Supplementary Material: Scalable Exact Inference in Multi-Output Gaussian Processes

Table of Contents

Notation

$\langle \cdot, \cdot \rangle$	Euclidean inner product
$\lVert \cdot \rVert$	Euclidean norm
$\lVert \cdot \rVert_{\text{op}}$	Operator norm
$\lVert \cdot \rVert_{\infty}$	Supremum norm
$\lVert \cdot \rVert_{F}$	Frobenius norm
$S^{\perp}$	Orthogonal complement of S
I_n	$n \times n$ identity matrix
$A > 0$	A is strictly-positive definite
$\lvert A \rvert$	Determinant of A
$A^{\dagger}$	Moore–Penrose pseudo-inverse of A
$\mathrm{col}(A)$	Column space of A
$A \otimes B$	Kronecker product of A and B
$\mathcal{N}(x \mid \mu, \Sigma)$	Density of the multivariate normal distribution with mean μ and covariance Σ at x

Assumptions

Throughout the appendix, we assume that the columns of H are linearly independent and that $\Sigma > 0$. As a consequence, $H^{\mathsf{T}}\Sigma^{-1}H > 0$.

A. How to Implement the OILMM

A.1. Parameters

The parameters of the OILMM are as follows:

Symbol	Type	Description
U	Truncated orthogonal $p \times m$ matrix	Orthogonal part of the basis $H = US^{\frac{1}{2}}$
S	Positive, diagonal $m \times m$ matrix	Diagonal part of the basis $H = US^{\frac{1}{2}}$
σ^2	Positive scalar	Part of the observation noise
D	Positive, diagonal $m \times m$ matrix	Part of the observation noise deriving from the latent processes
$(\theta_i)_{i=1}^m$	Hyperparameters	Hyperparameters for the latent processes, *e.g.* kernel parameters

A.2. Inference

Inference in the OILMM proceeds in three steps. Let $Y \in \mathbb{R}^{p \times n}$ be a matrix where the columns correspond to observations.

Projection step. In the projection step, we project the data to generate "observations for the latent processes". We denote these observations by $Y_{\text{proj}} \in \mathbb{R}^{m \times n}$, where again the columns corresponds to observations. We also construct the "projected noise", which is the observation noise under which the latent processes perform their observations.

(1) Construct the projection:

$$T = S^{-\frac{1}{2}} U^{\mathsf{T}} \in \mathbb{R}^{m \times p}.$$

(2) Project the observations:

$$Y_{\text{proj}} = TY \in \mathbb{R}^{m \times n}.$$

(3) Construct the projected noise:

$$\Sigma_T = \sigma^2 S^{-1} + D \in \mathbb{R}^{m \times m}_{\text{diag}}.$$

This is a diagonal matrix.

The i^{th} row of Y_{proj}, which we denote by $y^{(i)}_{\text{proj}} \in \mathbb{R}^n$, corresponds to observations for latent process i.

Projection step (missing data). In the case of missing data, certain elements of Y are missing. Partition the columns (time stamps) of Y into blocks $Y^{(1)} \in \mathbb{R}^{p \times n_1}, \ldots, Y^{(k)} \in \mathbb{R}^{p \times n_k}$ where $n_1 + \ldots + n_k = n$. These blocks should be chosen such that, for every block $Y^{(i)}$, the observations for an output are either all missing or all available, i.e. every row of $Y^{(i)}$ is either entirely missing or entirely available. Then consider the blocks $Y^{(1)} \in \mathbb{R}^{p \times n_1}, \ldots, Y^{(k)} \in \mathbb{R}^{p \times n_k}$ separately by repeatedly performing inference.

For every block—we henceforth suppress the dependence on the block index—denote by $Y_{\text{o}} \in \mathbb{R}^{p \times n}$ be the rows of the data matrix corresponding to observed outputs. Similarly, let $U_{\text{o}} \in \mathbb{R}^{p \times m}$ be the rows of U corresponding to observed outputs.

(1) Construct the projection:

$$T = S^{-\frac{1}{2}} (U_{\text{o}}^{\mathsf{T}} U_{\text{o}})^{-1} U_{\text{o}}^{\mathsf{T}} \in \mathbb{R}^{m \times p}.$$

(2) Project the observations:

$$Y_{\text{proj}} = TY_{\text{o}} \in \mathbb{R}^{m \times n}.$$

(3) Construct the projected noise:

$$\Sigma_T = \sigma^2 S^{-\frac{1}{2}} d[(U_{\text{o}}{}^{\mathsf{T}} U_{\text{o}})^{-1}] S^{-\frac{1}{2}} + D \in \mathbb{R}^{m \times m}_{\text{diag}}$$

where $d[A]$ sets the off-diagonal elements of A to zero. This is a diagonal matrix.

Latent process inference step. In this step, we perform inference on the latent processes.

(1) For $i = 1, \ldots, m$, do the following:

Conditioning: Condition latent process i on data $y_{\text{proj}}^{(i)} \in \mathbb{R}^n$ where the observation noise is $(\Sigma_T)_{ii}$. The latent process is just an independent GP, and any GP package can be used to do this step. Moreover, any single-output scaling technique can be used here, such as the variational inducing point approximation by Titsias (2009).

Prediction: Make predictions with the posterior of latent process i. Again, any GP package can be used to do this step. Denote the predictive means by $\mu^{(i)} \in \mathbb{R}^n$ and the predictive marginal variances by $\nu^{(i)} \in \mathbb{R}^n$.

(2) Collect the predictive means and marginal variances of the latent processes into matrices μ and ν:

$$\mu = \begin{bmatrix} (\mu^{(1)})^\mathsf{T} \\ \vdots \\ (\mu^{(m)})^\mathsf{T} \end{bmatrix} \in \mathbb{R}^{m \times n}, \quad \nu = \begin{bmatrix} (\nu^{(1)})^\mathsf{T} \\ \vdots \\ (\nu^{(m)})^\mathsf{T} \end{bmatrix} \in \mathbb{R}^{m \times n}.$$

Reconstruction step. In the reconstruction step, we construct the predictions of the OILMM from the predictions of the latent processes.

(1) Construct the basis: $H = US^{\frac{1}{2}} \in \mathbb{R}^{p \times m}$.

(2) Construct the predictive mean of the OILMM:

$$\text{predictive mean} = H\mu \in \mathbb{R}^{p \times n}.$$

(3) Construct the predictive marginal variances of the OILMM:

$$\text{predictive marginal variances} = (H \circ H)\nu \in \mathbb{R}^{p \times n}$$

where $\circ$ denotes the Hadamard product.

A.3. Posterior Sampling

Instead of computing posterior means and marginal variances, you might want to generate posterior samples.

Projection step. See App. A.2.

Latent process sampling step.

(1) For $i = 1, \ldots, m$, do the following:

Conditioning: Condition latent process i on data $y_{\text{proj}}^{(i)} \in \mathbb{R}^n$ where the observation noise is $(\Sigma_T)_{ii}$. The latent process is just an independent GP, and any GP package can be used to do this step. Moreover, any single-output scaling technique can be used here, such as the variational inducing point approximation by Titsias (2009).

Sampling: Sample from the posterior of latent process i. Again, any GP package can be used to do this step. Denote the sample by $\hat{x}^{(i)} \in \mathbb{R}^n$.

(2) Collect the samples into a matrix:

$$\hat{X} = \begin{bmatrix} (x^{(1)})^\mathsf{T} \\ \vdots \\ (x^{(m)})^\mathsf{T} \end{bmatrix} \in \mathbb{R}^{m \times n}.$$

Reconstruction step.

(1) Construct the basis: $H = US^{\frac{1}{2}} \in \mathbb{R}^{p \times m}$.

(2) Construct the posterior sample for the OILMM:

$$\text{posterior sample} = H\hat{X} \in \mathbb{R}^{p \times n}.$$

A.4. Computation of the Log-Marginal Likelihood

Projection step. See App. A.2.

Latent process marginal likelihood calculation.

(1) For $i = 1, \ldots, m$, do the following:

Marginal likelihood: Compute the log-probability of data $y_{\text{proj}}^{(i)} \in \mathbb{R}^n$ under latent process i where the observation noise is $(\Sigma_T)_{ii}$. Denote the resulting log-probability by LML_i. The latent process is just an independent GP, and any GP package can be used to do this step. Moreover, any single-output scaling technique can be used here, such as the variational inducing point approximation by Titsias (2009).

Reconstruction step.

(1) Construct the "regularisation term":

$$\text{regulariser} = -\frac{n}{2}\log|S| - \frac{n(p-m)}{2}\log 2\pi\sigma^2 - \frac{1}{2\sigma^2}(\|Y\|_F^2 - \|U^\mathsf{T} Y\|_F^2)$$

where $\|\bullet\|_F$ denotes the Frobenius norm.

(2) Construct the log-probability of the data Y under the OILMM:

$$\log p(Y) = \text{regulariser} + \sum_{i=1}^{m} \text{LML}_i.$$

Reconstruction step (missing data). In the case of missing data, (1) is slightly different:

(1) Construct the "regularisation term":

$$\text{regulariser} = -\frac{n}{2}\log|S| - \frac{n}{2}\log|U_\text{o}^\mathsf{T} U_\text{o}| - \frac{n(p-m)}{2}\log 2\pi\sigma^2 - \frac{1}{2\sigma^2}(\|Y_\text{o}\|_F^2 - \|\text{chol}(U_\text{o}^\mathsf{T} U_\text{o})^{-1} U_\text{o}^\mathsf{T} Y_\text{o}\|_F^2)$$

where $\|\bullet\|_F$ denotes the Frobenius norm and $\text{chol}(\bullet)$ the Cholesky decomposition. In this case, recall that n corresponds to the number of time points in the current block and to p to the number of observed outputs in the current block.

B. Unifying Presentation of Multi-Output Gaussian Processes

Our attempt at a unifying presentation of MOGP models starts from setting up what we call the *Mixing Model Hierarchy* (MMH). At the bottom of the Mixing Model Hierarchy stands the Instantaneous Linear Mixing Model (ILMM, Mod. 1 in Sec. 2.1), which is a simple, but general class of MOGP models typically characterised by low-rank covariance structure.

The graphical model of the ILMM is illustrated in the top-left corner of Fig. 7, which highlights two restrictions of the ILMM compared to a general MOGP: *(i)* the *instantaneous spatial covariance* of f, $\mathbb{E}[f(t)f^\mathsf{T}(t)] = HH^\mathsf{T}$, does not vary with time, because neither H nor $K(t,t) = I_m$ vary with time; and *(ii)* the noise-free observation $f(t)$ is a function of $x(t')$ for $t' = t$ only, meaning that, for example, f cannot be x with a delayed or a smoothed version of x. We hence call the ILMM a *time-invariant* (due to *(i)*) and *instantaneous* (due to *(ii)*) MOGP.

The ILMM can be generalised in three ways. First, the mixing matrix H may vary with time. Then $H \in \mathbb{R}^{p\times m}$ becomes a matrix-valued function $H\colon \mathcal{T} \to \mathbb{R}^{p\times m}$, and the mixing mechanism becomes

$$f(t) \,|\, H, x = H(t)x(t).$$

We call such MOGP models *time-varying* (see Fig. 7, top right). Second, $f(t)$ may depend on $x(t')$ for all $t' \in \mathcal{T}$. Then the mixing matrix $H \in \mathbb{R}^{p\times m}$ becomes a matrix-valued time-invariant filter $H\colon \mathcal{T} \to \mathbb{R}^{p\times m}$, and the mixing mechanism becomes

$$f(t) \,|\, H, x = \int H(t-\tau)x(\tau)\,\mathrm{d}\tau.$$

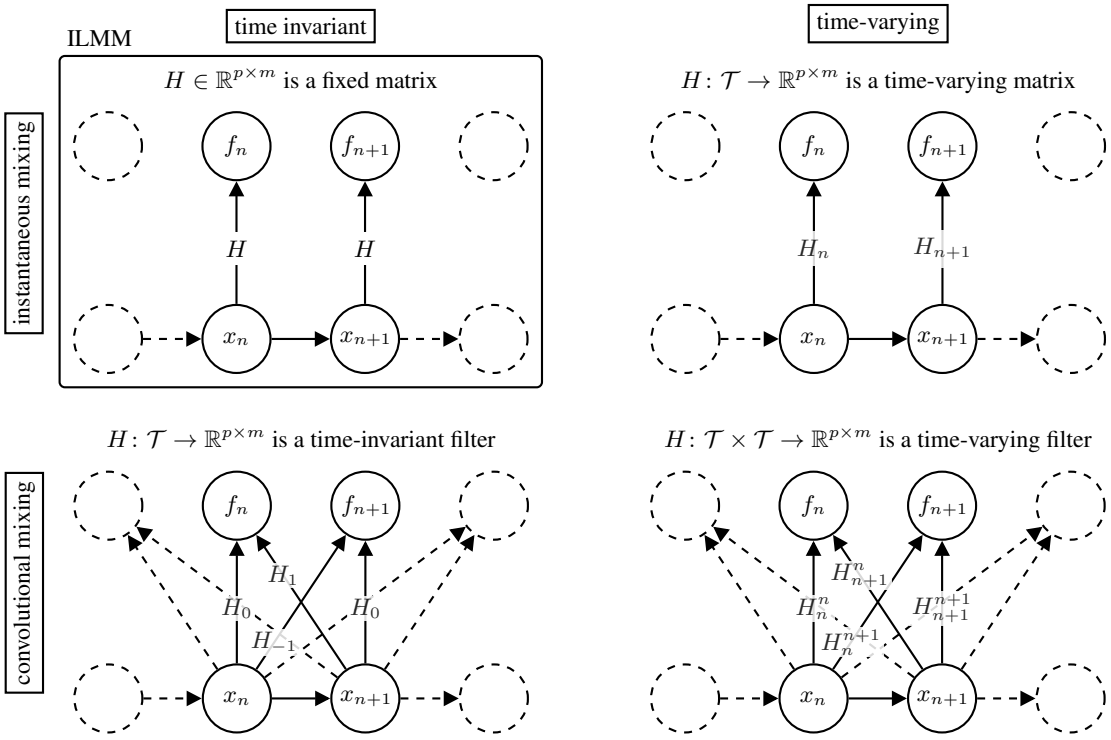
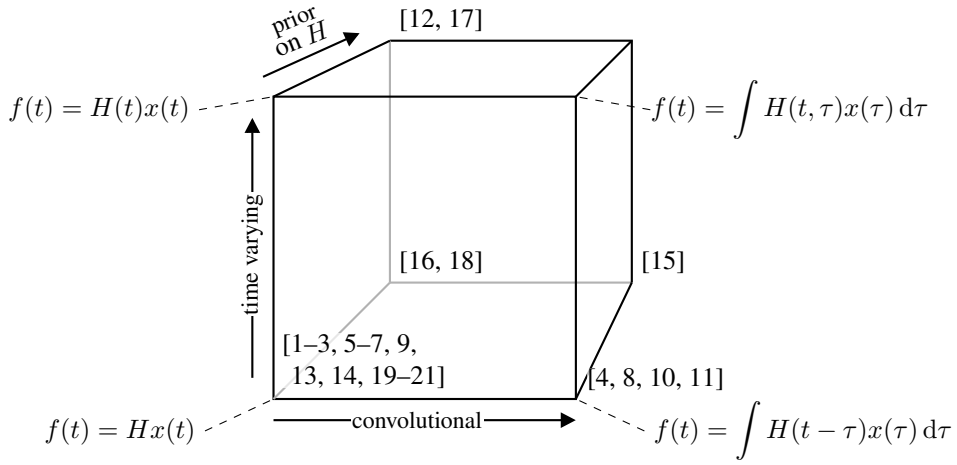


Figure 7. Graphical models illustrating the difference between time-invariant/time-varying and instantaneous/convolutional multi-output GP models, for data sampled at real-valued times $t_1, t_2, \ldots$ (sampling period Δt). Abbreviations used: $x_n = x(t_n)$, $f_n = f(t_n)$, $H_n = H(n\Delta t)$, and $H_n^m = H(t_m, t_n)$. For simplicity, the dynamics of x are depicted as a Markov chain; since x is modelled with a GP, x_n actually depends on $x_{n'}$ for all $n' \leq n$.



	Form of H	Form of K	Mixing
[1, 5, 6, 16]	H	$k(t,t')I$	Instantaneous
[2]	$\begin{bmatrix} H_1 & \cdots & H_q \end{bmatrix}$	$\mathrm{diag}(k_1(t,t')I, \ldots, k_q(t,t')I)$	Instantaneous
[3, 7, 9, 13, 20, 21]	H	$\mathrm{diag}(k_1(t,t'), \ldots, k_q(t,t'))$	Instantaneous
[4, 10, 11, 15]	$H(t-t')$	$\mathrm{diag}(\delta(t-t'), \ldots, \delta(t-t'))$	Convolutional
[8]	Green's function	$\mathrm{diag}(k_1(t,t'), \ldots, k_q(t,t'))$	Convolutional
[12, 17]	$H(t)$	$\mathrm{diag}(k_1(t,t'), \ldots, k_q(t,t'))$	Instantaneous
[14]	$\begin{bmatrix} H & I \end{bmatrix}$	$\mathrm{diag}(k_1(t,t'), \ldots, k_{q+p}(t,t'))$	Instantaneous
[18]	Lower triangular	$\mathrm{diag}(k_1(t,t'), \ldots, k_q(t,t'))$	Instantaneous
[19]	$H_1 \otimes \cdots \otimes H_q$	$k(t,t')I$	Instantaneous

[1] Intrinstic Coregionalisation Model (Goovaerts, 1997)
[2] Linear Model of Coregionalisation (Goovaerts, 1997)
[3] Semiparametric Latent Factor Model (Teh & Seeger, 2005)
[4] Dependent Gaussian Processes (Boyle & Frean, 2005)
[5] Multi-Task Gaussian Processes (Bonilla et al., 2008)
[6] Osborne et al. (2008)
[7] Higdon et al. (2008)
[8] Latent Force Models (Álvarez et al., 2009)
[9] Gaussian Process Factor Analysis (Yu et al., 2009)
[10] Multi-Output Gaussian Processes Through Variational Inducing Kernels (Álvarez et al., 2010)
[11] Convolved Multiple Output Gaussian Processes (Álvarez & Lawrence, 2011)
[12] Gaussian Process Regression Network (Wilson et al., 2012)
[13] Spatio–Temporal Bayesian Filtering and Smoothing (Särkkä et al., 2013)
[14] Collaborative Multi-Output Gaussian Processes (Nguyen & Bonilla, 2014)
[15] Generalised Gaussian Process Convolution Model (Bruinsma, 2016)
[16] Semi-Parametric Network Structure Discovery Models (Dezfouli et al., 2017)
[17] Grouped Gaussian Processes (Dahl & Bonilla, 2019)
[18] The Gaussian Process Autoregressive Regression Model (Requeima et al., 2019)
[19] High-Order Gaussian Process Regression (Zhe et al., 2019)
[20] Instantaneous Linear Mixing Model (Mod. 1)
[21] Orthogonal Instantaneous Linear Mixing Model (Mod. 2)

Figure 8. The Mixing Model Hierarchy, which organises MOGPs from the machine learning and geostatistics literature according to their distinctive modelling assumptions

Table 3. Complexities of learning and inference in the ILMM and OILMM, ignoring the projection. In the table, n is the number of time points; p is the number of outputs; m is the number of latent processes; r is the number of inducing points, typically $r \ll n$; and d is the state dimensionality, typically $d \ll n, m$.

Model	Runtime	Memory
General MOGP	$O(n^3p^3)$	$O(n^2p^2)$
ILMM (Mod. 1)	$O(n^3m^3)$	$O(n^2m^2)$
OILMM (Mod. 2)	$O(n^3m)$	$O(n^2m)$
OILMM (Mod. 2) + Titsias (2009)	$O(nmr^2)$	$O(nmr)$
OILMM (Mod. 2) + Hartikainen & Särkkä (2010)	$O(nmd^3)$	$O(nmd^2)$
APPLICATION TO SEPARABLE SPATIO–TEMPORAL GPS (SEC. 3.9)		
OILMM (Mod. 2)	$O(n^3p)$	$O(n^2p)$
OILMM (Mod. 2) + Titsias (2009)	$O(npr^2)$	$O(npr)$
OILMM (Mod. 2) + Hartikainen & Särkkä (2010)	$O(npd^3)$	$O(npd^2)$
Kronecker product factorisation (Saatçi, 2012, Ch. 5)	$O(n^3+p^3)$	$O(n^2+p^2)$

Table 4. Complexities of projecting the data and reconstructing the predictions in the ILMM and OILMM. In the table, n is the number of time points; p is the number of outputs; and m is the number of latent processes.

Action	Runtime	Memory
Storing data	–	$O(np)$
Construction of projection T	$O(m^2p)$	$O(mp)$
Projection	$O(nmp)$	$O(np)$
Construction of predictive marginal statistics	$O(nmp)$	$O(np)$
APPLICATION TO SEPARABLE SPATIO–TEMPORAL GPS (SEC. 3.9)		
Construction of projection T	$O(p^3)$	$O(p^2)$
Projection	$O(np^2)$	$O(np)$
Construction of predictive marginal statistics	$O(np^2)$	$O(np)$

We call such MOGP models *convolutional* (see Fig. 7, bottom left). Finally, $f(t)$ may depend on $x(t')$ for all $t' \in \mathcal{T}$ *and* this relationship may vary with time. Then the mixing matrix $H \in \mathbb{R}^{p\times m}$ becomes a matrix-valued time-varying filter $H\colon \mathcal{T}\times\mathcal{T}\to\mathbb{R}^{p\times m}$, and the mixing mechanism becomes

$$f(t) \mid H, x = \int H(t,\tau)x(\tau)\,\mathrm{d}\tau.$$

We call such MOGP models *time-varying* and *convolutional* (see Fig. 7, bottom right). The graphical models corresponding to these generalisations of the ILMM are depicted in Fig. 7.

The ILMM can be extended in one other way, which is to include a prior distribution on H. This extension and the two previously proposed generalisations together form the *Mixing Model Hierarchy* (MMH), which is depicted in Fig. 8. The MMH organises multi-output Gaussian process models according to their distinctive modelling assumptions. Fig. 8 shows how sixteen MOGP models from the machine learning and geostatistics literature can be recovered as special cases of the various generalisations of the ILMM.

Not all multi-output Gaussian process models are covered by the MMH, however. For example, Deep GPs (Damianou, 2015) and variations thereon (Kaiser et al., 2018) are excluded because they transform the latent processes *nonlinearly* to generate the observations.

C. Runtime and Memory Complexities

For the ILMM and OILMM, Tab. 3 gives an overview of the runtime and memory complexities associated to learning and inference, and Tab. 4 gives an overview of the runtime and memory complexities associated to projecting the data and reconstructing the predictions.

D. Maximum Likelihood Estimate

Prop. 2. Denote $p(y \mid x) = \mathcal{N}(y \mid Hx, \Sigma)$, and let T be the $m \times p$ matrix $(H^\mathsf{T}\Sigma^{-1}H)^{-1}H^\mathsf{T}\Sigma^{-1}$. Then

$$Ty = \arg\max_x p(y \mid x)$$

and Ty is an unbiased estimate of x: $\mathbb{E}[Ty \mid x] = x$.

Proof. Note that

$$\log p(y \mid x) \simeq -\tfrac{1}{2}(y - Hx)^\mathsf{T}\Sigma^{-1}(y - Hx)$$

Using invertibility of $H^\mathsf{T}\Sigma^{-1}H$, an elementary calculation then shows that the unique maximum with respect to x is given by

$$x = (H^\mathsf{T}\Sigma^{-1}H)^{-1}H^\mathsf{T}\Sigma^{-1}y = Ty.$$

To show that Ty is an unbiased estimate of x, we use that $\mathbb{E}[y \mid x] = Hx$:

$$\mathbb{E}[Ty \mid x] = THx = (H^\mathsf{T}\Sigma^{-1}H)^{-1}(H^\mathsf{T}\Sigma^{-1}H)x = x. \qquad \square$$

E. Sufficient Statistic

To prove sufficiency of Ty , we need the property of T that it "preserves the signal-to-noise ratio". This is characterised in the following lemma.

Lem. 1.

$$\frac{\mathcal{N}(y \mid Hx, \Sigma)}{\mathcal{N}(y \mid 0, \Sigma)} = \frac{\mathcal{N}(Ty \mid x, (H^\mathsf{T}\Sigma^{-1}H)^{-1})}{\mathcal{N}(Ty \mid 0, (H^\mathsf{T}\Sigma^{-1}H)^{-1})}.$$

Proof. It is simple to check the equality by direct verification. We show, however, how the equality may be derived. To begin with, we have

$$(y - Hx)^\mathsf{T}\Sigma^{-1}(y - Hx) = y^\mathsf{T}\Sigma^{-1}y - 2x^\mathsf{T}H^\mathsf{T}\Sigma^{-1}y + x^\mathsf{T}H^\mathsf{T}\Sigma^{-1}Hx.$$

Here

$$H^\mathsf{T}\Sigma^{-1}y = (H^\mathsf{T}\Sigma^{-1}H)(H^\mathsf{T}\Sigma^{-1}H)^{-1}H^\mathsf{T}\Sigma^{-1}y = (H^\mathsf{T}\Sigma^{-1}H)Ty,$$

so

$$(y - Hx)^\mathsf{T}\Sigma^{-1}(y - Hx) = y^\mathsf{T}\Sigma^{-1}y - 2x^\mathsf{T}(H^\mathsf{T}\Sigma^{-1}H)Ty + x^\mathsf{T}(H^\mathsf{T}\Sigma^{-1}H)x.$$

Adding and subtracting $yT^\mathsf{T}(H^\mathsf{T}\Sigma^{-1}H)Ty$, we find

$$(y - Hx)^\mathsf{T}\Sigma^{-1}(y - Hx) = y^\mathsf{T}\Sigma^{-1}y - yT^\mathsf{T}(H^\mathsf{T}\Sigma^{-1}H)Ty + (x - Ty)^\mathsf{T}(H^\mathsf{T}\Sigma^{-1}H)(x - Ty).$$

Hence, rearranging,

$$(y - Hx)^\mathsf{T}\Sigma^{-1}(y - Hx) - y^\mathsf{T}\Sigma^{-1}y = (x - Ty)^\mathsf{T}(H^\mathsf{T}\Sigma^{-1}H)(x - Ty) - yT^\mathsf{T}(H^\mathsf{T}\Sigma^{-1}H)Ty,$$

which yields the result. $\square$

Prop. 3. The MLE Ty of x is a minimal sufficient statistic for x.

Proof of Prop. 3. By a general characterisation of minimal sufficient statistics (see, *e.g.*, Th. 6.2.13 in Casella & Berger, 2001), Ty is a minimal sufficient statistic for x if and only if it is true that $p(y_1 \mid x)/p(y_2 \mid x)$ is constant as a function of x if and only if $Ty_1 = Ty_2$. Indeed, by Lem. 1,

$$\log \frac{p(y_1 \mid x)}{p(y_2 \mid x)} = (Ty_1 - Ty_2)^\mathsf{T}(H^\mathsf{T}\Sigma^{-1}H)^{-1}x + \text{const.}$$

which, by invertibility of $H^\mathsf{T}\Sigma^{-1}H$, does not depend on x if and only if $Ty_1 = Ty_2$. $\square$

F. Proof of Prop. 1

Proof of Prop. 1. By Prop. 3,

$$p(f \mid Y) = \int p(f \mid x) p(x \mid Y) \, \mathrm{d}x = \int p(f \mid x) p(x \mid TY) \, \mathrm{d}x = p(f \mid TY)$$

where TY are observations for the process Ty. Since

$$y \mid x \sim \mathcal{GP}(Hx, \delta[t - t']\Sigma),$$

the process Ty has distribution

$$Ty \mid x \sim \mathcal{GP}(THx, \delta[t - t']T\Sigma T^\mathsf{T}).$$

By explicit calculation, we find that

$$TH = (H^\mathsf{T}\Sigma^{-1}H)^{-1}H^\mathsf{T}\Sigma^{-1}H = I$$

and

$$T\Sigma T^\mathsf{T} = (H^\mathsf{T}\Sigma^{-1}H)^{-1}H^\mathsf{T}\Sigma^{-1}\Sigma\Sigma^{-1}H(H^\mathsf{T}\Sigma^{-1}H)^{-1} = (H^\mathsf{T}\Sigma^{-1}H)^{-1}.$$

Thus

$$Ty \mid x \sim \mathcal{GP}(x, \delta[t - t']\Sigma_T) \quad \text{where} \quad \Sigma_T = (H^\mathsf{T}\Sigma^{-1}H)^{-1}.$$

Moreover, using Lem. 1, the probability of the data Y is given by

$$p(Y) = \int \prod_{i=1}^{n} \mathcal{N}(y_i \mid Hx, \Sigma) p(x) \, \mathrm{d}x = \left[\frac{\mathcal{N}(y_i \mid 0, \Sigma)}{\mathcal{N}(y_i \mid 0, \Sigma_T)}\right] \int \prod_{i=1}^{n} \mathcal{N}(Ty_i \mid x, \Sigma_T) p(x) \, \mathrm{d}x. \qquad \square$$

G. Interpretation of the Likelihood

Prop. 4. The regularisation terms in like likelihood in Prop. 1 can be written as

$$\log \frac{\mathcal{N}(y \mid 0, \Sigma)}{\mathcal{N}(Ty \mid 0, \Sigma_T)} = -\frac{1}{2}(p - m) \log 2\pi - \overbrace{\frac{1}{2} \log \frac{|\Sigma|}{|\Sigma_T|}}^{\text{noise “lost by projection”}} - \underbrace{\frac{1}{2}\|(I_p - HT)y\|_\Sigma^2}_{\text{data “lost by projection”}},$$

where $\|\bullet\|_\Sigma$ denotes the norm induced by the weighted inner product $\langle \bullet, \bullet \rangle_\Sigma = \langle \Sigma^{-1} \bullet, \bullet \rangle$.

Proof. The first two terms come directly from the multivariate Gaussian densities. We show how the third term may be obtained. Rearrange

$$\langle y, T^\mathsf{T}\Sigma_T^{-1}Ty \rangle = \langle \Sigma^{-\frac{1}{2}}y, (\Sigma^{\frac{1}{2}}T^\mathsf{T}\Sigma_T^{-1}T\Sigma^{\frac{1}{2}})\Sigma^{-\frac{1}{2}}y \rangle = \langle \Sigma^{-\frac{1}{2}}y, P\Sigma^{-\frac{1}{2}}y \rangle$$

where

$$P = \Sigma^{\frac{1}{2}}(T^\mathsf{T}\Sigma_T^{-1}T)\Sigma^{\frac{1}{2}} = \Sigma^{-\frac{1}{2}}H(H^\mathsf{T}\Sigma^{-1}H)^{-1}H^\mathsf{T}\Sigma^{-\frac{1}{2}} = \Sigma^{-\frac{1}{2}}HT\Sigma^{\frac{1}{2}}$$

which is the orthogonal projection onto $\mathrm{col}(\Sigma^{-\frac{1}{2}}H)$. Recall that an orthogonal projection P is defined by $P^2 = P$ and $P^\mathsf{T} = P$. Then

$$\begin{aligned}
\langle y, \Sigma^{-1}y \rangle - \langle y, T^\mathsf{T}\Sigma_T^{-1}Ty \rangle &= \langle \Sigma^{-\frac{1}{2}}y, (I_p - P)\Sigma^{-\frac{1}{2}}y \rangle \\
&= \langle \Sigma^{-\frac{1}{2}}y, (I_p - P)^2\Sigma^{-\frac{1}{2}}y \rangle \\
&= \langle (I_p - P)^\mathsf{T}\Sigma^{-\frac{1}{2}}y, (I_p - P)\Sigma^{-\frac{1}{2}}y \rangle \\
&= \|(I_p - P)\Sigma^{-\frac{1}{2}}y\|^2,
\end{aligned}$$

where we note that $(I_p - P)^2 = I_p - P$ and that $I_p - P$ is symmetric. (In fact, $P^\perp = I_p - P$ is the orthogonal projection onto $\mathrm{col}(\Sigma^{-\frac{1}{2}}H)^\perp$.) To conclude, see that

$$\|(I_p - P)\Sigma^{-\frac{1}{2}}y\|^2 = \|\Sigma^{-\frac{1}{2}}(I_p - \Sigma^{\frac{1}{2}}P\Sigma^{-\frac{1}{2}})y\|^2 = \|(I_p - HT)y\|_\Sigma^2. \qquad \square$$

We note that HT is a projection, but not necessarily an *orthogonal* projection.

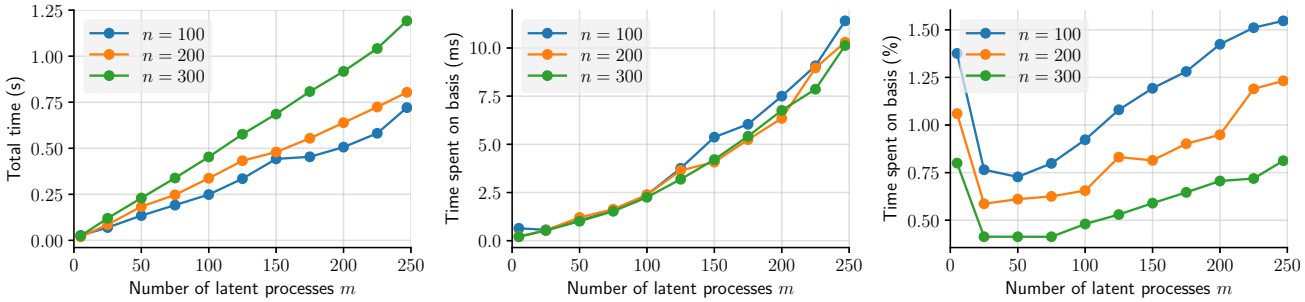


Figure 9. Comparison of the time it takes to construct the basis H to the total time of a log-marginal likelihood computation for a range of numbers of data points n and numbers of latent processes m. The data used is from the temperature extrapolation experiment (Sec. 4.3).

H. Tensor Product Basis

If the observations can be naturally represented as multi-index arrays in $\mathbb{R}^{p_1 \times \cdots \times p_q}$, where the total number of outputs is $p = \prod_{i=1}^{q} p_i$, to obtain a reduction in parameters of H, a natural choice is to correspondingly decompose $H = H_1 \otimes \cdots \otimes H_q$ where $\otimes$ is the Kronecker product and H_i a $p_i \times m_i$ matrix. The latent processes are then naturally seen as a $\mathbb{R}^{m_1 \times \cdots \times m_q}$-valued process, where their total number is $m = \prod_{i=1}^{q} m_i$. In this parametrisation of the ILMM, Prop. 5 shows that the projection and projected noise also become the Kronecker products: $T = T_1 \otimes \cdots \otimes T_q$ and $\Sigma_T = \Sigma_{T_1} \otimes \cdots \otimes \Sigma_{T_q}$. Using the vectorisation trick, TY can be computed efficiently without the need to explicitly construct T.

Prop. 5. Let H be a basis that is a tensor product of other bases and let the observation noise Σ factorise similarly:

$$H = H_1 \otimes \cdots \otimes H_q \quad \text{and} \quad \Sigma = \Sigma_1 \otimes \cdots \otimes \Sigma_q.$$

Then the projection is the tensor product of the projections and the projected noise is the tensor product of the projected noises:

$$T = T_1 \otimes \cdots \otimes T_q \quad \text{and} \quad \Sigma_T = \Sigma_{T_1} \otimes \cdots \otimes \Sigma_{T_q}$$

where $T_i = (H_i^\mathsf{T} \Sigma_i^{-1} H_i)^{-1} H_i^\mathsf{T} \Sigma_i^{-1}$ and $\Sigma_i = (H_i^\mathsf{T} \Sigma_i^{-1} H_i)^{-1}$.

Proof. Follows directly from the compatibility of the Kronecker product with matrix multiplication, transposition, and inversion. □

I. Cost of Parametrising the Basis

For the OILMM, the only computation that does not scale linearly with the number of latent processes m is the parametrisation of the orthogonal part U of the basis H, which takes $O(m^2 p)$ time. We argue that this cost is dominated by the cost $O(n^3 m + nmp)$ of computing the log-marginal likelihood of the projected data:

(i) typically $m \leq n, p$;

(ii) the cost of computing the log-marginal likelihood of the projected data scales with n, and often $n \gg m, p$; and

(iii) assuming that p is not much bigger than n, computing the log-marginal likelihood of the projected data costs at least $O(n)$ more, so the cost of parametrising the basis H should become insignificant as n grows.

We compare the time it takes to construct the basis H to the total time of a log-marginal likelihood computation for a range of numbers of data points n and numbers of latent processes m. We use the data from the temperature extrapolation experiment (Sec. 4.3). The results are depicted in Fig. 9. Observe that, even in the worst case when $m = p = 247$, parametrising the basis H takes no more than 1.5% of the total time at $n = 100$ data points and no more than 0.8% of the total time at $n = 300$ data points. This cost is negligible, even in this worst case.

J. Characterisation of Diagonal Projected Noise

Prop. 6 says that the projected noise is diagonal if and only if H is of the form $H = \Sigma^{\frac{1}{2}} U S^{\frac{1}{2}}$ with U a matrix with orthonormal columns and $S > 0$ diagonal. This condition is awkward, as it couples H and Σ. Fortunately, Prop. 6 also

shows that we may drop H's dependency on Σ if and only if every column of U is an eigenvector of Σ.

Prop. 6. The projected noise Σ_T is diagonal if and only if H is of the form $H = \Sigma^{\frac{1}{2}} U S^{\frac{1}{2}}$ with U a matrix with orthonormal columns and $S > 0$ diagonal. Suppose that this is the case, and fix such a U. Then H is of the form $H = UD^{\frac{1}{2}}$ with $D > 0$ diagonal if and only if every column of U is an eigenvector of Σ.

Proof. The projected noise is diagonal if and only if $H^\mathsf{T}\Sigma^{-1}H = S$ for some $S > 0$ diagonal. This condition is equivalent to

$$S^{-\frac{1}{2}} H^\mathsf{T} \Sigma^{-\frac{1}{2}} \Sigma^{-\frac{1}{2}} H S^{-\frac{1}{2}} = I_m,$$

which, in turn, holds if and only if $\Sigma^{-\frac{1}{2}} H S^{-\frac{1}{2}} = U$ is a matrix with orthonormal columns. Thus, the projected noise is diagonal if and only if H is of the form $H = \Sigma^{\frac{1}{2}} U S^{\frac{1}{2}}$ with U a matrix with orthonormal columns and $S > 0$ diagonal.

For the second statement, note that every column of U is an eigenvector of Σ if and only if it is an eigenvector of $\Sigma^{\frac{1}{2}}$. Suppose that H is of the form $H = UD^{\frac{1}{2}}$ with $D > 0$ diagonal. Then

$$\Sigma^{\frac{1}{2}} U = \Sigma^{\frac{1}{2}} U S^{\frac{1}{2}} S^{-\frac{1}{2}} = H S^{-\frac{1}{2}} = U D^{\frac{1}{2}} S^{-\frac{1}{2}},$$

so every column of U is an eigenvector of $\Sigma^{\frac{1}{2}}$. Conversely, suppose that every column of U is an eigenvector of $\Sigma^{\frac{1}{2}}$ with eigenvalues stacked into a diagonal matrix $D > 0$. Then

$$H = \Sigma^{\frac{1}{2}} U S^{\frac{1}{2}} = U D S^{\frac{1}{2}},$$

which is of the desired form. □

K. Kullback–Leibler Divergence Between an ILMM and OILMM

Prop. 7. Consider two ILMMs with equal $\Sigma = \sigma^2 I_p$, equal $K(t, t')$, but different bases H and $\hat{H}$. Let $t_1, \ldots, t_n \in \mathcal{T}$ and denote $x_i = x(t_i)$ and $y_i = y(t_i)$. It then holds that

$$\mathrm{D_{KL}}(p(y_{1:n}, x_{1:n}) \,\|\, \hat{p}(y_{1:n}, x_{1:n})) = \mathrm{D_{KL}}(\hat{p}(y_{1:n}, x_{1:n}) \,\|\, p(y_{1:n}, x_{1:n})) = n \frac{1}{2\sigma^2} \|H - \hat{H}\|_F^2$$

and

$$\inf_{\hat{H} \,:\, \text{OILMM}} \mathrm{D_{KL}}(p(y_{1:n}, x_{1:n}) \,\|\, \hat{p}(y_{1:n}, x_{1:n})) \leq n \frac{\mathbb{E}[\|f(t)\|^2]}{\sigma^2} \max_i (1 - V_{ii}) \leq n \frac{\mathbb{E}[\|f(t)\|^2]}{2\sigma^2} \|I_m - V\|_F^2$$

where $\hat{H}$ ranges over matrices of the form $US^{\frac{1}{2}}$ with U a matrix with orthonormal columns and $S^{\frac{1}{2}} > 0$ diagonal, V is the orthogonal matrix collecting the right singular vectors of H, and $\mathbb{E}[\|f(t)\|^2]$ denotes the variance of the observations under the first ILMM before adding noise.

Proof. Start out by expanding the Kullback–Leibler divergence and noting that $p(x_{1:n}) = \hat{p}(x_{1:n})$:

$$\begin{aligned}
\mathrm{D_{KL}}(p(y_{1:n}, x_{1:n}) \,\|\, \hat{p}(y_{1:n}, x_{1:n})) &= -\mathbb{E}_{p(y_{1:n}, x_{1:n})} \log \frac{\hat{p}(y_{1:n} \,|\, x_{1:n}) \cancel{\hat{p}(x_{1:n})}}{p(y_{1:n} \,|\, x_{1:n}) \cancel{p(x_{1:n})}} \\
&= -\sum_{i=1}^{n} \mathbb{E}_{p(y_i, x_i)} [\log \hat{p}(y_i \,|\, x_i) - \log p(y_i \,|\, x_i)] \\
&= -\sum_{i=1}^{n} \mathbb{E}_{p(y_i, x_i)} [\log \mathcal{N}(y_i \,|\, \hat{H} x_i, \sigma^2 I_p) - \log \mathcal{N}(y_i \,|\, H x_i, \sigma^2 I_p)].
\end{aligned}$$

Here

$$\mathbb{E}_{p(y_i, x_i)} [\log \mathcal{N}(y_i \,|\, \hat{H} x_i, \sigma^2 I_p)] = -\frac{p}{2} \log 2\pi\sigma^2 - \frac{1}{2\sigma^2} \mathbb{E}_{p(y_i, x_i)} [\|y_i - \hat{H} x_i\|^2]$$

where

$$\begin{aligned}
\mathbb{E}_{p(y_i, x_i)} [\|y_i - \hat{H} x_i\|^2] &= \mathbb{E}_{p(y_i, x_i)} \operatorname{tr}[y_i y_i^\mathsf{T} - 2 y_i x_i^\mathsf{T} \hat{H}^\mathsf{T} + x_i x_i^\mathsf{T} \hat{H} \hat{H}^\mathsf{T}] \\
&= \operatorname{tr}[H H^\mathsf{T} + \sigma^2 I - 2 H \hat{H}^\mathsf{T} + \hat{H} \hat{H}^\mathsf{T}] \\
&= p\sigma^2 + \operatorname{tr}[(H - \hat{H})(H - \hat{H})^\mathsf{T}] \\
&= p\sigma^2 + \|H - \hat{H}\|_F^2.
\end{aligned}$$

Therefore,

$$\mathrm{D_{KL}}(p(y_{1:n}, x_{1:n}) \,\|\, \hat{p}(y_{1:n}, x_{1:n})) = n\frac{1}{2\sigma^2}\|H - \hat{H}\|_F^2.$$

Let $H = USV^\mathsf{T}$ be the SVD of H where U is a truncated orthogonal matrix with the same shape as H, $S > 0$ is a square diagonal matrix, and V is a square orthogonal matrix. Note that $U^\mathsf{T}U = I_m$, but $UU^\mathsf{T} \neq I_p$. Then, choosing $\hat{H} = US$,

$$\inf_{\hat{H}\,:\,\text{OILMM}} \mathrm{D_{KL}}(p(y_{1:n}, x_{1:n}) \,\|\, \hat{p}(y_{1:n}, x_{1:n})) \leq n\frac{1}{2\sigma^2}\|U(SV^\mathsf{T} - S)\|_F^2 = n\frac{1}{2\sigma^2}\|SV^\mathsf{T} - S\|_F^2$$

since $\|UA\|_F^2 = \operatorname{tr}[A^\mathsf{T}U^\mathsf{T}UA] = \operatorname{tr}[A^\mathsf{T}A] = \|A\|_F^2$. We now further simplify:

$$\|SV^\mathsf{T} - S\|_F^2 = \operatorname{tr}[(SV^\mathsf{T} - S)(SV^\mathsf{T} - S)^\mathsf{T}] = \operatorname{tr}[SV^\mathsf{T}VS - SV^\mathsf{T}S - SVS + SS] = 2\operatorname{tr}[SS - SVS].$$

Hence, by definition of the trace and the fact that S is diagonal,

$$\|SV^\mathsf{T} - S\|_F^2 = 2\sum_{i=1}^m S_{ii}^2(1 - V_{ii}) \leq 2\left(\sum_{i=1}^m S_{ii}^2\right)\max_i(1 - V_{ii}) = 2\mathbb{E}[\|f\|^2]\max_i(1 - V_{ii}),$$

since

$$\mathbb{E}[\|f(t)\|^2] = \mathbb{E}\operatorname{tr}[f(t)f^\mathsf{T}(t)] = \operatorname{tr}[HH^\mathsf{T}] = \operatorname{tr}[S^2].$$

Therefore,

$$\|SV^\mathsf{T} - S\|_F^2 \leq 2\mathbb{E}[\|f\|^2]\max_i(1 - V_{ii}) \leq 2\mathbb{E}[\|f\|^2]\sum_{i=1}^m(1 - V_{ii}) = \mathbb{E}[\|f\|^2]\|I_m - V\|_F^2,$$

where the equality follows from a similar calculation:

$$\|I_m - V\|_F^2 = \operatorname{tr}[I_m - V^\mathsf{T} - V + V^\mathsf{T}V] = 2\operatorname{tr}[I_m - V].$$ □

L. OILMM: Projection and Projected Noise

Prop. 8. Consider the OILMM (Mod. 2). Then the projection and projected noise are given by

$$T = S^{-\frac{1}{2}}U^\mathsf{T} \quad \text{and} \quad \Sigma_T = \sigma^2 S^{-1} + D.$$

Proof. To begin with, note that

$$\begin{aligned} y &\sim \mathcal{GP}(\underbrace{HK(t,t')H^\mathsf{T} + \delta[t - t'](\sigma^2 I_p + HDH^\mathsf{T})}), \\ &= H(K(t,t') + \delta[t - t']D)H^\mathsf{T} + \delta[t - t']\sigma^2 I_p, \end{aligned}$$

so we can assume that $D = 0$ by "absorbing it into $K(t,t')$". We then find that

$$H^\mathsf{T}\Sigma^{-1}H = \sigma^{-2}S,$$

so

$$\Sigma_T = T\Sigma T^\mathsf{T} = (H^\mathsf{T}\Sigma^{-1}H)^{-1} = \sigma^2 S^{-1}.$$

Moreover, then

$$T = (H^\mathsf{T}\Sigma^{-1}H)^{-1}H^\mathsf{T}\Sigma^{-1} = (\sigma^2 S^{-1})(\sigma^{-2}S^{\frac{1}{2}}U^\mathsf{T}) = S^{-\frac{1}{2}}U^\mathsf{T}.$$

Finally, "pull D back out of $K(t,t')$", which we note is equivalent to adding it to Σ_T by Prop. 1. □

M. OILMM: Likelihood

Prop. 9. Consider the OILMM (Mod. 2). Let Y be an $p \times n$ matrix of observations for y. Then

$$\log p(Y) = -\frac{n}{2}\log|S| - \frac{n(p-m)}{2}\log 2\pi\sigma^2 - \frac{1}{2\sigma^2}\|(I_p - UU^\mathsf{T})Y\|_F^2 + \sum_{i=1}^{m}\log\mathcal{N}((TY)_{i:} \mid 0, K_i + (\sigma^2/S_{ii} + D_{ii})I_n)$$

where $\|\cdot\|_F$ denotes the Frobenius norm and K_i is the $n \times n$ kernel matrix for the i^{th} latent process x_i.

Proof. By Prop. 1 and Prop. 4, we have

$$\log p(Y) = -\frac{n(p-m)}{2}\log 2\pi - \frac{n}{2}\log\frac{|\Sigma|}{|\Sigma_T|} - \frac{1}{2}\sum_{i=1}^{n}\|(I_p - HT)y_i\|_\Sigma^2 + \log\int p(x)\prod_{i=1}^{n}\mathcal{N}(Ty_i \mid x_i, \Sigma_T)\,\mathrm{d}x.$$

Using the same trick as in the proof of Prop. 8, assume that $D = 0$ by "absorbing it into $K(t,t')$". We then simplify the terms one by one. First, we have that

$$\log\frac{|\Sigma|}{|\Sigma_T|} = \log\frac{|\sigma^2 I_p|}{|\sigma^2 S^{-1}|} = (p-m)\log\sigma^2 + \log|S|.$$

Second, note that $I_p - HT = I_p - UU^\mathsf{T}$, which we denote by $P_{U^\perp}$ and which is symmetric, so

$$\|(I_p - HT)y_i\|_\Sigma^2 = \|P_{U^\perp}y_i\|_\Sigma^2 = \langle P_{U^\perp}y_i, \Sigma^{-1}P_{U^\perp}y_i\rangle = \sigma^{-2}\langle P_{U^\perp}y_i, P_{U^\perp}y_i\rangle = \sigma^{-2}\operatorname{tr}[P_{U^\perp}P_{U^\perp}y_i y_i^\mathsf{T}].$$

Then sum over $i = 1, \ldots, n$ to obtain

$$\sum_{i=1}^{n}\|(I_p - HT)y_i\|_\Sigma^2 = \sigma^{-2}\operatorname{tr}[P_{U^\perp}P_{U^\perp}YY^\mathsf{T}] = \sigma^{-2}\|P_{U^\perp}Y\|_F^2.$$

Finally,

$$\log\int p(x)\prod_{i=1}^{n}\mathcal{N}(Ty_i \mid x_i, \Sigma_T)\,\mathrm{d}x = \sum_{i=1}^{m}\log\mathcal{N}((TY)_{i:} \mid 0, K_i + (\sigma^2/S_{ii} + D_{ii})I_n)$$

follows from independence of the latent processes and remembering that we "absorbed D into $K(t,t')$". □

Observe that

$$\|(I_p - UU^\mathsf{T})Y\|_F^2 = \|Y\|_F^2 - \|U^\mathsf{T}Y\|_F^2,$$

which is a computationally more efficient implementation.

N. OILMM: Decomposition of the Mean Squared Error

Prop. 10. Let $H = US^{\frac{1}{2}}$ with U a matrix with orthonormal columns and $S^{\frac{1}{2}} > 0$ diagonal. Then

$$\underbrace{\|y - Hx\|^2}_{\text{MSE}} = \underbrace{\|P_{U^\perp}y\|^2}_{\substack{\text{data not}\\ \text{captured by basis}}} + \sum_{i=1}^{m}\overbrace{S_{ii}}^{\substack{\text{variance of}\\ i^{\text{th}}\text{ latent process}}}\underbrace{((Ty)_i - x_i)^2}_{\substack{\text{MSE of}\\ i^{\text{th}}\text{ latent process}}}$$

where $T = S^{-\frac{1}{2}}U^\mathsf{T}$ and $P_{U^\perp}$ is the orthogonal projection onto the orthogonal complement of $\operatorname{col}(U)$.

Proof. By expanding and using orthogonality of U,

$$\begin{aligned}
\|y - Hx\|^2 &= \|y\|^2 - 2\langle y, US^{\frac{1}{2}}x\rangle + \|US^{\frac{1}{2}}x\|^2 \\
&= \|y\|^2 - \|U^\mathsf{T}y\|^2 + \|U^\mathsf{T}y\|^2 - 2\langle U^\mathsf{T}y, S^{\frac{1}{2}}x\rangle + \|S^{\frac{1}{2}}x\|^2 \\
&= \langle y, (I_p - UU^\mathsf{T})y\rangle + \sum_{i=1}^{m}(\langle u_i, y\rangle^2 - 2\langle u_i, y\rangle S_{ii}^{\frac{1}{2}}x_i + (S_{ii}^{\frac{1}{2}}x_i)^2) \\
&= \langle y, (I_p - UU^\mathsf{T})y\rangle + \sum_{i=1}^{m} S_{ii}(S_{ii}^{-1}\langle u_i, y\rangle^2 - 2S_{ii}^{-\frac{1}{2}}\langle u_i, y\rangle x_i + x_i^2) \\
&= \langle y, (I_p - UU^\mathsf{T})y\rangle + \sum_{i=1}^{m} S_{ii}((Ty)_i - x_i)^2,
\end{aligned}$$

where u_i is the i^{th} column of U. Note that $P_U = UU^\mathsf{T}$ is the orthogonal projection onto $\mathrm{col}(U)$, so $I - UU^\mathsf{T} = P_{U^\perp}$ is the orthogonal projection onto the orthogonal complement of $\mathrm{col}(U)$. Therefore,

$$\langle y, (I_p - UU^\mathsf{T})y\rangle = \langle y, P_{U^\perp}y\rangle = \langle y, P_{U^\perp}^2 y\rangle = \langle P_{U^\perp}^\mathsf{T}y, P_{U^\perp}y\rangle = \langle P_{U^\perp}y, P_{U^\perp}y\rangle = \|P_{U^\perp}y\|^2. \qquad \square$$

O. OILMM: Missing Data

For a matrix or vector A, let A_o and A_m denote the rows of A corresponding to respectively observed and missing values.

Prop. 11. Consider the OILMM (Mod. 2). For observed outputs y_o, which are a subset of all outputs y, the projection and projected noise are given by

$$T_\mathrm{o} = S^{-\frac{1}{2}}U_\mathrm{o}^\dagger \quad \text{and} \quad \Sigma_{T_\mathrm{o}} = \sigma^2 S^{-\frac{1}{2}}(U_\mathrm{o}^\mathsf{T}U_\mathrm{o})^{-1}S^{-\frac{1}{2}} + D$$

where $U_\mathrm{o}^\dagger$ is the pseudo-inverse of U_o.

Proof. Note that

$$y_\mathrm{o} \sim \mathcal{GP}(H_\mathrm{o}K(t,t')H_\mathrm{o}^\mathsf{T} + \delta[t-t'](\sigma^2 I_\mathrm{o} + H_\mathrm{o}DH_\mathrm{o}^\mathsf{T})),$$

so y_o is an ILMM with basis H_o and observation noise $\sigma^2 I + H_\mathrm{o}DH_\mathrm{o}^\mathsf{T}$. The proof proceeds like that of Prop. 8, also using trick of assuming that $D = 0$ by "absorbing it into $K(t,t')$". To begin with, we have

$$H^\mathsf{T}\Sigma^{-1}H = \sigma^{-2}S^{\frac{1}{2}}U_\mathrm{o}^\mathsf{T}U_\mathrm{o}S^{\frac{1}{2}},$$

so

$$\Sigma_T = T\Sigma T^\mathsf{T} = (H^\mathsf{T}\Sigma^{-1}H)^{-1} = \sigma^2 S^{-\frac{1}{2}}(U_\mathrm{o}^\mathsf{T}U_\mathrm{o})^{-1}S^{-\frac{1}{2}}.$$

Moreover, then

$$T = (H^\mathsf{T}\Sigma^{-1}H)^{-1}H^\mathsf{T}\Sigma^{-1} = (\sigma^2 S^{-\frac{1}{2}}(U_\mathrm{o}^\mathsf{T}U_\mathrm{o})^{-1}S^{-\frac{1}{2}})(\sigma^{-2}S^{\frac{1}{2}}U_\mathrm{o}^\mathsf{T}) = S^{-\frac{1}{2}}(U_\mathrm{o}^\mathsf{T}U_\mathrm{o})^{-1}U_\mathrm{o}^\mathsf{T} = S^{-\frac{1}{2}}U_\mathrm{o}^\dagger.$$

Finally, "pull D back out of $K(t,t')$", which, again, is equivalent to adding it to Σ_T. $\square$

Rem. 1. When using T_o and Σ_{T_o}, in the likelihood computation in Prop. 9, from Prop. 4, it can be seen that two things change: for every time point with missing data,

(1) $H_\mathrm{o}T_\mathrm{o} = U_\mathrm{o}U_\mathrm{o}^\dagger$, so UU^T becomes $U_\mathrm{o}U_\mathrm{o}^\dagger$; and

(2) $\frac{1}{2}\log|\Sigma_{T_\mathrm{o}}|$ gives an extra term $-\frac{1}{2}\log|U_\mathrm{o}^\mathsf{T}U_\mathrm{o}|$.

O.1. Diagonal Approximation of Projected Noise

For a matrix A, let $d[A]$ denote the diagonal matrix resulting from setting the off-diagonal entries of A to zero.

Prop. 12. For Σ_{T_o} from Prop. 11, we have

$$\frac{\|\Sigma_{T_\mathrm{o}} - d[\Sigma_{T_\mathrm{o}}]\|_{\mathrm{op}}}{\|d[\Sigma_{T_\mathrm{o}}]\|_{\mathrm{op}}} \leq \frac{S_{\max}}{S_{\min}} \max_{y \in \mathrm{col}(H):\|y\|=1} \|y_\mathrm{m}\|^2$$

where $\|\bullet\|_{\mathrm{op}}$ denotes the operator norm, and $S_{\min}$ and $S_{\max}$ are the smallest and largest diagonal values of S.

Proof. Let e_i be the i^{th} unit vector. Denote $A = (U_\mathrm{o}^\mathsf{T} U_\mathrm{o})^{-1}$, and let $\lambda_{\min}$ and $\lambda_{\max}$ be the minimum and maximum eigenvalue of A. To begin with,

$$d[A]_{ii} = \langle e_i, Ae_i \rangle \in [\lambda_{\min}, \lambda_{\max}].$$

Let $x \in \mathbb{R}^m$ be such that $\|x\| = 1$. Then

$$\langle x, (A - d[A])x \rangle = \langle x, Ax \rangle - \langle x, d[A]x \rangle \leq \lambda_{\max} - \lambda_{\min}.$$

Similarly,

$$\langle x, (A - d[A])x \rangle \geq -\lambda_{\max} + \lambda_{\min}.$$

Therefore,

$$|\langle x, (A - d[A])x \rangle| \leq \lambda_{\max} - \lambda_{\min},$$

so

$$\|A - d[A]\|_{\mathrm{op}} \leq \lambda_{\max} - \lambda_{\min}$$

Since S is diagonal, we have

$$\Sigma_{T_\mathrm{o}} - d[\Sigma_{T_\mathrm{o}}] = \sigma^2 S^{-\frac{1}{2}} (A - d[A]) S^{-\frac{1}{2}}.$$

Using the derived bound on the operator norm and submultiplicativity of the operator norm, it follows that

$$\|\Sigma_{T_\mathrm{o}} - d[\Sigma_{T_\mathrm{o}}]\|_{\mathrm{op}} \leq \sigma^2 S_{\min}^{-1} (\lambda_{\max} - \lambda_{\min}).$$

Moreover,

$$\|d[\Sigma_{T_\mathrm{o}}]\|_{\mathrm{op}} = \sigma^2 \max_{i=1,\ldots,m} (S_{ii}^{-1} d[A]_{ii} + D_{ii}) \geq \sigma^2 \max_{i=1,\ldots,m} S_{ii}^{-1} d[A]_{ii} \geq \sigma^2 S_{\max}^{-1} \max_{i=1,\ldots,m} d[A]_{ii} \geq \sigma^2 S_{\max}^{-1} \lambda_{\max}.$$

Therefore,

$$\frac{\|\Sigma_{T_\mathrm{o}} - d[\Sigma_{T_\mathrm{o}}]\|_{\mathrm{op}}}{\|d[\Sigma_{T_\mathrm{o}}]\|_{\mathrm{op}}} \leq \frac{S_{\max}}{S_{\min}} \left(1 - \frac{\lambda_{\min}}{\lambda_{\max}}\right).$$

By definition of $\lambda_{\min}$ and $\lambda_{\max}$ and orthogonality of U, we have that

$$\frac{1}{\lambda_{\min}} = \max_{x \in \mathbb{R}^m : \|x\|=1} \|U_\mathrm{o} x\|^2 \leq \max_{x \in \mathbb{R}^m : \|x\|=1} \|Ux\|^2 = 1 \quad \text{and} \quad \frac{1}{\lambda_{\max}} = \min_{x \in \mathbb{R}^m : \|x\|=1} \|U_\mathrm{o} x\|^2.$$

Substitute these results into the bound:

$$\frac{\|\Sigma_{T_\mathrm{o}} - d[\Sigma_{T_\mathrm{o}}]\|_{\mathrm{op}}}{\|d[\Sigma_{T_\mathrm{o}}]\|_{\mathrm{op}}} \leq \frac{S_{\max}}{S_{\min}} \left(1 - \min_{x \in \mathbb{R}^m : \|x\|=1} \|U_\mathrm{o} x\|^2\right) = \frac{S_{\max}}{S_{\min}} \max_{x \in \mathbb{R}^m : \|x\|=1} (1 - \|U_\mathrm{o} x\|^2).$$

By orthogonality of U, for $x \in \mathbb{R}^m$ such that $\|x\| = 1$, we have

$$1 = \|x\|^2 = \|Ux\|^2 = \|U_\mathrm{o} x\|^2 + \|U_\mathrm{m} x\|^2,$$

so $1 - \|U_\mathrm{o} x\|^2 = \|U_\mathrm{m} x\|^2$. Therefore,

$$\max_{x \in \mathbb{R}^m : \|x\|=1} (1 - \|U_\mathrm{o} x\|^2) = \max_{x \in \mathbb{R}^m : \|x\|=1} \|U_\mathrm{m} x\|^2 = \max_{x \in \mathbb{R}^m : \|x\|=1} \|(Ux)_\mathrm{m}\|^2 = \max_{y \in \mathrm{col}(U) : \|y\|=1} \|y_\mathrm{m}\|^2$$

and we conclude by noting that $\mathrm{col}(U) = \mathrm{col}(H)$. □

Cor. 1. Suppose $\|U\|_\infty^2 \leq C/p$ for some $C \geq 1$, and that s outputs are missing. Then

$$\frac{\|\Sigma_{T_\mathrm{o}} - d[\Sigma_{T_\mathrm{o}}]\|_{\mathrm{op}}}{\|d[\Sigma_{T_\mathrm{o}}]\|_{\mathrm{op}}} \leq C \frac{S_{\max}}{S_{\min}} \frac{ms}{p}.$$

Proof. Let $y \in \mathrm{col}(H)$ be such that $\|y\| = 1$. Then $y = Ux$ for some $x \in \mathbb{R}^m$ such that $\|x\| = 1$. Therefore,

$$\|y_\mathrm{m}\|^2 = \sum_{i \in \text{missing}} (Ux)_i^2 \leq \sum_{i \in \text{missing}} \|U_{i:}\|^2 \|x\|^2 = \sum_{i \in \text{missing}} \|U_{i:}\|^2 \leq \frac{Cms}{p},$$

so the result follows from the previous proposition. □

O.2. Variational Approach

Let Y_o be the observed data. Complement Y_o with missing data Y_m such that $Y = Y_\mathrm{o} \cup Y_\mathrm{m}$ is complete. Then a way to deal with missing data is to use variational inference. In particular, assume a Gaussian approximate posterior distribution $q(Y_\mathrm{m})$ over Y_m, and maximise the evidence lower bound (ELBO) $\mathcal{L}$ using gradient-based optimisation:

$$\log p(Y_\mathrm{o}) \geq \mathbb{E}_{q(Y_\mathrm{m})}[\log p(Y)] + H[q(Y_\mathrm{m})] = \mathcal{L}[q(Y_\mathrm{m})],$$

where the expectation can be approximated using the reparametrisation trick (Kingma & Welling, 2013), $\log p(Y)$ can be computed efficiently because Y is complete, and $H[q(Y_\mathrm{m})]$ denotes the entropy of $q(Y_\mathrm{m})$. This approach provides a tractable solution when the missing data are not too numerous.

P. OILMM: Heterogeneous Observation Noise

Although the specification of the observation noise $\Sigma = \sigma^2 I_p + HDH^\mathsf{T}$ in the OILMM does not allow for heterogeneous observation noise, it is possible to set $\Sigma = \mathrm{diag}(\sigma_1^2, \ldots, \sigma_p^2)$ and use Prop. 6 to include Σ in the parametrisation of H: $H = \Sigma^{\frac{1}{2}} U S^{\frac{1}{2}}$. This parametrisation can be interpreted in two ways:

(i) The model has a whitening transform built in. In the projection T, the (noise in the) data will first by whitened by $\Sigma^{-\frac{1}{2}}$. Hence, this parametrisation can be used as a more principled substitute for the usual data normalisation where the outputs are divided by their empirical standard deviation prior to feeding them to the model.

(ii) The basis is orthogonal with respect to a weighted Euclidean inner product: $\langle h_i, h_j \rangle_\Sigma = \sum_{k=1}^p h_{ik} h_{jk} / \sigma_k^2 = 0$ for $i \neq j$. Intuitively, this means that the basis is orthogonal in the usual sense after stretching the i^{th} dimension by σ_i^{-1}.

Although this construction provides additional flexiblity, it does require that $D = 0$ to avoid a circular dependency between Σ and H.

Q. Computational Scaling Experiment (Sec. 4.1) Additional Details

Measurements were performed using a MacBook Pro with a 2.7 GHz Intel Core i7 processor and 16 GB RAM. Code was implemented in Julia 1.0 (Bezanson et al., 2017) and memory and time were measured using the `@allocated` and the `@elapsed` macros, respectively, with the measurements averaged over 10 samples run serially. This means memory reported is the total memory allocated, not peak memory consumption.

R. Point Process Experiment (Sec. 4.2) Additional Details and Analysis

We consider a subset of the extensive rainforest data set credited to Hubbell et al. (2005); Condit (1998); Hubbell et al. (1999). The data features a 1000 m × 500 m rainforest dynamics plot in Barro Colorado Island, Panama. In the survey area, the locations of all *Trichilia tuberculata* (a tree species of the Mahogany family) have been measured (see Fig. 10).

We tackle this spatial point pattern with a log-Gaussian Cox process model, which is an inhomogeneous Poisson process model for count data. The unknown intensity function $\Sigma(x)$ is modelled with a Gaussian process such that $f(x) = \log \Sigma(x)$. Locally-constant intensity in subregions are modelled by discretising the region into np bins (Møller et al., 1998). This leads to a Poisson observation model for each bin. This model reaches posterior consistency in the limit of bin width going

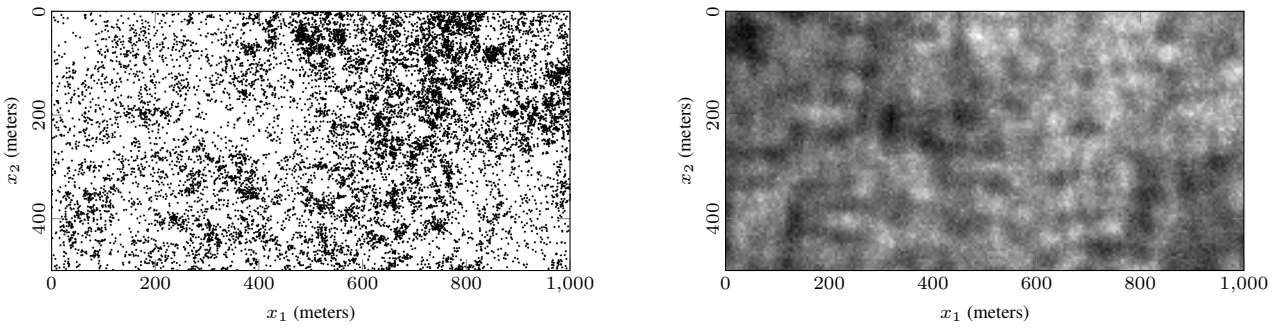
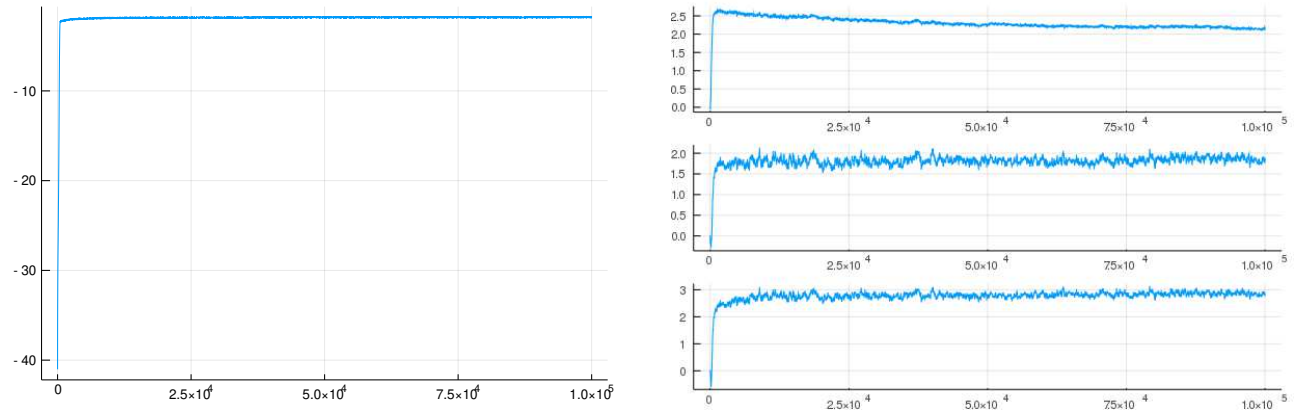


Figure 10. Observations of the rainforest tree locations (left), and posterior mean log-intensity for the log-Gaussian Cox process model (right) with a grid of $np = 20000$ observation bins.



(a) (b)

Figure 11. (a): Log–joint probability per iteration. (b): Hyperparameters per iteration. Shows the length scale, process variance, and nugget variance respectively.

to zero (Tokdar & Ghosh, 2007). The accuracy thus improves with tighter binning. We use a separable Matérn-$5/2$ GP prior over $f(x_1, x_2)$, and discretise the area into a $n \times p = 200 \times 100$ (each bin is $5\,\mathrm{m} \times 5\,\mathrm{m}$) grid with $np = 20000$ grid bins in total, and treat the first dimension as time. The conditional probability of the complete binned data set given the latent GP is therefore

$$p(Y \mid f) \approx \prod_{i=1}^{n} \prod_{j=1}^{p} \mathrm{Poisson}(Y_{ij} \mid a e^{f(r_{ij})}),$$

where r_{ij} is the coordinate of the ij^{th} bin, Y_{ij} is the number of data points in the ij^{th} bin, Y is the $n \times p$ matrix of counts, and a is the area of each bin.

We perform 10^5 iterations of block Gibbs sampling, each of which comprises 10 iterations Elliptical Slice Sampling (Murray et al., 2010; Murray & Adams, 2010) for the Gaussian process given its hyperparameters, and a single iteration of Metropolis Hastings (Hastings, 1970) with proposal distribution $\mathcal{N}(\theta, 0.05^2)$ for the log of the hyperparameters given the latent GP-distributed function. Each step of Elliptical Slice Sampling requires an additional sample from the GP prior at the current hyperparameter values, while each step of Metropolis Hastings requires a log marginal likelihood evaluation. As such approximately 10^6 samples from the prior were drawn, and 10^5 log marginal likelihood calculations undertaken. The kernel is a product of two Matérn-$5/2$ kernels with a shared length scale. A single process variance is utilised, and a nugget term is added. The log of each of the three hyperparameters was given a $\mathcal{N}(0, 1)$ prior. Fig. 11a shows the log joint of the entire state after each iteration, while Fig. 11b shows the progress of each hyperparameter per iteration.

The times in Fig. 4 were obtained via `BenchmarkTools.jl` (Chen & Revels, 2016). The implementation of the standard Kronecker product decomposition trick makes use of `Kronecker.jl`, and Julia's (Bezanson et al., 2017) standard linear algebra libraries, which make use of OpenBLAS and LAPACK to efficiently perform matrix-matrix products and compute

Table 5. Description of the data points associated with the timing experiment from Fig. 4

n	LML Kronecker	OILMM	RNG Kronecker	OILMM
2000	2.45 ± 0.0193	0.403 ± 0.00414	2.45 ± 0.0278	0.478 ± 0.00376
1000	0.365 ± 0.00256	0.0712 ± 0.000369	0.364 ± 0.00451	0.0892 ± 0.000435
200	0.0111 ± 0.000301	$0.00235 \pm 2.53 \times 10^{-5}$	$0.0112 \pm 9.89 \times 10^{-5}$	$0.00318 \pm 1.2 \times 10^{-5}$
100	$0.00237 \pm 8.66 \times 10^{-6}$	$0.000582 \pm 6.55 \times 10^{-7}$	$0.00237 \pm 3.1 \times 10^{-5}$	$0.000792 \pm 8.69 \times 10^{-7}$
40	$0.00044 \pm 4.35 \times 10^{-7}$	$0.000109 \pm 2.22 \times 10^{-7}$	$0.000436 \pm 3.19 \times 10^{-7}$	$0.000141 \pm 2.0 \times 10^{-7}$
20	$9.15 \times 10^{-5} \pm 1.48 \times 10^{-7}$	$2.38 \times 10^{-5} \pm 2.1 \times 10^{-7}$	$9.06 \times 10^{-5} \pm 1.89 \times 10^{-7}$	$3.13 \times 10^{-5} \pm 1.72 \times 10^{-7}$
10	$1.84 \times 10^{-5} \pm 1.54 \times 10^{-7}$	$9.87 \times 10^{-6} \pm 1.08 \times 10^{-7}$	$1.84 \times 10^{-5} \pm 3.02 \times 10^{-7}$	$1.15 \times 10^{-5} \pm 1.17 \times 10^{-7}$

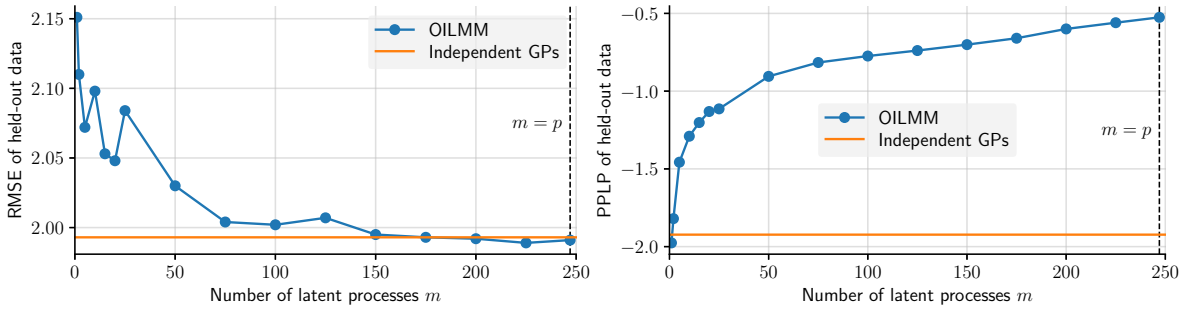


Figure 12. RMSE and PPLP achieved in the temperature extrapolation experiment.

eigendecompositions. The implementation of the state-space GP additionally makes use of `StaticArrays.jl` for efficient stack-allocated matrices, and `Stheno.jl` for GP-related functionality. Timing experiments were conducted on a single CPU core.

When computing the log marginal likelihood, the state-space implementation of the GP makes use of the infinite-horizon trick introduced to the GP literature by Solin et al. (2018). However, this trick is only exploited here once the filtering covariance has converged, which is determined by the point at which the Frobenius norm of the difference between the filtering covariance at the t^{th} and $(t-1)^{\text{th}}$ iterations drops below 10^{-12}. This produces log marginal likelihood evaluations and samples from the prior that are exact for all practical purposes.

R.1. Performance versus Kronecker Trick

Fig. 4 demonstrates that, for the particular approach taken to inference in the Poisson process and, importantly, the dimensions of the data, the Kronecker trick discussed by Saatçi (2012) takes slightly longer to compute log marginal likelihoods and generate samples than does the OILMM implemented in the manner described above. It would of course be unreasonable to assert that the OILMM dominates the Kronecker trick; rather, it seems appropriate to assert that they are competitive with each other in the regime considered.

This is perhaps surprising as the performance of the Kronecker trick is determined almost entirely by a couple of computationally intensive operations, the eigendecomposition and matrix-matrix multiplies. Carefully optimised implementations of these operations exist, and were used, to implement the Kronecker trick. Conversely, the OILMM implementation discussed above comprises many small operations. While our implementation benefits from *e.g.* the `StaticArrays.jl` library, which is suitable for operations on small matrices and vectors, it remains surprising that similar performance was found.

In general we anticipate the OILMM implemented in the described manner be significantly faster on data sets where n is much larger than p, whilst the Kronecker trick will likely do better when n is similar to p.

S. Temperature Extrapolation Experiment (Sec. 4.3) Additional Results

Fig. 12 depicts the RMSE and PPLP achieved in the temperature extrapolation experiment (Sec. 4.3).

T. Large-Scale Climate Model Calibration Experiment (Sec. 4.6) Additional Details and Analysis

We use the variational inducing point method by Titsias (2009), where the positions of the inducing points are initialised to one every two months. All hyperparameters and the locations of the inducing points are optimised until convergence using `scipy`'s implementation of the L-BFGS-B algorithm (Nocedal & Wright, 2006), which takes about 4 hours on a MacBook Pro (2.7 GHz Intel Core i7 processor and 16 GB RAM). The learned length scales were $23.3°$ for latitude and $43.6°$ for longitude.

Fig. 6a shows the empirical correlations and the correlations learned by the OILMM (derived from K_s). In order to get insight into the learned correlations, we hierarchically cluster the models using farthest point linkage with $1 - |\text{corr.}|$ as the distance. Fig. 6b shows the resulting dendrogram, in which models are grouped by their similarity. For two models, the further to the right the branch connecting them is, the less similar the models are.

In Figs. 6a and 6b, HadGEM2 is clearly singled out: it is one of the simplest models, not including several processes that can be found in others, such as ocean & sea-ice, terrestrial carbon cycle, stratosphere, and ocean biogeochemistry (Bellouin et al., 2011). Furthermore, if we inspect the names of the simulators in the groups in Fig. 6b, we observe that often simulators of the same family are grouped together. We observe some interesting cases:

(i) Although IPSL-CM5A-LR and IPSL-CM5A-MR are close, IPSL-CM5B-LR is grouped far apart. It turns out that IPSL-CM5A-LR and IPSL-CM5A-MR are different-resolution versions of the same model, while IPSL-CM5B-LR employs a different atmospheric model.[5]

(ii) ACCESS1.0 and ACCESS1.3 have a similar name, but differ greatly in their implementation: ACCESS1.0 is the basic model, while ACCESS1.3 is much more aspirational, including experimental atmospheric physics models and a particular land surface model (Bi et al., 2013).

(iii) The distance between BCC_CSM1.1(m) and BCC_CSM1.1 can be explained by the more realistic surface air temperature predictions obtained by the former (Wu et al., 2014), which is exactly the quantity we study.

Finally, Fig. 6c shows predictions for four latent processes ($i_s = 1, 2$ with $i_r = 1, 2$). The first spatial eigenvector ($i_r = 1$) is constant in space; combined with the strongest eigenvector of K_s ($i_s = 1$), we obtain a strong signal constituting seasonal temperature changes.

[5] See `https://portal.enes.org/models/earthsystem-models/ipsl/ipslesm`.