

## A Deriving Jacobians for matrix transformations

Following the approach in (Mathai, 1997; Mathai & Haubold, 2008), we define the Jacobian of a function from  $x$  to  $y$  as the ratio of volume elements,

$$\text{jacobian} = \frac{dy_1 dy_2 \cdots dy_N}{dx_1 dx_2 \cdots dx_N}.$$

Critically,  $dx_i$  and  $dy_i$  are basis-vectors, *not* scalars. As we are multiplying vectors, not scalars, we need to be careful about our choice of multiplication operation. The correct choice in our context is an anti-symmetric exterior product, representing a directed area or volume element, such that,

$$dx_i dx_j = -dx_j dx_i.$$

As the product is antisymmetric, the product of a basis-vector with itself is zero,

$$dx_i dx_i = -dx_i dx_i = 0,$$

which makes sense because the product represents an area, and the area is zero if the two vectors are aligned. To confirm that this matches usual expressions for Jacobians, consider a  $2 \times 2$  matrix-vector multiplication,  $\mathbf{y} = \mathbf{A}\mathbf{x}$ :

$$\begin{pmatrix} dy_1 \\ dy_2 \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} dx_1 \\ dx_2 \end{pmatrix} = \begin{pmatrix} A_{11}dx_1 + A_{12}dx_2 \\ A_{21}dx_1 + A_{22}dx_2 \end{pmatrix}.$$

Thus,

$$\begin{aligned} dy_1 dy_2 &= (A_{11}dx_1 + A_{12}dx_2)(A_{21}dx_1 + A_{22}dx_2) \\ dy_1 dy_2 &= A_{11}A_{21}dx_1^2 + A_{11}A_{22}dx_1 dx_2 + A_{12}A_{21}dx_2 dx_1 + A_{12}A_{22}dx_2^2 \end{aligned}$$

As  $dx_1^2 = dx_2^2 = 0$ , and  $dx_1 dx_2 = -dx_2 dx_1$ , we have,

$$\begin{aligned} dy_1 dy_2 &= (A_{11}A_{22} - A_{12}A_{21}) dx_1 dx_2 \\ dy_1 dy_2 &= |\mathbf{A}| dx_1 dx_2, \end{aligned}$$

so that the Jacobian computed using the determinant definition is equivalent to the expression for the determinant obtained by working with volume elements.

## B Jacobian for the product of a lower triangular matrix with itself

In this section, we compute the Jacobian for the transformation from  $\mathbf{\Lambda} = \mathbf{L}\mathbf{A}$  to  $\mathbf{G} = \mathbf{\Lambda}\mathbf{\Lambda}^T$ . We begin by noting that the top-left block of the product of a lower-triangular matrix with itself is a product of smaller lower-triangular matrices:

$$\begin{pmatrix} \mathbf{\Lambda}_{:,N,:N} & \mathbf{0} \\ \mathbf{\Lambda}_{N,:N} & \mathbf{\Lambda}_{:,N,:N} \end{pmatrix} \begin{pmatrix} \mathbf{\Lambda}_{:,N,:N}^T & \mathbf{\Lambda}_{N,:N}^T \\ \mathbf{0} & \mathbf{\Lambda}_{:,N,:N}^T \end{pmatrix} = \begin{pmatrix} \mathbf{\Lambda}_{:,N,:N}\mathbf{\Lambda}_{:,N,:N}^T & \cdots \\ \vdots & \ddots \end{pmatrix}.$$

As such, we can incrementally compute the Jacobian for this transformation by starting with the top-left  $1 \times 1$  matrix,

$$G_{11} = \Lambda_{11}^2 \tag{18}$$

$$dG_{11} = 2\Lambda_{11}d\Lambda_{11}. \tag{19}$$

Next, we consider the top-left  $2 \times 2$  matrix,

$$\begin{pmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{pmatrix} = \begin{pmatrix} \Lambda_{11} & 0 \\ \Lambda_{21} & \Lambda_{22} \end{pmatrix} \begin{pmatrix} \Lambda_{11} & \Lambda_{21} \\ 0 & \Lambda_{22} \end{pmatrix} = \begin{pmatrix} \Lambda_{11}^2 & \Lambda_{21}\Lambda_{11} \\ \Lambda_{21}\Lambda_{11} & \Lambda_{21}^2 + \Lambda_{22}^2 \end{pmatrix}.$$

Thus

$$\begin{aligned} dG_{21} &= \Lambda_{21}d\Lambda_{11} + \Lambda_{11}d\Lambda_{21} \\ dG_{22} &= 2\Lambda_{22}d\Lambda_{22} + 2\Lambda_{21}d\Lambda_{21}. \end{aligned}$$

Combining  $dG_{11}$  and  $dG_{21}$  gives

$$\begin{aligned} dG_{11}dG_{21} &= (2\Lambda_{11}d\Lambda_{11})(\Lambda_{21}d\Lambda_{11} + \Lambda_{11}d\Lambda_{21}) \\ dG_{11}dG_{21} &= 2\Lambda_{11}^2(d\Lambda_{11}d\Lambda_{21}), \end{aligned}$$

and then combining  $dG_{11}dG_{21}$  and  $dG_{22}$  gives

$$\begin{aligned} dG_{11}dG_{21}dG_{22} &= (2\Lambda_{11}^2(d\Lambda_{11}d\Lambda_{21}))(2\Lambda_{22}d\Lambda_{22} + 2\Lambda_{21}d\Lambda_{21}) \\ dG_{11}dG_{21}dG_{22} &= 4\Lambda_{11}^2\Lambda_{22}(d\Lambda_{11}d\Lambda_{21}d\Lambda_{22}). \end{aligned}$$

Thus, we can prove by induction that the volume element for the top-left  $p \times p$  block of  $\mathbf{G}$ , and in addition the first  $K < p + 1$  off-diagonal elements of the  $p + 1$ th row is

$$\underbrace{\left( \prod_{i=1}^p \prod_{k=1}^i dG_{ik} \right)}_{\text{vol. elem. for } \mathbf{G}_{:p,:p}} \underbrace{\left( \prod_{k=1}^K dG_{p+1,k} \right)}_{\text{vol. elem. for } \mathbf{G}_{p+1,:K}} = 2^p \left( \prod_{i=1}^p \prod_{k=1}^i \Lambda_{kk} d\Lambda_{ik} \right) \left( \prod_{k=1}^K d\Lambda_{kk} \Lambda_{p+1,k} \right).$$

The proof consists of three parts: the base case, adding an off-diagonal element and adding an on-diagonal element. For the base-case, note that the expression is correct for  $p = 1$  and  $K = 0$  (Eq. 19). Next, we add an off-diagonal element,  $G_{p+1,K+1}$ , where  $K + 1 < p + 1$ . We begin by computing  $dG_{p+1,K+1}$ . Note that the sum only goes to  $K + 1$ , because  $\Lambda_{K+1,j} = 0$  for  $j > (K + 1)$ :

$$\begin{aligned} G_{p+1,K+1} &= \sum_{j=1}^{K+1} \Lambda_{p+1,j} \Lambda_{K+1,j}, \\ dG_{p+1,K+1} &= \sum_{j=1}^{K+1} (\Lambda_{K+1,j} d\Lambda_{p+1,j} + \Lambda_{p+1,j} d\Lambda_{K+1,j}). \end{aligned}$$

Remembering that  $d\Lambda_{ij}^2 = 0$ , the only term that does not cancel when we multiply by the volume element for the previous terms is that for  $d\Lambda_{p+1,K+1}$ :

$$\begin{aligned} \underbrace{\left( \prod_{i=1}^p \prod_{k=1}^i dG_{ik} \right)}_{\text{vol. elem. for } \mathbf{G}_{:p,:p}} \underbrace{\left( \prod_{k=1}^{K+1} dG_{p+1,k} \right)}_{\text{vol. elem. for } \mathbf{G}_{p+1,:K+1}} &= \left( \prod_{i=1}^p \prod_{k=1}^i dG_{ik} \right) \left( \prod_{k=1}^K dG_{p+1,k} \right) dG_{p+1,K+1} \\ &= 2^p \left( \prod_{i=1}^p \prod_{k=1}^i \Lambda_{kk} d\Lambda_{ik} \right) \left( \prod_{k=1}^K d\Lambda_{kk} \Lambda_{p+1,k} \right) dG_{p+1,K+1} \\ &= 2^p \left( \prod_{i=1}^p \prod_{k=1}^i \Lambda_{kk} d\Lambda_{ik} \right) \left( \prod_{k=1}^K d\Lambda_{kk} \Lambda_{p+1,k} \right) (\Lambda_{K+1,K+1} d\Lambda_{p+1,K+1}) \\ &= 2^p \left( \prod_{i=1}^p \prod_{k=1}^i \Lambda_{kk} d\Lambda_{ik} \right) \left( \prod_{k=1}^{K+1} d\Lambda_{kk} \Lambda_{p+1,k} \right). \end{aligned}$$

So the expression is consistent when adding an on-diagonal element. Finally, the volume element for  $G_{p+1,p+1}$  is,

$$G_{p+1,p+1} = \sum_{j=1}^{p+1} \Lambda_{p+1,j}^2 dG_{p+1,K+1} = \sum_{j=1}^{K+1} \Lambda_{p+1,j} d\Lambda_{p+1,j}.$$

Remembering that  $d\Lambda_{ij}^2 = 0$ , the only term that does not cancel when we multiply by the volume element for the previous terms is that for  $d\Lambda_{p+1,p+1}$ ,

$$\begin{aligned}
\underbrace{\left( \prod_{i=1}^{p+1} \prod_{k=1}^i dG_{ik} \right)}_{\text{vol. elem. for } \mathbf{G}_{:p+1,:p+1}} &= \underbrace{\left( \prod_{i=1}^p \prod_{k=1}^i dG_{ik} \right)}_{\text{vol. elem. for } \mathbf{G}_{:p,:p}} \underbrace{\left( \prod_{k=1}^{p+1} dG_{p+1,k} \right)}_{\text{vol. elem. for } \mathbf{G}_{p+1,:p+1}} \\
&= \left( \prod_{i=1}^p \prod_{k=1}^i dG_{ik} \right) \left( \prod_{k=1}^p dG_{p+1,k} \right) dG_{p+1,p+1} \\
&= 2^p \left( \prod_{i=1}^p \prod_{k=1}^i \Lambda_{kk} d\Lambda_{ik} \right) \left( \prod_{k=1}^p d\Lambda_{kk} \Lambda_{p+1,k} \right) dG_{p+1,p+1} \\
&= 2^p \left( \prod_{i=1}^p \prod_{k=1}^i \Lambda_{kk} d\Lambda_{ik} \right) \left( \prod_{k=1}^p d\Lambda_{kk} \Lambda_{p+1,k} \right) (2\Lambda_{p+1,p+1} d\Lambda_{p+1,p+1}) \\
&= 2^{p+1} \left( \prod_{i=1}^p \prod_{k=1}^i \Lambda_{kk} d\Lambda_{ik} \right) \left( \prod_{k=1}^{p+1} d\Lambda_{kk} \Lambda_{p+1,k} \right) \\
&= 2^{p+1} \left( \prod_{i=1}^{p+1} \prod_{k=1}^i \Lambda_{kk} d\Lambda_{ik} \right).
\end{aligned}$$

Thus, the final result is:

$$\left( \prod_{i=1}^P \prod_{k=1}^i dG_{ik} \right) = \left( 2^P \prod_{i=1}^{\min(P,\nu)} \Lambda_{ii}^{P+1-i} \right) \left( \prod_{i=1}^P \prod_{k=1}^i d\Lambda_{ik} \right). \quad (20)$$

$$d\mathbf{G} = d\Lambda \prod_{i=1}^P 2\Lambda_{ii}^{P+1-i}. \quad (21)$$

## B.1 Singular matrices

The above derivation can be extended to the singular case, where  $\Lambda$  has a form mirroring that of  $\mathbf{A}$  in Eq. (7a):

$$\Lambda = \begin{pmatrix} \Lambda_{11} & \dots & 0 \\ \vdots & \ddots & \vdots \\ \Lambda_{\nu 1} & \dots & \Lambda_{\nu\nu} \\ \vdots & \vdots & \vdots \\ \Lambda_{P1} & \dots & \Lambda_{P\nu} \end{pmatrix}. \quad (22)$$

To form a valid Jacobian, we need the same number of inputs as outputs. We therefore consider differences in only the corresponding part of  $\mathbf{G}$  (i.e.  $G_{i,j \leq \min(i,\nu)}$ ). The recursive expression is

$$\underbrace{\left( \prod_{i=1}^p \prod_{k=1}^{\min(i,\nu)} dG_{ik} \right)}_{\text{vol. elem. for } \mathbf{G}_{:p,:p}} \underbrace{\left( \prod_{k=1}^K dG_{p+1,k} \right)}_{\text{vol. elem. for } \mathbf{G}_{p+1,:K}} = 2^{\min(p,\nu)} \left( \prod_{i=1}^p \prod_{k=1}^{\min(i,\nu)} \Lambda_{kk} d\Lambda_{ik} \right) \left( \prod_{k=1}^K d\Lambda_{kk} \Lambda_{p+1,k} \right),$$

where  $K < \min(p, \nu)$ . For  $P \leq \nu$ , the recursion is exactly as in the full-rank case. For  $P > \nu$ , the key difference is that there are no longer any on-diagonal elements. As such, for  $K = \nu$  we have

$$\begin{aligned}
\left( \prod_{i=1}^{p+1} \prod_{k=1}^{\min(i, \nu)} dG_{ik} \right) &= \underbrace{\left( \prod_{i=1}^p \prod_{k=1}^{\min(i, \nu)} dG_{ik} \right)}_{\text{vol. elem. for } \mathbf{G}_{:,p,:p}} \underbrace{\left( \prod_{k=1}^{\nu} dG_{p+1,k} \right)}_{\text{vol. elem. for } \mathbf{G}_{p+1,:,\nu}} \\
&= 2^{\min(p, \nu)} \left( \prod_{i=1}^p \prod_{k=1}^{\min(i, \nu)} \Lambda_{kk} d\Lambda_{ik} \right) \left( \prod_{k=1}^K d\Lambda_{kk} \Lambda_{p+1,k} \right) \\
&= 2^{\min(p, \nu)} \left( \prod_{i=1}^{p+1} \prod_{k=1}^{\min(i, \nu)} \Lambda_{kk} d\Lambda_{ik} \right).
\end{aligned}$$

The final expression, allowing for the possibility of singular and non-singular matrices, is thus

$$\begin{aligned}
\left( \prod_{i=1}^P \prod_{k=1}^{\min(i, \nu)} dG_{ik} \right) &= \left( \prod_{i=1}^{\min(P, \nu)} 2\Lambda_{ii}^{P+1-i} \right) \left( \prod_{i=1}^P \prod_{k=1}^{\min(i, \nu)} d\Lambda_{ik} \right) \\
d\mathbf{G} &= d\mathbf{\Lambda} \prod_{i=1}^{\min(P, \nu)} 2\Lambda_{ii}^{P+1-i}.
\end{aligned}$$

## C Jacobian for product of two different lower triangular matrices

In this section, we compute the Jacobian for the transformation from  $\mathbf{A}$  to  $\mathbf{\Lambda} = \mathbf{L}\mathbf{A}$ . We begin by noting that  $\mathbf{\Lambda}$  (Eq. 22) is a potentially rectangular lower-triangular matrix, with the same structure as  $\mathbf{A}$ . Writing this out,

$$\begin{pmatrix} \Lambda_{11} & 0 & 0 \\ \Lambda_{21} & \Lambda_{22} & 0 \\ \Lambda_{31} & \Lambda_{32} & \Lambda_{33} \\ \Lambda_{41} & \Lambda_{42} & \Lambda_{43} \\ \Lambda_{51} & \Lambda_{52} & \Lambda_{53} \end{pmatrix} = \begin{pmatrix} L_{11} & 0 & 0 & 0 & 0 \\ L_{21} & L_{22} & 0 & 0 & 0 \\ L_{31} & L_{32} & L_{33} & 0 & 0 \\ L_{41} & L_{42} & L_{43} & L_{44} & 0 \\ L_{51} & L_{52} & L_{53} & L_{54} & L_{55} \end{pmatrix} \begin{pmatrix} A_{11} & 0 & 0 \\ A_{21} & A_{22} & 0 \\ A_{31} & A_{32} & A_{33} \\ A_{41} & A_{42} & A_{43} \\ A_{51} & A_{52} & A_{53} \end{pmatrix}.$$

For the first column,

$$\begin{pmatrix} \Lambda_{11} \\ \Lambda_{21} \\ \Lambda_{31} \\ \Lambda_{41} \\ \Lambda_{51} \end{pmatrix} = \begin{pmatrix} L_{11} & 0 & 0 & 0 & 0 \\ L_{21} & L_{22} & 0 & 0 & 0 \\ L_{31} & L_{32} & L_{33} & 0 & 0 \\ L_{41} & L_{42} & L_{43} & L_{44} & 0 \\ L_{51} & L_{52} & L_{53} & L_{54} & L_{55} \end{pmatrix} \begin{pmatrix} A_{11} \\ A_{21} \\ A_{31} \\ A_{41} \\ A_{51} \end{pmatrix},$$

i.e.

$$\mathbf{\Lambda}_{:,1} = \mathbf{L}\mathbf{A}_{:,1}.$$

For the second column,

$$\begin{pmatrix} 0 \\ \Lambda_{22} \\ \Lambda_{32} \\ \Lambda_{42} \\ \Lambda_{52} \end{pmatrix} = \begin{pmatrix} L_{11} & 0 & 0 & 0 & 0 \\ L_{21} & L_{22} & 0 & 0 & 0 \\ L_{31} & L_{32} & L_{33} & 0 & 0 \\ L_{41} & L_{42} & L_{43} & L_{44} & 0 \\ L_{51} & L_{52} & L_{53} & L_{54} & L_{55} \end{pmatrix} \begin{pmatrix} 0 \\ A_{22} \\ A_{32} \\ A_{42} \\ A_{52} \end{pmatrix}.$$

We can eliminate the first row and column of  $\mathbf{C}$ ,

$$\begin{pmatrix} \Lambda_{22} \\ \Lambda_{32} \\ \Lambda_{42} \\ \Lambda_{52} \end{pmatrix} = \begin{pmatrix} L_{22} & 0 & 0 & 0 \\ L_{32} & L_{33} & 0 & 0 \\ L_{42} & L_{43} & L_{44} & 0 \\ L_{52} & L_{53} & L_{54} & L_{55} \end{pmatrix} \begin{pmatrix} A_{22} \\ A_{32} \\ A_{42} \\ A_{52} \end{pmatrix},$$

i.e.

$$\mathbf{\Lambda}_{2:,2} = \mathbf{L}_{2:,2} \mathbf{A}_{2:,2}.$$

For the third column,

$$\begin{pmatrix} 0 \\ 0 \\ \Lambda_{33} \\ \Lambda_{43} \\ \Lambda_{53} \end{pmatrix} = \begin{pmatrix} L_{11} & 0 & 0 & 0 & 0 \\ L_{21} & L_{22} & 0 & 0 & 0 \\ L_{31} & L_{32} & L_{33} & 0 & 0 \\ L_{41} & L_{42} & L_{43} & L_{44} & 0 \\ L_{51} & L_{52} & L_{53} & L_{54} & L_{55} \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ A_{33} \\ A_{43} \\ A_{53} \end{pmatrix},$$

so we can eliminate the first two rows and columns of  $\mathbf{C}$ :

$$\begin{pmatrix} \Lambda_{33} \\ \Lambda_{43} \\ \Lambda_{53} \end{pmatrix} = \begin{pmatrix} L_{33} & 0 & 0 \\ L_{43} & L_{44} & 0 \\ L_{53} & L_{54} & L_{55} \end{pmatrix} \begin{pmatrix} A_{33} \\ A_{43} \\ A_{53} \end{pmatrix},$$

i.e.

$$\mathbf{\Lambda}_{3:,3} = \mathbf{L}_{3:,3} \mathbf{A}_{3:,3}.$$

As such, the full computation  $\mathbf{\Lambda} = \mathbf{L}\mathbf{A}$  can be written as a matrix-vector multiplication by rearranging the columns of  $\mathbf{\Lambda}$  and  $\mathbf{A}$  into a single vector:

$$\begin{pmatrix} \mathbf{\Lambda}_{1:,1} \\ \mathbf{\Lambda}_{2:,2} \\ \vdots \\ \mathbf{\Lambda}_{\nu:,\nu} \end{pmatrix} = \begin{pmatrix} \mathbf{L}_{1:,1} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{L}_{2:,2} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{L}_{\nu:,\nu} \end{pmatrix} \begin{pmatrix} \mathbf{A}_{1:,1} \\ \mathbf{A}_{2:,2} \\ \vdots \\ \mathbf{A}_{\nu:,\nu} \end{pmatrix}.$$

The Jacobian is given by the determinant of the square matrix, and as the matrix is lower-triangular, the determinant can be written in terms of the diagonal elements of  $\mathbf{L}$ ,

$$\left( \prod_{i=1}^P \prod_{k=1}^{\min(i,\nu)} d\Lambda_{ik} \right) = \left( \prod_{i=1}^P L_{ii}^{\min(i,\nu)} \right) \left( \prod_{i=1}^P \prod_{k=1}^{\min(i,\nu)} dA_{ik} \right).$$

## D Proving the singular Bartlett (above) corresponds to the Wishart

We need to change variables to  $A_{jj}$  rather than  $A_{jj}^2$ <sup>1</sup>:

$$\begin{aligned} P(A_{jj}) &= P(A_{jj}^2) \left| \frac{\partial A_{jj}^2}{\partial A_{jj}} \right|, \\ &= \text{Gamma}(A_{jj}^2; \frac{\nu-j+1}{2}, \frac{1}{2}) 2A_{jj}, \\ &= \frac{(A_{jj}^2)^{(\nu-j+1)/2-1} e^{-A_{jj}^2/2}}{2^{(\nu-j+1)/2} \Gamma(\frac{\nu-j+1}{2})} 2A_{jj}, \\ &= \frac{A_{jj}^{\nu-j} e^{-A_{jj}^2/2}}{2^{(\nu-j-1)/2} \Gamma(\frac{\nu-j+1}{2})}. \end{aligned}$$

Thus, the probability density for  $\mathbf{A}$  under the Bartlett sampling operation is

$$P(\mathbf{A}) = \underbrace{\prod_{j=1}^{\tilde{\nu}} \frac{A_{jj}^{\nu-j} e^{-T_{jj}^2/2}}{2^{\frac{\nu-j-1}{2}} \Gamma(\frac{\nu-j+1}{2})}}_{\text{on-diagonals}} \underbrace{\prod_{i=j+1}^P \frac{1}{\sqrt{2\pi}} e^{-A_{ij}^2/2}}_{\text{off-diagonals}}, \quad (23)$$

<sup>1</sup>[djalil.chafai.net/blog/2015/10/20/bartlett-decomposition-and-other-factorizations/](http://djalil.chafai.net/blog/2015/10/20/bartlett-decomposition-and-other-factorizations/)

where  $\tilde{\nu} = \min(\nu, P)$ . To convert this to a distribution on  $\mathbf{G}$ , we need the volume element for the transformation from  $\mathbf{A}$  to  $\mathbf{Z} = \mathbf{A}\mathbf{A}^T$ , which is given in Appendix B (Eq. (21)):

$$d\mathbf{Z} = d\mathbf{A} \prod_{j=1}^{\tilde{\nu}} 2A_{jj}^{P-j+1}.$$

Thus

$$\begin{aligned} P(\mathbf{Z}) &= P(\mathbf{A}) \left( \prod_{j=1}^{\tilde{\nu}} \frac{1}{2} A_{jj}^{-(P-j+1)} \right) \\ &= \prod_{j=1}^{\tilde{\nu}} \frac{A_{jj}^{\nu-P-1} e^{-T_{jj}^2/2}}{2^{\frac{\nu-j+1}{2}} \Gamma(\frac{\nu-j+1}{2})} \prod_{i=j+1}^P \frac{1}{\sqrt{2\pi}} e^{-A_{ij}^2/2}. \end{aligned}$$

Now, we break this expression down into separate components and perform standard algebraic manipulations. First, we manipulate a product over the diagonal elements of  $\mathbf{A}$  to obtain the determinant of  $\mathbf{Z}$ :

$$\prod_{j=1}^{\tilde{\nu}} A_{jj}^{\nu-P-1} = \left( \prod_{j=1}^{\tilde{\nu}} A_{jj} \right)^{\nu-P-1} = |\mathbf{A}_{:\tilde{\nu},:} \mathbf{A}_{:\tilde{\nu},:}^T|^{\nu-P-1} = |\mathbf{Z}_{:\tilde{\nu},:\tilde{\nu}}|^{\nu-P-1}.$$

Next, we manipulate the exponential terms to form an exponentiated trace. We start by combining on- and off-diagonal terms, and noting that  $A_{ij} = 0$  for  $i < j$ , we can extend the sum

$$\prod_{j=1}^{\tilde{\nu}} e^{-A_{jj}^2/2} \prod_{i=j+1}^P e^{-A_{ij}^2/2} = \prod_{j=1}^{\tilde{\nu}} \prod_{i=j}^P e^{-A_{ij}^2/2} = \prod_{j=1}^{\tilde{\nu}} \prod_{i=1}^P e^{-A_{ij}^2/2}.$$

Then we take the product inside the exponential and note that as  $\mathbf{Z} = \mathbf{A}\mathbf{A}^T$ , we can write the sum as a trace of  $\mathbf{Z}$ ,

$$= e^{\sum_{j=1}^{\tilde{\nu}} \sum_{i=1}^P -A_{ij}^2/2} = e^{-\text{Tr}(\mathbf{Z})/2}.$$

Next, we consider powers of 2. We begin by computing the number of  $1/\sqrt{2}$  terms, arising from the off-diagonal elements,

$$\prod_{j=1}^{\tilde{\nu}} \prod_{i=j+1}^P \frac{1}{\sqrt{2}} = \left( \frac{1}{\sqrt{2}} \right)^{\nu(p-\tilde{\nu}) + \tilde{\nu}(\tilde{\nu}-1)/2}.$$

Note that the  $\tilde{\nu}(\tilde{\nu}-1)$  term corresponds to the off-diagonal terms in the square block  $\mathbf{A}_{:\tilde{\nu},:}$ , and the  $\nu(p-\tilde{\nu})$  term corresponds to the terms from  $\mathbf{A}_{\tilde{\nu},:}$ . Next we consider the on-diagonal terms,

$$\prod_{j=1}^{\tilde{\nu}} \frac{1}{2^{(\nu-j+1)/2}} = \left( \frac{1}{\sqrt{2}} \right)^{\tilde{\nu}(\nu+1)} \prod_{j=1}^{\tilde{\nu}} \left( \frac{1}{\sqrt{2}} \right)^{-j} = \left( \frac{1}{\sqrt{2}} \right)^{\tilde{\nu}(\nu+1) - \tilde{\nu}(\tilde{\nu}+1)/2}.$$

Combining the on and off-diagonal terms,

$$\begin{aligned} \prod_{j=1}^{\tilde{\nu}} \frac{1}{2^{(\nu-j+1)/2}} \prod_{i=j+1}^P \frac{1}{\sqrt{2}} &= \left( \frac{1}{\sqrt{2}} \right)^{\nu(P-\tilde{\nu}) + \tilde{\nu}(\tilde{\nu}-1)/2 + \tilde{\nu}(\nu+1) - \tilde{\nu}(\tilde{\nu}+1)/2} \\ &= \left( \frac{1}{\sqrt{2}} \right)^{(\nu P - \nu \tilde{\nu}) + (\tilde{\nu}^2/2 - \tilde{\nu}/2) + (\tilde{\nu}\nu + \tilde{\nu}) + (-\tilde{\nu}^2/2 - \tilde{\nu}/2)} \\ &= \left( \frac{1}{\sqrt{2}} \right)^{\nu P}. \end{aligned}$$

Finally, using the definition of the multivariate Gamma function,

$$\begin{aligned} \prod_{j=1}^{\tilde{\nu}} \Gamma\left(\frac{\nu-j+1}{2}\right) \prod_{i=j+1}^P \sqrt{\pi} &= \pi^{\nu(P-\tilde{\nu})/2} \underbrace{\pi^{\tilde{\nu}(\tilde{\nu}-1)/4} \prod_{j=1}^{\tilde{\nu}} \Gamma\left(\frac{\nu-j+1}{2}\right)}_{=\Gamma_{\tilde{\nu}}\left(\frac{\nu}{2}\right)} \\ &= \pi^{\nu(P-\tilde{\nu})/2} \Gamma_{\tilde{\nu}}\left(\frac{\nu}{2}\right). \end{aligned}$$

We thereby re-obtain the probability density for the standard Wishart distribution,

$$P(\mathbf{Z}) = \frac{\pi^{\nu(\tilde{\nu}-P)/2}}{2^{\nu P/2} \Gamma_{\tilde{\nu}}\left(\frac{\nu}{2}\right)} |\mathbf{Z}_{:, \tilde{\nu}, : \tilde{\nu}}|^{(\nu-P-1)/2} e^{-\text{Tr}(\mathbf{Z})/2}.$$

For  $\tilde{\nu} = \nu$ , this matches Eq. 3.2 in Srivastava et al. (2003), and for  $\tilde{\nu} = P$  it matches the standard full-rank Wishart probability density function.

## E Choice of $\mathbf{F}_i$

Here, we establish that the distribution over  $\mathbf{F}_t \mathbf{F}_t^T$  and  $\mathbf{F}_t \mathbf{F}_t^T$  does not depend on the choice of  $\mathbf{F}_i$ . Due to the definition of  $\mathbf{F}_t$  (Eq. 17) we can write,

$$\mathbf{F}_t = \mathbf{K}_u^T \mathbf{K}_{ii}^{-1} \mathbf{F}_i + \mathbf{K}_{u-i}^{1/2} \Xi.$$

where  $\Xi$  is a matrix with IID standard Gaussian elements. Thus,

$$\begin{aligned} \mathbf{F}_t \mathbf{F}_i^T &= \mathbf{K}_u^T \mathbf{K}_{ii}^{-1} \mathbf{F}_i \mathbf{F}_i^T + \mathbf{K}_{u-i}^{1/2} \Xi \mathbf{F}_i^T \\ \mathbf{F}_t \mathbf{F}_i^T &\sim \mathcal{MN}(\mathbf{K}_u^T \mathbf{K}_{ii}^{-1} \mathbf{G}_{ii}, \mathbf{K}_{u-i}, \mathbf{G}_{ii}). \end{aligned}$$

We can do the same for  $\mathbf{F}_t \mathbf{F}_t^T$ :

$$\begin{aligned} \mathbf{F}_t \mathbf{F}_t^T &= \mathbf{K}_u^T \mathbf{K}_{ii}^{-1} \mathbf{F}_i \mathbf{F}_i^T \mathbf{K}_{ii}^{-1} \mathbf{K}_u \\ &\quad + \mathbf{K}_u^T \mathbf{K}_{ii}^{-1} \mathbf{F}_i \Xi \mathbf{K}_{u-i}^{1/2} + \mathbf{K}_{u-i}^{1/2} \Xi \mathbf{F}_i^T \mathbf{K}_{ii}^{-1} \mathbf{K}_u \\ &\quad + \mathbf{K}_{u-i}^{1/2} \Xi \Xi \mathbf{K}_{u-i}^{1/2}. \end{aligned}$$

The first term is independent of the choice of  $\mathbf{F}_i$  because  $\mathbf{G}_{ii} = \mathbf{F}_i \mathbf{F}_i^T$ . The term on the last line does not depend on  $\mathbf{F}_i$  at all. Finally, the two terms in the middle are Gaussian with covariance that depends on  $\mathbf{G}_{ii}$  but not the specific choice of  $\mathbf{F}_i$ :

$$\begin{aligned} \mathbf{K}_u^T \mathbf{K}_{ii}^{-1} \mathbf{F}_i \Xi \mathbf{K}_{u-i}^{1/2} &\sim \mathcal{MN}(\mathbf{0}, \mathbf{K}_u^T \mathbf{K}_{ii}^{-1} \mathbf{G}_i \mathbf{K}_{ii}^{-1} \mathbf{K}_u, \mathbf{K}_{u-i}), \\ \mathbf{K}_{u-i}^{1/2} \Xi \mathbf{F}_i^T \mathbf{K}_{ii}^{-1} \mathbf{K}_u &\sim \mathcal{MN}(\mathbf{0}, \mathbf{K}_{u-i}, \mathbf{K}_u^T \mathbf{K}_{ii}^{-1} \mathbf{G}_i \mathbf{K}_{ii}^{-1} \mathbf{K}_u). \end{aligned}$$

Thus, the additional components of  $\mathbf{G}$ ,  $\mathbf{G}_{ii} = \mathbf{F}_i \mathbf{F}_i$  and  $\mathbf{G}_{ii} = \mathbf{F}_i \mathbf{F}_i^T$  depend on  $\mathbf{G}_{ii}$  but not on the specific choice of  $\mathbf{F}_i$ . Thus, any  $\mathbf{F}_i$  can be used as long as  $\mathbf{G}_{ii} = \mathbf{F}_i \mathbf{F}_i^T$ .

## F Experimental details

**Datasets** All experiments were performed using the UCI splits from Gal & Ghahramani (2015), available at [https://github.com/yaringal/DropoutUncertaintyExps/tree/master/UCI\\_Datasets](https://github.com/yaringal/DropoutUncertaintyExps/tree/master/UCI_Datasets). For each dataset there are twenty splits, with the exception of protein, which only has five. We report mean plus or minus one standard error over the splits.

**Model details** As standard, we set  $\nu$  (the ‘width’ of each layer) to be equal to the dimensionality of the input space. We use the squared exponential kernel, with automatic relevance determination (ARD) in the first layer, but without for the intermediate layers as ARD relies on explicit features existing. However, we found in practice that using ARD for intermediate layers in a DGP did not hugely affect the results, as each output GP in a layer shares the same prior and hence output prior variance. For the final, GP, layer of the DWP model we use a global inducing approximate posterior (Ober & Aitchison, 2021), as done in the DGP. We leave the particular implementation details for the code provided with the paper, but we note that we use the ‘sticking the landing’ gradient estimator (Roeder et al., 2017) for the  $\{\alpha_\ell, \beta_\ell, \mu_\ell, \sigma_\ell\}_{\ell=1}^L$  approximate posterior parameters of the DWP (using it for the other parameters, as well as for the DGP parameters, is difficult as the parameters of one layer will affect the KL estimate of the following layers for global inducing posteriors).

**Training details** We train all models using the same training scheme. We use 20,000 gradient steps to train each model, using the Adam optimizer (Kingma & Ba, 2015) with an initial learning rate of 1e-2. We anneal the KL using a factor increasing linearly from 0 to 1 over the first 1,000 gradient steps, and step the learning rate down to 1e-3 after 10,000 gradient steps. We use 10 samples from the approximate posterior for training, and 100 for testing. Experiments were performed using an internal cluster of machines with NVIDIA GeForce 2080 Ti GPUs, although we used CPU (Intel Core i9-10900X) for the smaller datasets (boston, concrete, energy, wine, yacht).

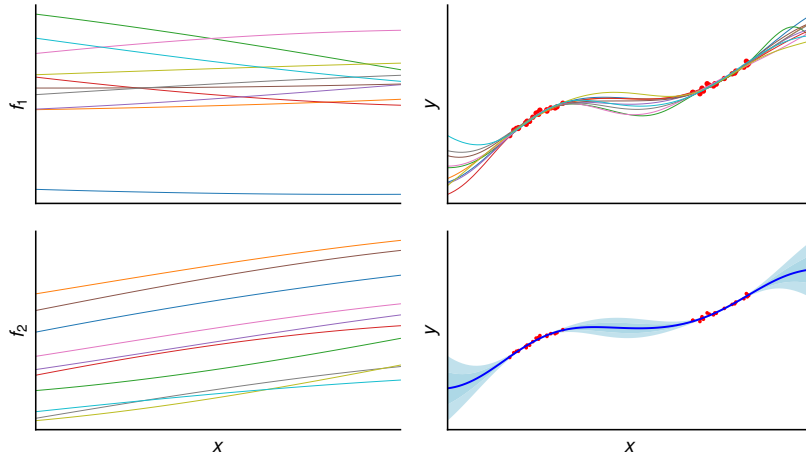


Figure 2: Features from a 2-layer DGP posterior with intermediate width 2: feature samples  $f_1$  (first layer, first output; top left),  $f_2$  (first layer, second output; bottom left), posterior samples (top right), posterior predictive (bottom right)

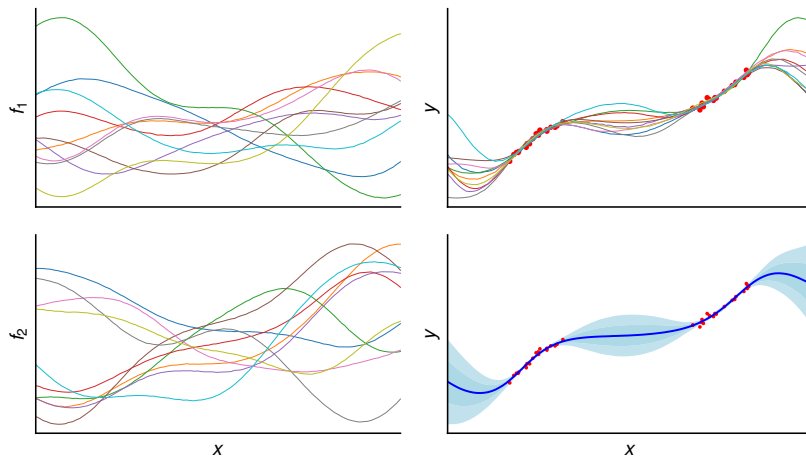


Figure 3: Features from a 2-layer DWP posterior with intermediate width 2: feature samples  $f_1$  (first layer, first output; top left),  $f_2$  (first layer, second output; bottom left), posterior samples (top right), posterior predictive (bottom right)

## G Toy comparison of DGP and DWP posteriors

In this section, we compare intermediate-layer features for trained 2-layer, width-2 DWP and DGP models on a 1-dimensional toy example (following the toy example from Ober & Aitchison (2021)). We plot the intermediate samples for the DGP in Figure 2 and those for the DWP in Figure 3. We observe that the features learned by the DWP are both more interesting, and more varied. This allows for a greater predictive uncertainty in the posterior away from the data. These improved characteristics are due to the increased ability of the DWP to capture true-posterior symmetries.



## H Tables

In Table 3, we include a comparison to the reported results for the 3-layer DIWP with squared exponential kernel from Aitchison et al. (2021); they did not provide ELBOs. However, it should be noted that the specific implementation and architectural details differ significantly from those presented in this paper, and so these results are not directly comparable. Additionally, Aitchison et al. (2021) bases its error bars on paired comparisons to the other methods instead of the standard error bars we use here; we therefore omit the error bars completely.

Table 2: ELBOs per datapoint. We report mean plus or minus one standard error over the splits.

{dataset} - {depth}	DWP	DGP
boston - 2	<b>-0.33 ± 0.00</b>	-0.38 ± 0.01
	<b>-0.34 ± 0.01</b>	-0.40 ± 0.01
	<b>-0.36 ± 0.01</b>	-0.43 ± 0.00
	<b>-0.38 ± 0.01</b>	-0.45 ± 0.01
concrete - 2	<b>-0.42 ± 0.00</b>	-0.44 ± 0.00
	<b>-0.43 ± 0.00</b>	-0.47 ± 0.00
	<b>-0.46 ± 0.00</b>	-0.50 ± 0.00
	<b>-0.49 ± 0.00</b>	-0.50 ± 0.00
energy - 2	<b>1.46 ± 0.00</b>	1.44 ± 0.00
	<b>1.44 ± 0.00</b>	1.42 ± 0.00
	<b>1.42 ± 0.00</b>	1.40 ± 0.00
	<b>1.41 ± 0.00</b>	1.39 ± 0.00
kin8nm - 2	-0.16 ± 0.00	<b>-0.15 ± 0.00</b>
	-0.15 ± 0.00	<b>-0.14 ± 0.00</b>
	-0.14 ± 0.00	-0.14 ± 0.00
	-0.14 ± 0.00	<b>-0.13 ± 0.00</b>
naval - 2	3.75 ± 0.07	<b>3.91 ± 0.08</b>
	3.76 ± 0.11	3.82 ± 0.10
	3.68 ± 0.07	<b>3.91 ± 0.06</b>
	3.65 ± 0.07	<b>3.91 ± 0.10</b>
power - 2	0.03 ± 0.00	0.03 ± 0.00
	0.03 ± 0.00	0.03 ± 0.00
	0.03 ± 0.00	0.03 ± 0.00
	<b>0.03 ± 0.00</b>	0.02 ± 0.00
protein - 2	-1.07 ± 0.00	-1.07 ± 0.00
	-1.04 ± 0.00	-1.04 ± 0.00
	-1.02 ± 0.00	-1.02 ± 0.00
	-1.01 ± 0.00	<b>-1.00 ± 0.00</b>
wine - 2	-1.18 ± 0.00	-1.18 ± 0.00
	<b>-1.18 ± 0.00</b>	-1.19 ± 0.00
	<b>-1.18 ± 0.00</b>	-1.19 ± 0.00
	-1.19 ± 0.00	-1.19 ± 0.00
yacht - 2	<b>2.02 ± 0.01</b>	1.80 ± 0.03
	<b>1.89 ± 0.02</b>	1.59 ± 0.01
	<b>1.74 ± 0.02</b>	1.48 ± 0.02
	<b>1.65 ± 0.01</b>	1.37 ± 0.03

Table 3: Average test log likelihoods. We report mean plus or minus one standard error over the splits, along with quoted results for the DIWP model from Aitchison et al. (2021). We only directly compare between DWP and DGP models and do not quote error bars for the DIWP due to the differences noted above.

{dataset} - {depth}	DWP	DGP	
boston - 2	-2.39 ± 0.05	-2.42 ± 0.05	-
	3 -2.38 ± 0.04	-2.41 ± 0.05	-
	4 -2.38 ± 0.04	-2.41 ± 0.04	-2.40
	5 -2.39 ± 0.04	-2.44 ± 0.04	-
concrete - 2	-3.13 ± 0.02	-3.10 ± 0.02	-
	3 -3.11 ± 0.02	-3.08 ± 0.02	-
	4 -3.12 ± 0.02	-3.11 ± 0.02	-3.08
	5 -3.13 ± 0.01	-3.14 ± 0.02	-
energy - 2	-0.70 ± 0.03	-0.70 ± 0.03	-
	3 -0.71 ± 0.03	-0.70 ± 0.03	-
	4 -0.70 ± 0.03	-0.70 ± 0.03	-0.70
	5 -0.70 ± 0.03	-0.70 ± 0.03	-
kin8nm - 2	1.35 ± 0.00	1.35 ± 0.00	-
	3 1.37 ± 0.00	1.38 ± 0.01	-
	4 <b>1.40 ± 0.00</b>	1.38 ± 0.01	1.01
	5 1.40 ± 0.01	1.39 ± 0.01	-
naval - 2	8.11 ± 0.10	8.24 ± 0.08	-
	3 8.22 ± 0.07	8.13 ± 0.14	-
	4 8.18 ± 0.05	8.27 ± 0.05	5.92
	5 8.21 ± 0.05	8.31 ± 0.07	-
power - 2	-2.77 ± 0.01	-2.78 ± 0.01	-
	3 -2.77 ± 0.01	-2.77 ± 0.01	-
	4 -2.77 ± 0.01	-2.78 ± 0.01	-2.78
	5 -2.77 ± 0.01	-2.78 ± 0.01	-
protein - 2	<b>-2.81 ± 0.00</b>	-2.82 ± 0.00	-
	3 -2.78 ± 0.00	<b>-2.77 ± 0.00</b>	-
	4 -2.73 ± 0.01	-2.75 ± 0.01	-2.74
	5 -2.73 ± 0.00	-2.73 ± 0.00	-
wine - 2	-0.96 ± 0.01	-0.96 ± 0.01	-
	3 -0.96 ± 0.01	-0.96 ± 0.01	-
	4 -0.96 ± 0.01	-0.96 ± 0.01	-1.00
	5 -0.96 ± 0.01	-0.96 ± 0.01	-
yacht - 2	<b>-0.04 ± 0.09</b>	-0.43 ± 0.10	-
	3 <b>-0.16 ± 0.07</b>	-0.65 ± 0.04	-
	4 <b>-0.43 ± 0.10</b>	-0.70 ± 0.04	-0.39
	5 <b>-0.43 ± 0.10</b>	-0.88 ± 0.08	-

Table 4: Average runtime (seconds) for an epoch of boston and protein. Error bars are negligible.

{dataset} - {depth}	DWP	DGP
boston - 2	0.214	0.318
	5 0.351	0.668
protein - 2	0.942	0.987
	5 1.815	2.017