ii}^L\right), \tag{34}$$

$$\mathrm{Q}\left(\{\mathbf{G}_\mathrm{ii}^\ell\}_{\ell=1}^L, \mathbf{F}_\mathrm{i}^{L+1}|\mathbf{X}_\mathrm{i}\right) = \left[\prod_{\ell=2}^L \mathrm{Q}\left(\mathbf{G}_\mathrm{ii}^\ell|\mathbf{G}_\mathrm{ii}^{\ell-1}\right)\right] \mathrm{Q}\left(\mathbf{F}_\mathrm{i}^{L+1}|\mathbf{G}_\mathrm{ii}^L\right). \tag{35}$$

Substituting Eqs. (32–35) into the ELBO (Eq. 30), the conditional prior cancels and we obtain,

$$\mathcal{L} = \mathbb{E}\left[\log \mathrm{P}\left(\mathbf{Y}|\mathbf{F}_\mathrm{t}^{L+1}\right) + \log \frac{\left[\prod_{\ell=1}^L \mathrm{Q}\left(\mathbf{G}_\mathrm{ii}^\ell|\mathbf{G}_\mathrm{ii}^{\ell-1}\right)\right] \mathrm{Q}\left(\mathbf{F}_\mathrm{i}^{L+1}|\mathbf{G}_\mathrm{ii}^L\right)}{\left[\prod_{\ell=1}^L \mathrm{P}\left(\mathbf{G}_\mathrm{ii}^\ell|\mathbf{G}_\mathrm{ii}^{\ell-1}\right)\right] \mathrm{P}\left(\mathbf{F}_\mathrm{i}^{L+1}|\mathbf{G}_\mathrm{ii}^L\right)}\right]. \tag{36}$$

The first term is a summation across test/train datapoints, and the second term depends only on the inducing points, so as in Salimbeni & Deisenroth (2017) we can compute unbiased estimates of the expectation by taking only a minibatch of datapoints. We also never need to compute the density of the conditional prior in Eq. (33), we only need to be able to sample from it,

$$\begin{aligned} &\mathrm{P}\left(\{\mathbf{G}_\mathrm{ti}^\ell, \mathbf{G}_\mathrm{tt}^\ell\}_{\ell=1}^L, \mathbf{F}_\mathrm{t}^{L+1}|\{\mathbf{G}_\mathrm{ii}^\ell\}_{\ell=1}^L, \mathbf{F}_\mathrm{i}^{L+1}, \mathbf{X}\right) = \\ &\qquad \mathrm{P}\left(\mathbf{F}_\mathrm{t}^{L+1}|\mathbf{F}_\mathrm{i}^{L+1}, \mathbf{G}_L\right) \prod_{\ell=1}^L \mathrm{P}\left(\mathbf{G}_\mathrm{ti}^\ell, \mathbf{G}_\mathrm{tt}^\ell|\mathbf{G}_\mathrm{ii}^\ell, \mathbf{G}_{\ell-1}\right). \end{aligned} \tag{37}$$

The first distribution, $\mathrm{P}\left(\mathbf{F}_\mathrm{t}^{L+1}|\mathbf{F}_\mathrm{i}^{L+1}, \mathbf{G}_L\right)$, is a multivariate Gaussian, and can be evaluated using methods from the GP literature (Williams & Rasmussen, 2006; Salimbeni & Deisenroth, 2017). Specifically, we use the global inducing point scheme from Ober & Aitchison (2020). The second

distribution is more difficult to sample from. To address this issue, we introduce sampled features $\hat{\mathbf{F}}_\ell$ (not to be confused with the features $\mathbf{F}_\ell$ in the corresponding DGP) such that

$$\hat{\mathbf{F}}_\ell \hat{\mathbf{F}}_\ell^T = \mathbf{G}_\ell \sim \mathcal{W}\left(\mathbf{\Sigma}, \nu\right), \tag{38}$$

with

$$\hat{\mathbf{F}}_\ell = \begin{pmatrix} \hat{\mathbf{F}}_{\mathrm{i}}^\ell \\ \hat{\mathbf{F}}_{\mathrm{t}}^\ell \end{pmatrix} \qquad \mathbf{\Sigma} = \begin{pmatrix} \mathbf{\Sigma}_{\mathrm{ii}} & \mathbf{\Sigma}_{\mathrm{ti}}^T \\ \mathbf{\Sigma}_{\mathrm{ti}} & \mathbf{\Sigma}_{\mathrm{tt}} \end{pmatrix} = \tfrac{1}{\nu}\mathbf{K}_\ell\left(\mathbf{G}_{\ell-1}\right), \tag{39}$$

where $\hat{\mathbf{F}}_\ell \in \mathbb{R}^{(P_{\mathrm{i}}+P_{\mathrm{t}})\times\nu_\ell}$, $\hat{\mathbf{F}}_{\mathrm{i}} \in \mathbb{R}^{P_{\mathrm{i}}\times\nu_\ell}$ and $\hat{\mathbf{F}}_{\mathrm{t}} \in \mathbb{R}^{P_{\mathrm{t}}\times\nu_\ell}$. Our goal is to sample $\mathbf{G}_{\mathrm{it}}^\ell$ and $\mathbf{G}_{\mathrm{tt}}^\ell$ given $\mathbf{G}_{\mathrm{ii}}^\ell$. Our approach is to note that, $\hat{\mathbf{F}}_{\mathrm{t}}$ conditioned on $\hat{\mathbf{F}}_{\mathrm{i}}$ is given by a matrix normal, (Eaton et al., 2007, page 310),

$$\mathrm{P}\left(\hat{\mathbf{F}}_{\mathrm{t}}^\ell \middle| \hat{\mathbf{F}}_{\mathrm{i}}^\ell\right) = \mathcal{MN}\left(\mathbf{\Sigma}_{\mathrm{ti}}^T\mathbf{\Sigma}_{\mathrm{ii}}^{-1}\hat{\mathbf{F}}_{\mathrm{i}}, \mathbf{\Sigma}_{\mathrm{tt}\cdot\mathrm{i}}, \mathbf{I}\right), \tag{40}$$

where

$$\mathbf{\Sigma}_{\mathrm{tt}\cdot\mathrm{i}} = \mathbf{\Sigma}_{\mathrm{tt}} - \mathbf{\Sigma}_{\mathrm{it}}^T\mathbf{\Sigma}_{\mathrm{ii}}^{-1}\mathbf{\Sigma}_{\mathrm{it}}. \tag{41}$$

Note that we sample each test/train point one-at-a-time/independently, in which case, $P_{\mathrm{t}} = 1$ and $\mathbf{\Sigma}_{22\cdot 1}$ is scalar.

Then $\mathbf{G}_\ell$, which includes $\mathbf{G}_{\mathrm{it}}^\ell$ and $\mathbf{G}_{\mathrm{tt}}^\ell$ is given by,

$$\mathbf{G}_\ell = \begin{pmatrix} \mathbf{G}_{\mathrm{ii}}^\ell & \mathbf{G}_{\mathrm{it}}^\ell \\ \mathbf{G}_{\mathrm{ti}}^\ell & \mathbf{G}_{\mathrm{tt}}^\ell \end{pmatrix} = \begin{pmatrix} \hat{\mathbf{F}}_{\mathrm{i}}^\ell\left(\hat{\mathbf{F}}_{\mathrm{i}}^\ell\right)^T & \hat{\mathbf{F}}_{\mathrm{i}}^\ell\left(\hat{\mathbf{F}}_{\mathrm{t}}^\ell\right)^T \\ \hat{\mathbf{F}}_{\mathrm{t}}^\ell\left(\hat{\mathbf{F}}_{\mathrm{i}}^\ell\right)^T & \hat{\mathbf{F}}_{\mathrm{t}}^\ell\left(\hat{\mathbf{F}}_{\mathrm{t}}^\ell\right)^T \end{pmatrix} = \begin{pmatrix} \hat{\mathbf{F}}_{\mathrm{i}}^\ell \\ \hat{\mathbf{F}}_{\mathrm{t}}^\ell \end{pmatrix}\begin{pmatrix} \left(\hat{\mathbf{F}}_{\mathrm{i}}^\ell\right)^T & \left(\hat{\mathbf{F}}_{\mathrm{t}}^\ell\right)^T \end{pmatrix} = \hat{\mathbf{F}}_\ell\hat{\mathbf{F}}_\ell^T \tag{42}$$

For $\hat{\mathbf{F}}_{\mathrm{i}}$, we can use any value as long as $\mathbf{G}_{\mathrm{ii}}^\ell = \hat{\mathbf{F}}_{\mathrm{i}}^\ell\left(\hat{\mathbf{F}}_{\mathrm{i}}^\ell\right)^T$, as the resulting distribution over $\mathbf{G}_\ell$ arising from Eq. (38) does not depend on the specific choice of $\hat{\mathbf{F}}_{\mathrm{i}}$ (App. E). Remembering that to sample $\mathbf{G}_{\mathrm{ii}}$, we explicitly sample its potentially low-rank Cholesky, $\mathbf{L}_\ell\mathbf{A}_\ell$, we can directly use

$$\hat{\mathbf{F}}_{\mathrm{i}}^\ell = \mathbf{L}_\ell\mathbf{A}_\ell \tag{43}$$

However, this only works if $\nu \leq P_{\mathrm{i}}$, in which case, $\mathbf{L}_\ell\mathbf{A}_\ell \in \mathbb{R}^{P_{\mathrm{i}}\times\nu}$. In the unusual case where we have fewer inducing points than degrees of freedom, $P_{\mathrm{i}} < \nu$, then $\mathbf{L}_\ell\mathbf{A}_\ell \in \mathbb{R}^{P_{\mathrm{i}}\times P_{\mathrm{i}}}$, so we need to pad to achieve the required size of $P_{\mathrm{i}} \times \nu_\ell$,

$$\hat{\mathbf{F}}_{\mathrm{i}}^\ell = \begin{pmatrix} \mathbf{L}_\ell\mathbf{A}_\ell & \mathbf{0} \end{pmatrix}. \tag{44}$$

Finally, note that we can optimise all the variational parameters using standard reparameterised variational inference (Kingma & Welling, 2013; Rezende et al., 2014). For an algorithm, see Alg. 1.

### 3.5 Computational complexity

Recalling that $\nu_\ell$ is the width of the $\ell$th layer, $P_i$ is the number of inducing points, and $P_t$ is the number of train or test points, the computational complexity of one DWP layer is given by $O(P_i^3 + P_tP_i^2)$. This is a decrease of a factor of $\nu_{\ell+1}$ over the complexity for standard DGP inference, such as doubly stochastic variational inference (Salimbeni & Deisenroth, 2017), which has complexity $O(\nu_{\ell+1}(P_i^3 + P_tP_i^2))$. The difference arises from the fact that in a DGP, $\nu_{\ell+1}$ Gaussian processes are sampled in each layer, whereas for a DWP we sample a single Gram matrix.

## 4 Results

The DWP prior is equivalent to a DGP prior (Sec. 2.4) (Aitchison et al., 2020); the only difference is that in a DGP, we use features as the latent variables, whereas in the DWP we use Gram matrices. Using Gram matrices as in the DWP should be beneficial as the true posteriors are expected to be simpler than in the DGP (Sec. 2.5).

We trained a DWP and a DGP with the exact same generative model with a squared exponential kernel. We trained both models for 20000 gradient steps using the Adam optimizer Kingma & Ba (2014);

**Algorithm 1** Computing predictions/ELBO for one batch

**P parameters:** $\{\nu_\ell\}_{\ell=1}^L$.
**Q parameters:** $\{\mathbf{V}_\ell, p_\ell, \boldsymbol{\alpha}_\ell, \boldsymbol{\beta}_\ell, \boldsymbol{\mu}_\ell, \boldsymbol{\sigma}_\ell\}_{\ell=1}^L, \mathbf{X}_{\mathrm{i}}$.
**Inputs:** $\mathbf{X}_{\mathrm{t}}$; **Targets:** $\mathbf{Y}$
combine inducing and test/train inputs
$\mathbf{X} = (\mathbf{X}_{\mathrm{i}} \quad \mathbf{X}_{\mathrm{t}})$
sample first Gram matrix and update ELBO
$\mathbf{G}_0 = \frac{1}{\nu_0}\mathbf{X}\mathbf{X}^T$
**for** $\ell$ **in** $\{1, \ldots, L\}$ **do**
  sample inducing Gram matrix and its Cholesky, $\mathbf{L}_\ell\mathbf{A}_\ell$ and update ELBO
  $\mathbf{L}_\ell\mathbf{A}_\ell\mathbf{L}_\ell^T\mathbf{A}_\ell^T = \mathbf{G}_{\mathrm{ii}}^\ell \sim \mathrm{Q}\left(\mathbf{G}_{\mathrm{ii}}^\ell | \mathbf{G}_{\mathrm{ii}}^{\ell-1}\right)$
  $\mathcal{L} \leftarrow \mathcal{L} + \log \mathrm{P}\left(\mathbf{G}_{\mathrm{ii}}^\ell | \mathbf{G}_{\mathrm{ii}}^{\ell-1}\right) - \log \mathrm{Q}\left(\mathbf{G}_{\mathrm{ii}}^\ell | \mathbf{G}_{\mathrm{ii}}^{\ell-1}\right)$
  sample full Gram matrix from conditional prior
  $\boldsymbol{\Sigma} = \frac{1}{\nu_\ell}\mathbf{K}_\ell(\mathbf{G}_{\ell-1})$
  $\boldsymbol{\Sigma}_{\mathrm{tt\cdot i}} = \boldsymbol{\Sigma}_{\mathrm{tt}} - \boldsymbol{\Sigma}_{\mathrm{it}}^T\boldsymbol{\Sigma}_{\mathrm{ii}}^{-1}\boldsymbol{\Sigma}_{\mathrm{it}}$
  $\hat{\mathbf{F}}_{\mathrm{i}}^\ell = \mathbf{L}_\ell\mathbf{A}_\ell$
  $\hat{\mathbf{F}}_{\mathrm{t}}^\ell \sim \mathcal{MN}\left(\boldsymbol{\Sigma}_{\mathrm{ti}}^T\boldsymbol{\Sigma}_{\mathrm{ii}}^{-1}\hat{\mathbf{F}}_{\mathrm{i}}, \boldsymbol{\Sigma}_{\mathrm{tt\cdot i}}, \mathbf{I}\right)$
  $\mathbf{G}_\ell = \begin{pmatrix} \mathbf{G}_{\mathrm{ii}}^\ell & \hat{\mathbf{F}}_{\mathrm{i}}^\ell(\hat{\mathbf{F}}_{\mathrm{t}}^\ell)^T \\ \hat{\mathbf{F}}_{\mathrm{t}}^\ell(\hat{\mathbf{F}}_{\mathrm{i}}^\ell)^T & \hat{\mathbf{F}}_{\mathrm{t}}^\ell(\hat{\mathbf{F}}_{\mathrm{t}}^\ell)^T \end{pmatrix}$
**end for**
sample GP inducing outputs and update ELBO
$\mathbf{F}_{\mathrm{i}}^{L+1} \sim \mathrm{Q}\left(\mathbf{F}_{\mathrm{i}}^{L+1} | \mathbf{G}_{\mathrm{ii}}^L\right)$
$\mathcal{L} \leftarrow \mathcal{L} + \log \mathrm{P}\left(\mathbf{F}_{\mathrm{i}}^{L+1} | \mathbf{G}_{\mathrm{ii}}^L\right) - \log \mathrm{Q}\left(\mathbf{F}_{\mathrm{i}}^{L+1} | \mathbf{G}_{\mathrm{ii}}^L\right)$
sample GP predictions conditioned on inducing points
$\mathbf{F}_{\mathrm{t}}^{L+1} \sim \mathrm{Q}\left(\mathbf{F}_{\mathrm{t}}^{L+1} | \mathbf{G}^L, \mathbf{F}_{\mathrm{i}}^{L+1}\right)$
add likelihood to ELBO
$\mathcal{L} \leftarrow \mathcal{L} + \log \mathrm{P}\left(\mathbf{Y} | \mathbf{F}_{\mathrm{t}}^{L+1}\right)$

we detail the exact experimental setup in Appendix F. We report ELBOs and test log likelihoods for depth 5 in Table 1; we report other depths in Appendix G. We found strong benefits for the DWP, which are especially evident if we look at the ELBOs and smaller datasets (boston, concrete, energy, wine and yacht). On larger datasets, the benefits become smaller, as accurate uncertainty modelling is less relevant. Note that we compared against the recently introduced DGP method based on global inducing points (Ober & Aitchison, 2020). Global inducing point methods were particularly important in our setting because we use a standard feedforward architecture without skip connections to ensure equivalence between the DGP and DWP. Standard DSVI has considerable difficulties with optimizing the approximate posterior in such models; to get optimization to work effectively Salimbeni & Deisenroth (2017) were forced to modify the prior to introduce skip connections.

## 5 Related Work

The DWP prior was introduced by Aitchison et al. (2020). However, they were not able to do variational inference with the DWP because they did not have a sufficiently flexible approximate posterior over positive semi-definite matrices. Instead, they were forced to work with a deep *inverse* Wishart process, which is easier because the inverse Wishart itself is a suitable approximate posterior. Here, we give a flexible generalised Wishart distribution over positive semi-definite matrices which is suitable for use as a variational approximate posterior in the DWP. As the deep Wishart process prior is equivalent to a DGP prior, we were able to directly compare DGP and DWP inference in models with the exact same prior. Such a comparison with equivalent priors was not possible in Aitchison et al. (2020), because their deep *inverse* Wishart process priors are not equivalent to DGP priors.

There is an alternative line of work using *generalised* Wishart processes (Wilson & Ghahramani, 2010, as opposed to our *deep* Wishart processes). Note that the "generalised Wishart process" terminology does not seem to have spread as widely as it should, but it is very useful in our

Table 1: ELBOs and log-likelihoods for UCI datasets from (Gal & Ghahramani, 2015) for a five-layer network. See Appendix G for other depths. Significantly better results are highlighted.

| | dataset | DWP | DGP |
|---|---|---|---|
| ELBO | boston | **-0.38 ± 0.01** | -0.46 ± 0.01 |
| | concrete | **-0.49 ± 0.00** | -0.55 ± 0.00 |
| | energy | **1.41 ± 0.00** | 1.37 ± 0.00 |
| | kin8nm | **-0.14 ± 0.00** | -0.18 ± 0.00 |
| | naval | 3.62 ± 0.07 | **3.74 ± 0.08** |
| | power | **0.03 ± 0.00** | 0.01 ± 0.00 |
| | protein | **-1.01 ± 0.00** | -1.02 ± 0.00 |
| | wine | -1.19 ± 0.00 | -1.19 ± 0.00 |
| | yacht | **1.63 ± 0.01** | 1.30 ± 0.02 |
| LL | boston | **-2.39 ± 0.04** | -2.48 ± 0.04 |
| | concrete | **-3.13 ± 0.01** | -3.18 ± 0.01 |
| | energy | -0.70 ± 0.03 | -0.73 ± 0.03 |
| | kin8nm | **1.40 ± 0.01** | 1.38 ± 0.01 |
| | naval | 8.20 ± 0.04 | 8.15 ± 0.07 |
| | power | **-2.77 ± 0.01** | -2.79 ± 0.01 |
| | protein | -2.73 ± 0.00 | -2.74 ± 0.01 |
| | wine | -0.96 ± 0.01 | -0.96 ± 0.01 |
| | yacht | **-0.46 ± 0.12** | -0.77 ± 0.03 |

context. A generalised Wishart process specifies a distribution over an infinite number of finite-dimensional Wishart-distributed matrices. These matrices might represent e.g. the noise covariance in a dynamical system, in which case there might be an infinite number of such matrices, one for each time or location in the state-space (Wilson & Ghahramani, 2010; Heaukulani & van der Wilk, 2019; Jorgensen et al., 2020). In contrast, the Wishart process (Dawid, 1981; Bru, 1991) describes finite dimensional marginals of a single, potentially infinite dimensional matrix. In our context, we stack (non-generalised) Wishart processes to form a deep Wishart process. Importantly, these generalised Wishart priors do not have the flexibility to capture a DGP prior because the underlying features at all locations are jointly multivariate Gaussian (Sec. 4 in Wilson & Ghahramani, 2010) and therefore lack the required nonlinearities between layers. Further, not only do the underlying stochastic processes (deep vs generalised Wishart process) differ, inference is also radically different. In particular, work on the generalised Wishart performs inference on the underlying multivariate Gaussian feature vectors (Eq. 3 e.g. Eq. 15-18 in Wilson & Ghahramani 2010, Eq. 12 in Heaukulani & van der Wilk 2019 or Eq. 24 in Jorgensen et al. 2020). Unfortunately, variational approximate posteriors defined over multivariate Gaussian feature vectors fail to capture symmetries in the true posterior (Sec. 2.5). In contrast, we define approximate posteriors directly over the symmetric positive semi-definite Gram matrices themselves, which required us to develop new, more flexible distributions over these matrices.

# 6 Conclusions

We introduced a flexible distribution over positive semi-definite matrices which formed the basis of a variational approximate posterior for the deep Wishart process. We adapted the doubly stochastic variational inference scheme from Aitchison et al. (2020) to the deep Wishart process. Thus, we were able to directly compare the performance for inference in a DWP vs. DGP with exactly the same prior. This isolates the effects on performance of the prior and the inference procedure. We found considerable benefits, both in terms of predictive performance and the ELBO from doing inference in the DWP rather than the DGP.

There are no anticipated social impacts as the work is largely theoretical.

## References


Laurence Aitchison. Why bigger is not always better: on finite and infinite neural networks. *arXiv preprint arXiv:1910.08013*, 2019.

Laurence Aitchison, Adam X Yang, and Sebastian W Ober. Deep kernel processes. *arXiv preprint arXiv:2010.01590*, 2020.

Maurice Stevenson Bartlett. On the theory of statistical regression. *Proceedings of the Royal Society of Edinburgh*, 53:260–283, 1934.

Christopher M Bishop et al. *Neural networks for pattern recognition*. Oxford university press, 1995.

Marie-France Bru. Wishart processes. *Journal of Theoretical Probability*, 4(4):725–751, 1991.

Andreas Damianou and Neil Lawrence. Deep gaussian processes. In *Artificial Intelligence and Statistics*, pp. 207–215, 2013.

A Philip Dawid. Some matrix-variate distribution theory: notational considerations and a Bayesian application. *Biometrika*, 68(1):265–274, 1981.

Morris L Eaton et al. The wishart distribution. In *Multivariate Statistics*, pp. 302–333. Institute of Mathematical Statistics, 2007.

Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. *arXiv:1506.02142*, 2015.

Adrià Garriga-Alonso, Carl Edward Rasmussen, and Laurence Aitchison. Deep convolutional networks as shallow gaussian processes. *arXiv preprint arXiv:1808.05587*, 2018.

Creighton Heaukulani and Mark van der Wilk. Scalable bayesian dynamic covariance modeling with variational wishart and inverse wishart processes. In *Advances in Neural Information Processing Systems*, pp. 4582–4592, 2019.

Martin Jorgensen, Marc Deisenroth, and Hugh Salimbeni. Stochastic differential equations with variational wishart diffusions. In *International Conference on Machine Learning*, pp. 4974–4983. PMLR, 2020.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Plamen Koev and Alan Edelman. The efficient evaluation of the hypergeometric function of a matrix argument. *Mathematics of Computation*, 75(254):833–846, 2006.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pp. 1097–1105, 2012.

Jaehoon Lee, Yasaman Bahri, Roman Novak, Samuel S Schoenholz, Jeffrey Pennington, and Jascha Sohl-Dickstein. Deep neural networks as gaussian processes. *arXiv preprint arXiv:1711.00165*, 2017.

David JC MacKay. A practical bayesian framework for backpropagation networks. *Neural computation*, 4(3):448–472, 1992.

Arak M Mathai. *Jacobians of matrix transformation and functions of matrix arguments*. World Scientific Publishing Company, 1997.

Arakaparampil M Mathai and Hans J Haubold. *Special functions for applied scientists*. Springer, 2008.

Alexander G de G Matthews, Mark Rowland, Jiri Hron, Richard E Turner, and Zoubin Ghahramani. Gaussian process behaviour in wide deep neural networks. *arXiv preprint arXiv:1804.11271*, 2018.

David A Moore. Symmetrized variational inference. In *NIPS Workshop on Advances in Approximate Bayesian Inferece*, volume 4, pp. 31, 2016.

Roman Novak, Lechao Xiao, Jaehoon Lee, Yasaman Bahri, Greg Yang, Jiri Hron, Daniel A Abolafia, Jeffrey Pennington, and Jascha Sohl-Dickstein. Bayesian deep convolutional networks with many channels are gaussian processes. *arXiv preprint arXiv:1810.05148*, 2018.

Sebastian W Ober and Laurence Aitchison. Global inducing point variational posteriors for bayesian neural networks and deep gaussian processes. *arXiv preprint arXiv:2005.08140*, 2020.

Arya A Pourzanjani, Richard M Jiang, and Linda R Petzold. Improving the identifiability of neural networks for bayesian inference. In *NIPS Workshop on Bayesian Deep Learning*, volume 4, pp. 31, 2017.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint arXiv:1401.4082*, 2014.

Geoffrey Roeder, Yuhuai Wu, and David Duvenaud. Sticking the landing: Simple, lower-variance gradient estimators for variational inference. In *Advances in Neural Information Processing Systems*, 2017.

Hugh Salimbeni and Marc Deisenroth. Doubly stochastic variational inference for deep gaussian processes. In *Advances in Neural Information Processing Systems*, pp. 4588–4599, 2017.

Muni S Srivastava et al. Singular wishart and multivariate beta distributions. *The Annals of Statistics*, 31(5):1537–1560, 2003.

Héctor J Sussmann. Uniqueness of the weights for minimal feedforward nets with a given input-output map. *Neural networks*, 5(4):589–593, 1992.

Christopher KI Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006.

Andrew Gordon Wilson and Zoubin Ghahramani. Generalised Wishart processes. *arXiv preprint arXiv:1101.0240*, 2010.

## Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] Though none were anticipated
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] (Code in supplementary material)
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [N/A] UCI datasets have unclear licenses.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] (Code in supplementary material
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A] Standard datasets.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]