Supplemental material – Conditionally Independent Multiresolution Gaussian Processes

Jalil Taghia
Systems and Control Division
Department of Information Technology, Uppsala University
Uppsala, Sweden
jalil.taghia@it.uu.se

Thomas B. Schön
Systems and Control Division
Department of Information Technology, Uppsala University
Uppsala, Sweden
thomas.schon@it.uu.se

A Prior model

This section describes our choice of the prior model parameters, and details of their initializations.

A.1 Prior over basis-axis scales

We assign a product of zero-mean Gaussian densities conditional on the basis-axis scale-precision variables as the prior over basis-axis scales,

$$p(\underline{\boldsymbol{a}} \mid \boldsymbol{r}) = \prod_{j=0}^{m} \prod_{l=1}^{|\mathcal{T}^{(j)}|} \prod_{i=1}^{p} \mathcal{N}\left(a_{i,l}^{(j)}; 0, \left(\frac{r_i}{S^{(j)}\Big(\sqrt{\lambda_i^{(j)}(\boldsymbol{\tau}_l^{(j)})}\Big)}\right)^{-1}\right), \tag{A.1}$$

where $S^{(j)}\big(\,\cdot\,\big)$ is the spectral density of the covariance function and $\lambda_i^{(j)}(\boldsymbol{\tau}_l^{(j)})$ is the eigenvalue of the basis function $\phi_i^{(j)}(\cdot)$ at resolution j. There are various choices of covariance functions [12]. Among them, we are interested in those for which $S(\nu) \to 0$ for all $\nu \to \infty$, that is the case for most classes of covariance functions, including Matérn and exponentiated quadratic covariance functions. We have indicated spectral densities with indexing on j, as in general, we are free to choose different covariance functions at different resolutions. Similarly, there are various choices of basis functions which are interpretable as GPs. As discussed in the paper, the choice of basis functions can in general be resolution-specific.

Our choice of prior implies that $a_{i,l}^{(j)}$ are resolution-region specific, which means that regardless of the resolution or the region the prior must be initialized with zero-mean even though the posterior mean is non-zero.

A.2 Prior over basis axes

Considering the possible index permutation across resolutions, we assign a product of independent Bingham densities [3; 11], conditional on the binary index-mapping matrix $\boldsymbol{\Gamma}$, as the prior over basis

Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics (AISTATS) 2019, Naha, Okinawa, Japan.

axes

$$p(\boldsymbol{U} \mid \boldsymbol{\Gamma}) = \prod_{i=1}^{p}\prod_{k=1}^{p}\Big[\mathcal{B}\big(\boldsymbol{u}_i;\ \mathbf{B}'_k\big)\Big]^{\Gamma_{ik}} = \prod_{i=1}^{p}\prod_{k=1}^{p}\left[\frac{1}{\mathcal{C}(\boldsymbol{\kappa}'_k)}\exp\left(\boldsymbol{u}_i^{\top}\mathbf{B}'_k\boldsymbol{u}_i\right)\right]^{\Gamma_{ik}}. \tag{A.2}$$

Here, $\mathbf{B}'_k = \mathbf{M}'_k \times \mathrm{diag}[\boldsymbol{\kappa}'_k] \times {\mathbf{M}'_k}^{\top}$, and the pair of $\mathbf{M}'_k = (\boldsymbol{\mu}'_{k1}, \ldots, \boldsymbol{\mu}'_{kd_y})$, $\boldsymbol{\mu}'_{kd_y} \in \mathcal{S}^{d_y-1}$, $\boldsymbol{\kappa}'_k = (\kappa'_{k1}, \ldots, \kappa'_{kd_y})^{\top}$ are given by the eigendecomposition of $\mathbf{B}'_k$ and $\mathcal{C}(\boldsymbol{\kappa}'_k)$ is the Bingham normalization factor, which is algebraically problematic, but the saddle-point approximation [9] provides an accurate numerical result.

Notice that, at resolution $j > 0$, $\mathbf{B}'_k$ is given by the posterior hyper-parameter from the previous resolution $j-1$. At resolution $j = 0$, we set simply $\mathbf{B}'_k = \mathbf{B}_0 = \mathbf{0}$.

A.3 Prior over basis-axis scale-precision

Considering the possible index permutation across resolutions, we express the prior over precision of the basis scales as conditional on the binary index-mapping matrix $\boldsymbol{\Gamma}$ using Gamma densities

$$p(\boldsymbol{r} \mid \boldsymbol{\Gamma}) = \prod_{i=1}^{p}\prod_{k=1}^{p}\big[\mathcal{G}\big(r_i;\ \alpha'_k, \beta'_k\big)\big]^{\Gamma_{ik}}, \tag{A.3}$$

where α'_k and β'_k are the Gamma densities shape and inverse scale hyper-parameters. At resolution $j > 0$, α'_k and β'_k are the posterior hyper-parameters computed from resolution $j-1$. At $j = 0$, in absence of prior data, non-informative distributions may be assigned with $\alpha'_k \to 0$, but β'_k may still be assigned an informative value. Values of β'_k for which $\alpha'_k/\beta'_k \to 0$ reduces the overall influence of the prior toward a non-regularized basis function expansion.

A.4 Prior over basis-axis index mapping

As discussed earlier, the index-mapping binary matrix $\boldsymbol{\Gamma}$ has exactly one element $\Gamma_{ik} = 1$ in each row and each column, indicating that the basis axis identified by index k in the previous resolution $j-1$ is identical to the basis axis denoted by index i at the current resolution j. The prior probability mass for these index-mapping variables is assigned as totally non-informative, except for the uniqueness requirement

$$\sum_{k=1}^{p} p(\Gamma_{ik} = 1) = 1, \qquad \forall i \in \{1, \ldots, p\}, \tag{A.4a}$$

$$\sum_{i=1}^{p} p(\Gamma_{ik} = 1) = 1, \qquad \forall k \in \{1, \ldots, p\}. \tag{A.4b}$$

A.5 Prior over overall bias and residual noise precision

We assign product of Gaussian-Gamma densities over the joint distribution of the overall bias and the residual noise precision as

$$p(\underline{\boldsymbol{b}}, \underline{\gamma}) = \prod_{j=0}^{m}\prod_{l=1}^{|\mathcal{T}^{(j)}|}\mathcal{N}\Big(\boldsymbol{b}_l^{(j)}; \boldsymbol{\nu}_{\mathfrak{o}_l}^{(j)}, \frac{1}{\vartheta_{\mathfrak{o}_l}^{(j)}\gamma_l^{(j)}}\Big)\mathcal{G}\Big(\gamma_l^{(j)}; \mathfrak{c}_{\mathfrak{o}_l}^{(j)}, \mathfrak{d}_{\mathfrak{o}_l}^{(j)}\Big). \tag{A.5}$$

In the absence of prior information, a non-informative prior must be applied by setting $\boldsymbol{\nu}_{\mathfrak{o}_l}^{(j)} = \mathbf{0}$ and $\vartheta_{\mathfrak{o}_l}^{(j)} \to 0$. The hyper-parameters $\mathfrak{c}_{\mathfrak{o}_l}^{(j)}$ and $\mathfrak{d}_{\mathfrak{o}_l}^{(j)}$ are shape and inverse scale parameters of the corresponding Gamma distributions. In the absence of prior information, a noninformative distribution is assigned by $\mathfrak{c}_{\mathfrak{o}_l}^{(j)} \to 0$, but $\mathfrak{d}_{\mathfrak{o}_l}^{(j)}$ may still be assigned an informative value to indicate the most likely value (mode), $\mathfrak{d}_{\mathfrak{o}_l}^{(j)}/(\mathfrak{c}_{\mathfrak{o}_l}^{(j)} + 1)$, for the residual variance which has an inverse-gamma distribution.

B Posterior model

In this section, we summarize the optimized posterior distribution which is obtained by maximizing the lower bound $\mathcal{L}$ in (8). For ease of notation, we use: $\langle \cdot \rangle_{q(\cdot)} \equiv \langle \cdot \rangle$ wherever possible. Descriptive statistics of the posterior distributions are summarized in Appendix C.

B.1 Conditional posterior over basis-axis scales

Optimized conditional posterior distribution of $q(\underline{\boldsymbol{a}}\,|\,\boldsymbol{U})$ is given by the following product of Gaussian densities

$$q(\underline{\boldsymbol{a}}\,|\,\boldsymbol{U}) = \prod_{j=0}^{m} \prod_{l=1}^{|\mathcal{T}^{(j)}|} \prod_{i=1}^{p} \mathcal{N}\big(a_{i,l}^{(j)};\, \mathfrak{m}_{i,l}^{(j)}(\boldsymbol{u}_i), \mathfrak{v}_{i,l}^{(j)^{-1}}\big),$$

with the mean value $\mathfrak{m}_{i,l}^{(j)}(\boldsymbol{u}_i)$ as the function of the basis axis vector $\boldsymbol{u}_i$ and the precision $\mathfrak{v}_{i,l}^{(j)}$ given by

$$\begin{aligned}
\mathfrak{v}_{i,l}^{(j)} &= \frac{\left\langle r_i^{(j)} \right\rangle}{S^{(j)}\Big(\sqrt{\lambda_i^{(j)}(\widehat{\boldsymbol{\tau}}_l^{(j)})}\Big)} + \left\langle \gamma_l^{(j)} \right\rangle \sum_{t \in \mathcal{T}_l^{(j)}} \Big(\phi_i^{(j)}(\boldsymbol{x}_t, \widehat{\boldsymbol{\tau}}_l^{(j)})\Big)^2, \\
\mathfrak{m}_{i,l}^{(j)}(\boldsymbol{u}_i) &= \zeta_{i,l}^{(j)} \boldsymbol{u}_i^\top \widetilde{\boldsymbol{z}}_{i,l}^{(j)},
\end{aligned}$$

where we have defined

$$\begin{aligned}
\zeta_{i,l}^{(j)} &= \frac{\left\langle \gamma_l^{(j)} \right\rangle}{\mathfrak{v}_{i,l}^{(j)}}, \\
\widetilde{\boldsymbol{z}}_{i,l}^{(j)} &= \begin{cases} \widetilde{\boldsymbol{y}}_i = \sum_{t \in \mathcal{T}^{(0)}} \phi_i^{(0)}(\boldsymbol{x}_t, \widehat{\boldsymbol{\tau}}^{(0)}) \Big(\boldsymbol{y}_t - \bar{\widetilde{\boldsymbol{y}}}_i\Big), & \forall j = 0 \\ \sum_{t \in \mathcal{T}_l^{(j)}} \phi_i^{(j)}(\boldsymbol{x}_t, \widehat{\boldsymbol{\tau}}_l^{(j)}) \Big(\Big\langle \boldsymbol{z}_{t,l}^{(j)} \Big\rangle - \bar{\widetilde{\boldsymbol{z}}}_{i,t,l}^{(j)}\Big), & \forall j \geq 1 \end{cases}, \\
\bar{\widetilde{\boldsymbol{y}}}_i &= \langle \boldsymbol{b} \rangle + \sum_{k \neq i}^{p} \langle a_k \boldsymbol{u}_k \rangle \, \phi_k^{(0)}(\boldsymbol{x}_t, \widehat{\boldsymbol{\tau}}^{(0)}), \\
\bar{\widetilde{\boldsymbol{z}}}_{i,t,l}^{(j)} &= \left(\sum_{j'=0}^{j-1} \bar{\mathbf{f}}_{t,l}^{j'} \right) + \left\langle \boldsymbol{b}_l^{(j)} \right\rangle + \sum_{k \neq i}^{p} \left\langle a_{k,l}^{(j)} \boldsymbol{u}_k \right\rangle \phi_k^{(j)}(\boldsymbol{x}_t, \widehat{\boldsymbol{\tau}}_l^{(j)}).
\end{aligned}$$

The seemingly complicated form of this result makes intuitively good sense: The conditional expected value of the basis-axis scale variables, given by $\mathfrak{m}_{i,l}^{(j)}(\boldsymbol{u}_i)$, is determined by the mean predictions from the previous resolution plus the remaining part of the observed (or latent for $j > 0$) vector that is not already explained by its components along the other basis axes. The conditional expected value is scaled by $\zeta_{i,l}^{(j)}$, which is the currently estimated proportion of the variance of the observed data (or latent variables for $j > 0$) that is explained by the basis-axis scale variables in the ith axis, in relation to the total variance that also includes the residual noise component along this axis.

B.2 Posterior over basis axes

Given a posterior distribution $q(\boldsymbol{a}_l^{(j)} \mid \boldsymbol{U})$ and using $q(\boldsymbol{a}_l^{(j)}, \boldsymbol{U}) = q(\boldsymbol{U}) q(\boldsymbol{a}_l^{(j)} \mid \boldsymbol{U})$, it can be shown that the optimized posterior distribution $q(\boldsymbol{U})$ is given by the product of Bingham densities

$$q(\boldsymbol{U}) = \prod_{i=1}^{p} \mathcal{B}\big(\boldsymbol{u}_i;\, \mathbf{B}_i\big) = \prod_{i=1}^{p} \frac{1}{\mathcal{C}(\boldsymbol{\kappa}_i)} \exp\left(\boldsymbol{u}_i^\top \mathbf{M}_i \times \mathrm{diag}[\boldsymbol{\kappa}_i] \times \mathbf{M}_i^\top \boldsymbol{u}_i\right),$$

where as before the pair of $\boldsymbol{\kappa}_i$ and $\mathbf{M}_i$ are eigenvalues and the corresponding eigenvectors of

$$\mathbf{B}_i = \left(\sum_{k=1}^{p} \langle \Gamma_{ik} \rangle \, \mathbf{B}'_k \right) + \sum_{l=1}^{|\mathcal{T}^{(j)}|} \frac{\left\langle \gamma_l^{(j)} \right\rangle}{2} \zeta_{i,l}^{(j)} \widetilde{\boldsymbol{z}}_{i,l}^{(j)} \widetilde{\boldsymbol{z}}_{i,l}^{(j)\top}.$$

Note the first term where Bingham's posterior hyper-parameter from the previous resolution, $\mathbf{B}'_k$, has been weighted by $\langle \Gamma_{ik} \rangle$. This ensures that the axis indices remain aligned throughout and hence allows for recursive (successive) learning of these parameters.

B.3 Posterior over basis-scale precision

The optimized posterior distribution of the latent variables $\boldsymbol{r}$ is given by the product of Gamma densities as

$$q(\boldsymbol{r}) = \prod_{i=1}^{p} \mathcal{G}\big(r_i;\ \alpha_i, \beta_i\big),$$

where α_i and β_i are the shape and inverse scale posterior hyper-parameters of Gamma density given by

$$\alpha_i = \left(\sum_{k=1}^{p} \langle \Gamma_{ik} \rangle\, \alpha'_k\right) + \frac{|\mathcal{T}^{(j)}|}{2},$$
$$\beta_i = \left(\sum_{k=1}^{p} \langle \Gamma_{ik} \rangle\, \beta'_k\right) + \frac{1}{2} \sum_{l=1}^{|\mathcal{T}^{(j)}|} \frac{\left\langle \left(a_{i,l}^{(j)}\right)^2 \right\rangle}{S^{(j)}\Big(\sqrt{\lambda_i^{(j)}(\widehat{\boldsymbol{\tau}}_l^{(j)})}\Big)},$$

where α'_k and β'_k are posterior hyper-parameters from the previous resolution, k, weighted by the posterior mean of the basis-axis index-mapping variable, $\langle \Gamma_{ik} \rangle$.

B.4 Posterior over basis-axis index mapping

The optimized posterior distribution of the latent variables $q(\boldsymbol{\Gamma})$ is given by $q(\boldsymbol{\Gamma}) = \prod_{i=1}^{p} \prod_{k=1}^{p} \omega_{ik}^{\Gamma_{ik}}$, where the probability parameters are normalized using scale factors η_i and η_k as

$$\omega_{ik} = \eta_i \eta_k \widetilde{\omega}_{ik},$$
$$\text{such that}: \begin{cases} \sum_{k=1}^{p} \omega_{ik} = 1,\ \forall i \in \{1, \ldots, p\} \\ \sum_{i=1}^{p} \omega_{ik} = 1,\ \forall k \in \{1, \ldots, p\} \end{cases},$$

to satisfy the prior requirements, Eq. (A.4), with

$$\log \widetilde{\omega}_{ik} = \left\langle \boldsymbol{u}_i^{\top} \mathbf{B}'_k \boldsymbol{u}_i \right\rangle - \log \mathcal{C}(\mathbf{B}'_k) + \alpha'_k \log \beta'_k - \log \digamma(\alpha'_k) + (\alpha'_k - 1) \left\langle \log r_i \right\rangle - \beta'_k \left\langle r_i \right\rangle,$$

where $\digamma(\cdot)$ denotes the digamma function. We may view $\log \widetilde{\omega}_{ik}$ as a logarithmic similarity measure between the kth prior axes at the previous resolution and ith posterior axes at the current resolution.

B.5 Posterior distribution of the latent remainder term

The optimal posterior distribution of $q(\underline{\boldsymbol{z}_{\mathcal{T}}})$ is given by

$$q(\underline{\boldsymbol{z}_{\mathcal{T}}}) = \prod_{j=1}^{m} \prod_{l=1}^{|\mathcal{T}^{(j)}|} \prod_{t \in \mathcal{T}_l^{(j)}} \mathcal{N}\Big(\boldsymbol{z}_{t,l}^{(j)}; \Big\langle \bar{\boldsymbol{z}}_{t,l}^{(j)} \Big\rangle, \Big\langle \gamma^{(j)} \Big\rangle^{-1}\Big),$$
$$\Big\langle \bar{\boldsymbol{z}}_{t,l}^{(j)} \Big\rangle = \sum_{j'=0}^{j-1} \bar{\mathbf{f}}_{t,l}^{j'} + \Big\langle \boldsymbol{b}_l^{(j)} \Big\rangle + \sum_{i=1}^{p} \Big\langle a_{i,l}^{(j)} \boldsymbol{u}_i \Big\rangle \phi_i^{(j)}(\boldsymbol{x}_t, \widehat{\boldsymbol{\tau}}_l^{(j)}).$$

B.6 Posterior distribution of overall bias and residual noise precision

The optimized posterior of the joint distribution of the mean vector and the residual noise is given by

$$q(\underline{\boldsymbol{b}}, \underline{\gamma}) = \prod_{j=0}^{m} \prod_{l=1}^{|\mathcal{T}^{(j)}|} \mathcal{N}\Big(\boldsymbol{b}_l^{(j)}; \boldsymbol{\nu}_l^{(j)}, \frac{1}{\vartheta_l^{(j)} \gamma_l^{(j)}}\Big) \mathcal{G}\Big(\gamma_l^{(j)}; \mathfrak{c}_l^{(j)}, \mathfrak{d}_l^{(j)}\Big),$$

with the posterior hyper-parameters given by

$$\vartheta_l^{(j)} = \vartheta_{\mathfrak{o}l}^{(j)} + |\mathcal{T}_l^{(j)}|,$$

$$\boldsymbol{\nu}_l^{(j)} = \frac{1}{\vartheta_l^{(j)}} \left(\vartheta_{\mathfrak{o}l}^{(j)} \boldsymbol{\nu}_{\mathfrak{o}l}^{(j)} + \bar{\boldsymbol{\nu}}_l^{(j)} \right),$$

$$\mathfrak{c}_l^{(j)} = \mathfrak{c}_{\mathfrak{o}l}^{(j)} + \frac{d_y}{2} |\mathcal{T}_l^{(j)}|,$$

$$\mathfrak{d}_l^{(j)} = \mathfrak{d}_{\mathfrak{o}l}^{(j)} + \frac{1}{2} \bar{\mathfrak{d}}_l^{(j)},$$

where $\bar{\boldsymbol{\nu}}^{(0)}$ and $\bar{\mathfrak{d}}^{(0)}$ are given by

$$\bar{\boldsymbol{\nu}}^{(0)} = \sum_{t \in \mathcal{T}^{(0)}} \left(\boldsymbol{y}_t - \sum_{i=1}^{p} \langle a_i \boldsymbol{u}_i \rangle \, \phi_i^{(0)}(\boldsymbol{x}_t, \widehat{\boldsymbol{\tau}}^{(0)}) \right),$$

$$\bar{\mathfrak{d}}^{(0)} = \vartheta_{\mathfrak{o}}^{(0)} \|\boldsymbol{\nu}_{\mathfrak{o}}^{(0)}\|^2 - \vartheta^{(0)} \|\boldsymbol{\nu}^{(0)}\|^2 +$$
$$+ \sum_{t \in \mathcal{T}^{(0)}} \left(\left\| \boldsymbol{y}_t - \sum_{i=1}^{p} \phi_i^{(0)}(\boldsymbol{x}_t, \boldsymbol{\tau}^{(0)}) \, \langle a_i \boldsymbol{u}_i \rangle \right\|^2 + \sum_{i=1}^{p} \left(\phi_i^{(0)}(\boldsymbol{x}_t, \boldsymbol{\tau}^{(0)}) \right)^2 \left\langle \| a_i \boldsymbol{u}_i - \langle a_i \boldsymbol{u}_i \rangle \|^2 \right\rangle \right).$$

and similarly $\bar{\boldsymbol{\nu}}_l^{(j)}$ and $\bar{\mathfrak{d}}_l^{(j)}$, $\forall j \geq 1$, are given by

$$\bar{\boldsymbol{\nu}}_l^{(j)} = \sum_{t \in \mathcal{T}_l^{(j)}} \left(\left\langle \boldsymbol{z}_{t,l}^{(j)} \right\rangle - \sum_{j'=0}^{j-1} \bar{\mathbf{f}}_{t,l}^{(j')} - \sum_{i=1}^{p} \left\langle a_{i,l}^{(j)} \boldsymbol{u}_i \right\rangle \phi_i^{(j)}(\boldsymbol{x}_t, \widehat{\boldsymbol{\tau}}_l^{(j)}) \right),$$

$$\bar{\mathfrak{d}}_l^{(j)} = \vartheta_{\mathfrak{o}l}^{(j)} \|\boldsymbol{\nu}_{\mathfrak{o}l}^{(j)}\|^2 - \vartheta_l^{(j)} \|\boldsymbol{\nu}_l^{(j)}\|^2 +$$
$$+ \sum_{t \in \mathcal{T}_l^{(j)}} \left(\left\| \left\langle \boldsymbol{z}_{t,l}^{(j)} \right\rangle - \sum_{i=1}^{p} \phi_i^{(j)}(\boldsymbol{x}_t, \boldsymbol{\tau}_l^{(j)}) \left\langle a_{i,l}^{(j)} \boldsymbol{u}_i \right\rangle - \sum_{j'=0}^{j-1} \bar{\mathbf{f}}_{t,l}^{(j')} \right\|^2 \right.$$
$$+ \left\langle \left\| \boldsymbol{z}_{t,l}^{(j)} - \left\langle \boldsymbol{z}_{t,l}^{(j)} \right\rangle \right\|^2 \right\rangle + \sum_{j'=0}^{j-1} \mathrm{E}\left[\left\| \mathbf{f}_{t,l}^{(j')} - \bar{\mathbf{f}}_{t,l}^{(j')} \right\|^2 \right]$$
$$\left. + \sum_{i=1}^{p} \left(\phi_i^{(j)}(\boldsymbol{x}_t, \boldsymbol{\tau}_l^{(j)}) \right)^2 \left\langle \| a_{i,l}^{(j)} \boldsymbol{u}_i - \left\langle a_{i,l}^{(j)} \boldsymbol{u}_i \right\rangle \|^2 \right\rangle \right).$$

Estimating the noise precision at resolution $j \geq 1$ also includes the second central moments of the predictive processes and the latent remainder terms at the previous resolutions.

C Descriptive statistics

Descriptive statistics of the posterior distributions $q(r_i)$, $q(a_{i,l}^{(j)} \mid \boldsymbol{u}_i)$, and $q(\boldsymbol{z}_{\mathcal{T}^{(j)}})$ are conveniently given by the known statistics of the Gamma and Gaussian distributions. For $q(\boldsymbol{\Gamma})$, we have the standard result of $\langle \Gamma_{ik} \rangle = \omega_{ik}$. With a special notational treatment for $j = 0$, the required statistics for the joint posterior $q(\boldsymbol{u}_i, a_{i,l}^{(j)})$, $\forall j$, are summarized as

$$\left\langle \boldsymbol{u}_i \boldsymbol{u}_i^\top \right\rangle = \sum_{d=1}^{d_y} \rho_{id}(\boldsymbol{\kappa}_i) \boldsymbol{\mu}_{id} \boldsymbol{\mu}_{id}^\top,$$

$$\left\langle a_{i,l}^{(j)} \boldsymbol{u}_i \right\rangle = \zeta_{i,l}^{(j)} \left\langle \boldsymbol{u}_i \boldsymbol{u}_i^\top \right\rangle \widetilde{\boldsymbol{z}}_{i,l}^{(j)},$$

$$\left\langle \left(a_{i,l}^{(j)} \right)^2 \right\rangle = \frac{1}{\mathfrak{v}_{i,l}^{(j)}} + \left(\zeta_{i,l}^{(j)} \right)^2 \widetilde{\boldsymbol{z}}_{i,l}^{(j)\top} \left\langle \boldsymbol{u}_i \boldsymbol{u}_i^\top \right\rangle \widetilde{\boldsymbol{z}}_{i,l}^{(j)},$$

$$\left\langle \left\| a_{i,l}^{(j)} \boldsymbol{u}_i - \left\langle a_{i,l}^{(j)} \boldsymbol{u}_i \right\rangle \right\|^2 \right\rangle = \frac{1}{\mathfrak{v}_{i,l}^{(j)}} + {\zeta_{i,l}^{(j)}}^2 \left(\widetilde{\boldsymbol{z}}_{i,l}^{(j)} \right)^\top \left(\left\langle \boldsymbol{u}_i \boldsymbol{u}_i^\top \right\rangle - \left\langle \boldsymbol{u}_i \boldsymbol{u}_i^\top \right\rangle \left\langle \boldsymbol{u}_i \boldsymbol{u}_i^\top \right\rangle \right) \widetilde{\boldsymbol{z}}_{i,l}^{(j)},$$

where $\rho_{id}(\boldsymbol{\kappa}_i)$ is the d-th element of $\boldsymbol{\rho}_i(\boldsymbol{\kappa}_i)$ given by

$$\boldsymbol{\rho}_i(\boldsymbol{\kappa}_i) = \frac{\partial \log \mathcal{C}(\boldsymbol{\kappa}_i)}{\partial \boldsymbol{\kappa}_i}, \quad \forall i \in \mathcal{P}.$$

The saddle-point approximation of Kume and Wood [9] is used to calculate the derivatives above.

The mean and the second central moment of the predictive processes can be computed using Eq. (D.1c) and (D.1d),

$$\begin{aligned}
&\mathrm{E}\left[\mathbf{f}_{t,l}^{(j)}\right] \equiv \bar{\mathbf{f}}_{t,l}^{(j)} = \bar{\mathbf{f}}_{l}^{(j)}(\boldsymbol{x}_t), \\
&\mathrm{E}\left[\left\|\mathbf{f}_{t,l}^{(j)} - \bar{\mathbf{f}}_{t,l}^{(j)}\right\|^2\right] = \mathrm{trace}\left[\bar{\mathbf{F}}_{l}^{(j)}(\boldsymbol{x}_t)\right], \quad \forall \boldsymbol{x}_t \in \mathcal{T}_l^{(j)}.
\end{aligned}$$

D Predictive process

For a new test input $\boldsymbol{x}^*$, we shall first determine if we know to which region it belongs in each resolution. If such information is available the required statistics of the approximate predictive process at $\boldsymbol{x}^*$ can be computed from the sum of their contributions across all resolutions, as

$$\mathrm{E}\left[p(\boldsymbol{f}(\boldsymbol{x}^*) \mid \boldsymbol{x}^*, \underline{\boldsymbol{x}_{\mathcal{T}}}, \boldsymbol{y}_{\mathcal{T}}, \underline{\boldsymbol{z}_{\mathcal{T}}})\right] = \sum_{j=0}^{m} \bar{\mathbf{f}}^{(j)}(\boldsymbol{x}^*), \tag{D.1a}$$

$$\mathrm{Cov}\left[p(\boldsymbol{f}(\boldsymbol{x}^*) \mid \boldsymbol{x}^*, \underline{\boldsymbol{x}_{\mathcal{T}}}, \boldsymbol{y}_{\mathcal{T}}, \underline{\boldsymbol{z}_{\mathcal{T}}})\right] = \sum_{j=0}^{m} \bar{\mathbf{F}}^{(j)}(\boldsymbol{x}^*), \tag{D.1b}$$

where $\bar{\mathbf{f}}_{l}^{(j)}(\boldsymbol{x}^*)$ and $\bar{\mathbf{F}}_{l}^{(j)}(\boldsymbol{x}^*)$ are given by

$$\bar{\mathbf{f}}_{l}^{(j)}(\boldsymbol{x}^*) = \left\langle \boldsymbol{b}_{l}^{(j)} \right\rangle + \sum_{i=1}^{p} \left\langle a_{i,l}^{(j)} \boldsymbol{u}_i \right\rangle \phi_i^{(j)}(\boldsymbol{x}^*, \widehat{\boldsymbol{\tau}}_l^{(j)}), \tag{D.1c}$$

$$\begin{aligned}
\bar{\mathbf{F}}_{l}^{(j)}(\boldsymbol{x}^*) = &\left\langle \boldsymbol{b}_{l}^{(j)} \boldsymbol{b}_{l}^{(j)\top} \right\rangle - \left\langle \boldsymbol{b}_{l}^{(j)} \right\rangle \left\langle \boldsymbol{b}_{l}^{(j)} \right\rangle^{\top} + \\
&+ \sum_{i=1}^{p} \left(\phi_i^{(j)}(\boldsymbol{x}^*, \widehat{\boldsymbol{\tau}}_l^{(j)})\right)^2 \times \left[\left\langle \left(a_{i,l}^{(j)}\right)^2 \right\rangle \left\langle \boldsymbol{u}_i \boldsymbol{u}_i^{\top} \right\rangle - \left\langle a_{i,l}^{(j)} \boldsymbol{u}_i \right\rangle \left\langle a_{i,l}^{(j)} \boldsymbol{u}_i \right\rangle^{\top} \right].
\end{aligned} \tag{D.1d}$$

In many applications however we may indeed not know the position of $\boldsymbol{x}^*$ in the training index sets, $\mathcal{T}^{(j)}, \forall j$,—in other words we may not know to which region $\boldsymbol{x}^*$ belongs at a given resolution. In such cases, since the basis axes are shared across all resolutions and learnt in a group fashion, predictions are made only from $j = 0$,

$$\mathrm{E}\left[p(\boldsymbol{f}(\boldsymbol{x}^*) \mid \boldsymbol{x}^*, \underline{\boldsymbol{x}_{\mathcal{T}}}, \boldsymbol{y}_{\mathcal{T}}, \underline{\boldsymbol{z}_{\mathcal{T}}})\right] = \bar{\mathbf{f}}^{(0)}(\boldsymbol{x}^*), \tag{D.2a}$$

$$\mathrm{Cov}\left[p(\boldsymbol{f}(\boldsymbol{x}^*) \mid \boldsymbol{x}^*, \underline{\boldsymbol{x}_{\mathcal{T}}}, \boldsymbol{y}_{\mathcal{T}}, \underline{\boldsymbol{z}_{\mathcal{T}}})\right] = \bar{\mathbf{F}}^{(0)}(\boldsymbol{x}^*). \tag{D.2b}$$

We emphasize that, among others, this is one of the advantages of the conditional independence over models with full independence.

E Optimization of basis interval variables

The basis interval variables $\boldsymbol{\tau}_l^{(j)} = (\tau_{1,l}^{(j)}, \ldots, \tau_{d_x,l}^{(j)})^{\top}$ are optimized using maximum likelihood estimation, as an analytical solution within our standard variational inference may not exist in general form for various choices of basis functions and spectral densities. The optimized point estimate values are given from

$$\begin{aligned}
&\widehat{\tau}_{d,l}^{(j)} = \underset{\tau_{d,l}^{(j)}}{\mathrm{argmax}}\ h(\tau_{d,l}^{(j)}), \\
&\text{s.t.} \quad L_{d_x,l}^{(j)} < \tau_{d,l}^{(j)} < L_{d_x,l}^{(j)} + \frac{p}{L_{d_x,l}^{(j)}},
\end{aligned}$$

where $L^{(j)}_{d_x,l}$ is the input range at (j,l), and

$$h(\tau^{(j)}_{d,l}) \propto h_{\text{prior}}(\tau^{(j)}_{d,l}) + h_{\text{likelihood}}(\tau^{(j)}_{d,l}),$$

where $h_{\text{prior}}(\tau^{(j)}_{d,l})$ includes all relevant terms from the prior,

$$h_{\text{prior}}(\tau^{(j)}_{d,l}) = -\frac{1}{2}\sum_{i=1}^{p}\left(\log\left(\mathfrak{S}^{(j)}_{d,l,i}(\tau^{(j)}_{d,l})\right) - \frac{\langle r_i\rangle\left\langle\left(a^{(j)}_{i,l}\right)^2\right\rangle}{2\mathfrak{S}^{(j)}_{d,l,i}(\tau^{(j)}_{d,l})}\right),$$

and $h_{\text{likelihood}}(\tau^{(j)}_{d,l})$ includes all relevant terms in the likelihood term,

$$\begin{aligned} h_{\text{likelihood}}(\tau^{(j)}_{d,l}) = -\left\langle\gamma^{(j)}_l\right\rangle \sum_{t=1}^{\mathcal{T}^{(j)}_l}\sum_{i=1}^{p}\Bigg(&\left\|\mathfrak{X}(x^{(j)}_{t,d,i},\tau^{(j)}_{d,l})\right\|^2 + 2\left(\mathfrak{k}^{(j)}_{t,l}\right)^\top\mathfrak{X}(x^{(j)}_{t,d,i},\tau^{(j)}_{d,l}) \\ &+\left\langle\left\|a^{(j)}_{i,l}\boldsymbol{u}_i - \left\langle a^{(j)}_{i,l}\boldsymbol{u}_i\right\rangle\right\|^2\right\rangle\left(\tilde{\phi}^{(j)}_{i,d,l,t}\phi^{(j)}_i(x_{t,d},\tau^{(j)}_{d,l})\right)^2\Bigg), \end{aligned}$$

where we have defined

$$\begin{aligned} \mathfrak{k}^{(j)}_{t,l} &= \left\langle b^{(j)}_l\right\rangle + \sum_{j'=0}^{j-1}\bar{\mathfrak{f}}^{(j')}_{t,l} - \frac{1}{2}\boldsymbol{z}^{(j)}_{t,l}, \\ \mathfrak{S}^{(j)}_{d,l,i}(\tau^{(j)}_{d,l}) &= S^{(j)}\left(\sqrt{\lambda^{(j)}_i(\tau^{(j)}_{d,l}) + \tilde{\lambda}^{(j)}_{i,d,l}}\right), \\ \mathfrak{X}(x^{(j)}_{t,d,i},\tau^{(j)}_{d,l}) &= \left\langle a^{(j)}_{i,l}\boldsymbol{u}_i\right\rangle\tilde{\phi}^{(j)}_{i,d,l,t}\phi^{(j)}_i(x_{t,d},\tau^{(j)}_{d,l}), \\ \tilde{\lambda}^{(j)}_{i,d,l} &= \sum_{k\neq d}^{d_x}\lambda^{(j)}_i(\widehat{\tau'}^{(j)}_{k,l}), \\ \tilde{\phi}^{(j)}_{i,d,l,t} &= \prod_{k\neq d}^{d_x}\phi^{(j)}_i(x_{t,k},\widehat{\tau'}^{(j)}_{k,l}), \end{aligned}$$

where $\widehat{\tau'}^{(j)}_{k,l}$ are the previous optimized values. The optimization problem is solved numerically.

F Algorithm

- Initialize the basis intervals $\tau^{(j)}_l$.

1. Assign priors
   - Initialize the resolution-region-specific prior distributions $p(\underline{\boldsymbol{a}} \mid \boldsymbol{r}), p(\underline{\boldsymbol{b}},\underline{\gamma})$ by setting their hyperparameters to the default values according to Appendices A.1, A.5. During recursive learning the prior hyperparamters remain unaltered and will not be updated.
   - Initialize shared prior distributions $p(\boldsymbol{U} \mid \boldsymbol{\Gamma}), p(\boldsymbol{r} \mid \boldsymbol{\Gamma}), p(\boldsymbol{\Gamma})$ according to Appendices A.2, A.3, A.4. During recursive learning at resolution $j>1$, the posterior hyperparameters from the previous resolution $j-1$ are used as the prior hyperparameters for the current resolution j.
2. Update posteriors
   - Resolution-region-specific posteriors $q(\underline{\boldsymbol{a}} \mid \boldsymbol{U}), q(\underline{\boldsymbol{b}},\underline{\gamma})$ are updated according to Appendices B.1, B.6.
   - Shared posteriors $q(\boldsymbol{U}), p(\boldsymbol{r}), p(\boldsymbol{\Gamma})$ are updated according to Appendices B.2, B.3, B.4.
3. If necessary, update the basis intervals according to Appendix E.

- Repeat steps 1 to 3 until convergence criteria are met.

G Experiment details

This section provides further details on the experiments in Sec. 5.

G.1 Datasets

G.1.1 oes10 and oes97

The datasets oes10 and oes97 were obtained from [13]. The Occupational Employment Survey (OES) datasets contain records from the years of 1997 (OES97) and 2010 (OES10) of the annual Occupational Employment Survey compiled by the US Bureau of Labor Statistics. As described in [13], "each row provides the estimated number of full-time equivalent employees across many employment types for a specific metropolitan area". We selected the same 16 target variables as listed in [13, Table 5]. The remaining 298 and 263 variables serve as the inputs in the case of oes10 and oes97, respectively. Data samples were randomly divided into training and test sets (refer to Table 1).

G.1.2 atp1d and atp7d

The datasets atp1d and atp7d were obtained from [13]. The Airline Ticket Price (ATP) dataset includes the prediction of airline ticket prices. As described in [13], the target variables are either the next day price, atp1d, or minimum price observed over the next seven days atp7d for 6 target flight preferences listed in [13, Table 5]. There are 411 input variables in each case. The inputs for each sample are values considered to be useful for prediction of the airline ticket prices for a specific departure date, for example, the number of days between the observation date and the departure date, or the boolean variables for day-of-the-week of the observation date. Data samples were randomly divided into training and test sets (refer to Table 1).

G.1.3 scm1d, scm1d-a and scm20d

The datasets scm1d and scm20d were obtained from [13]. The Supply Chain Management (SCM) datasets are derived from the Trading Agent Competition in Supply Chain Management (TAC SCM) tournament from 2010. As described in [13], each row corresponds to an observation day in the tournament. There are 280 input variables in these datasets which are observed prices for a specific tournament day. The datasets contain 16 regression targets, where each target corresponds to the next day mean price scm1d or mean price for 20 days in the future scm20d for each product [13, Table 5]. Dataset scm1d-a is a subset of scm1d which includes the first 3000 samples. Data samples were randomly divided into training and test sets (refer to Table 1).

G.1.4 naval

The dataset naval [4] was obtained from UCI Machine Learning Repository[1]. The input variables are 16-dimensional feature vectors containing the gas turbine (GT) measures at steady state of the physical asset, for example, GT rate of revolutions, and Gas Generator rate of revolutions. The targets are two dimensional vectors measuring GT Compressor decay state coefficients and GT Turbine decay state coefficients. Data samples were randomly divided into training and test sets (refer to Table 1).

G.1.5 vicon

The dataset vicon contains measurements recorded from a magnetic field which maps a 3-dimensional (3D) position to a 3D magnetic field strength [8][2]. The inputs are (x, y, z)-coordinates and the responses measured at there different heights are the target values. Data samples were randomly divided into training and test sets (refer to Table 1).

[1] http://archive.ics.uci.edu/ml/datasets/condition+based+maintenance+of+naval+propulsion+plants

[2] More information about data can be found in [8]. The data is available from https://github.com/carji475/linearly-constrained-gaussian-processes

G.1.6 hrtf

The dataset hrtf was obtained from the CIPIC HRTF database [1] which is a public-domain database of high-spatial-resolution head-related transfer function (HRTF) measurements[3]. We used the datasets of 37 subjects divided into training and test sets (refer to Table 1). Data for each subject includes 200-dimensional measurements of head-related impulse responses (HRIRs) and 8 input variables which are in fact the anthropometric parameters considered to have strong direct physical effect on HRIRs. The objective is to predict the HRIRs of the test subjects given their individualized anthropometric parameters[4].

G.1.7 nengo

The dataset nengo for this analysis was generated using The neural engineering object (Nengo) simulator [2; 14]. The generated time series data is constructed from a Nengo-based spiking model of action selection in the cortex-basal ganglia-thalamus circuit with timing predictions that are well matched to both single-cell recordings in rats and psychological paradigms in humans. Target measurements here are ensembles of leaky integrate-and-fire neurons comprised from seven nodes of the basal ganglia circuit (namely: globus pallidus internal, globus pallidus external, subthalamic nucleus, striatum D1, striatum D2; thalamus; motor cortex). Measurements from these 7 nodes are the target outputs. The advantage of using the Nengo neural simulator in the regression task is that we also have access to the ground-truth, the function generating the noisy target measurements at each node. Data samples[5] were randomly divided into training and test sets (refer to Table 1).

G.1.8 lorenz96

The synthetic dataset lorenz96 was generated using the Lorenz model [10, Eq. 3.2]. Using a locally defined notation, consider the Lorenz model of

$$\frac{\partial x_k}{\partial t} = -x_{k-1}(x_{k-2} - x_{k+1}) - x_k + F, \qquad \forall k \in K,$$

where x_k represent the state of the system and F is the forcing constant. In our simulation, we let $K = 20$ and set $F = 8$, which cause chaotic behavior. The initial state was set to equilibrium and a small perturbation was given to a randomly selected state. A small amount of noise was added to the resulting $d_y = 20$ dimensional feature vector. For the input ranging from 0 to 8, 1000 samples were collected on a linear space from the system. The objective is to identify the latent function generating data and perform predictions at 10^5locations in this interval, $[0, 8]$.

G.2 Datasets used in the illustrative experiment in Section 5, Figure 2.

G.2.1 ToyData

The synthetic dataset ToyData for the regression task in Figure 2-(a) is generated using the following nonlinear functions

$$f_1(x) = \exp\{\sin(\cos(x))\sin(\log(1 + |x^2 - 3x|))\},$$
$$f_2(x) = \log(|\tan(-2x)\cos(2x) + 1|)\sin(x).$$

We generated 32 noisy samples for input values in the range of $x \in [0, 12]$. The objective is to estimate the latent functions and perform predictions at 10^5 locations in this interval, $[0, 12]$.

G.2.2 vicon2

The dataset vicon2 is a subset of the vicon dataset (G.1.5) which includes 6000 samples from which 5000 randomly selected samples are used in the test set and 1000 samples in the training set. vicon2 is used in our numerical simulation presented in Figure 2-(b).

[3]Details of the database can be found at: https://www.ece.ucdavis.edu/cipic/spatial-sound/hrtf-data/.

[4]The preprocessed data can be obtained from our GitHub page: <GitHub link to data>.

[5]Data can be downloaded from our GitHib page: <GitHub link to data>.

G.3 Methods

G.3.1 SGPMC

MCMC for Variational Sparse Gaussian Processes (SGPMC) model of [7] using GPflow implementation[6] with RBF-ARD kernels, Gaussian likelihood, and 1000 pseudo inputs.

G.3.2 SVGP

The scalable variational Gaussian process (SVGP) model of [6] using a GPflow implementation with RBF-ARD kernels, Gaussian likelihood, and 1000 pseudo inputs.

G.3.3 SVIGP

Stochastic variational GP (SVIGP) model of [5] using a GPy[7] implementation with RBF-ARD kernels, Gaussian likelihood, and 1000 pseudo inputs.

[6] https://github.com/GPflow

[7] https://github.com/SheffieldML/GPy

References

[1] R. V. Algazi, R. O. Duda, D. M. Thompson, and C. Avendano. The CIPIC HRTF Database. In *WASSAP*, 2001.

[2] T. Bekolay, J. Bergstra, E. Hunsberger, T. DeWolf, T. Stewart, D. Rasmussen, X. Choo, A. Voelker, and C. Eliasmith. Nengo: a Python tool for building large-scale functional brain models. *Frontiers in Neuroinformatics*, 7(1), 2014.

[3] C. Bingham. An antipodally symmetric distribution on the sphere. *Annals of Statistics*, 2(6): 1201–1225, 1974.

[4] A. Coraddu, L. Oneto, A. Ghio, S. Savio, D. Anguita, and M. Figari. Machine learning approaches for improving condition-based maintenance of Naval propulsion plants. *Journal of Engineering for the Maritime Environment*, 230(1), 2014.

[5] J. Hensman, N. Fusi, and N. D. Lawrence. Gaussian processes for big data. In *Conference on Uncertainty in Artificial Intellegence (UAI)*, 2013.

[6] J. Hensman, A. G. de G. Matthews, and Z. Ghahramani. Scalable variational Gaussian process classification. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics AISTATS 2015, San Diego, California, USA, May 9-12, 2015*, 2015.

[7] J. Hensman, A. G. Matthews, M. Filippone, and Z. Ghahramani. MCMC for variationally sparse Gaussian processes. In *Advances in Neural Information Processing Systems (NIPS)*. 2015.

[8] C. Jidling, N. Wahlström, A. Wills, and T. B. Schön. Linearly constrained Gaussian processes. In *Advances in Neural Information Processing Systems (NIPS)*. 2017.

[9] A. Kume and A. T. A. Wood. Saddlepoint approximations for the Bingham and Fisher-Bingham normalising constants. *Biometrika*, 92(2):465–476, 2005.

[10] E. Lorenz. Predictability: a problem partly solved. In *Seminar on Predictability, 4-8 September 1995*, volume 1, pages 1–18, Shinfield Park, Reading, 1995.

[11] K. V. Mardia and P. E. Jupp. *Directional Statistics*. John Wiley & Sons, 2009.

[12] C. E. Rasmussen and C. K. I. Williams. *Gaussian processes for machine learning*. New York, NY, USA, 2006.

[13] E. Spyromitros-Xioufis, G. Tsoumakas, W. Groves, and I. Vlahavas. Multi-target regression via input space expansion: treating targets as inputs. *Machine Learning*, 104(1):55–98, 2016.

[14] J. Taghia, W. Cai, S. Ryali, J. Kochalka, J. Nicholas, T. Chen, and V. Menon. Uncovering hidden brain state dynamics that regulate performance and decision-making during cognition. *Nature Communications*, 9(2505), 2018.