Supplementary Material: Scaling Multidimensional Gaussian Processes using Projected Additive Approximations

A. Conclusions of VBEM

A number of conclusions can be drawn from the fact that it is sufficient to run the standard state-space model inference procedure using the pseudo observations to update the factors in the E step. First, since VBEM iterations are guaranteed to converge, any moment computed using the factors $q(\mathbf{Z}_i)$ is also guaranteed to converge. Convergence of these moments is important because they are used to learn the hyperparameters. Second, since the *true* posterior $p(\mathbf{Z}_1, \ldots, \mathbf{Z}_D|\mathbf{y}, \theta)$ is a large joint Gaussian over all the latent variables, $\mathbb{E}_q(\mathbf{Z})$ (the mean of the *approximate* posterior) will be equal to the true posterior mean. This is true because the true posterior is Gaussian (unimodal with the mean as its mode) and the VB approximation is mode-seeking. This is easily shown, since the mode of a multivariate Gaussian will have the same mode as the product of its marginals. Specifically, since the VB approximation is a product of its exclusive marginals, its mode will be reached when the marginals are at their mode, specifically

$$\max_{\mathbf{Z}} q(\mathbf{Z}) = \prod_{i=1}^{D} \max_{\mathbf{Z_i}} q(\mathbf{Z}_i).$$

Thus, since the marginals are Gaussian, the VB approximation is Gaussian, and its mode equals the true posterior mean (conditioned on θ).

Although the VB approximation of the posterior mean is unbiased, as is typical for variational methods, the posterior covariance will be underestimated because $\mathrm{KL}(q(\mathbf{Z})||p(\mathbf{Z}|\mathbf{y}, \theta))$ is an *exclusive* divergence measure (Minka, 2005). As a result, this can cause a sense of false confidence in the estimate, and could discard important off diagonal covariance information (Barber et al., 2011).

B. PPGPR Algorithm and Derivations

Here we expand the PPGPR algorithm and show examples of the derivations. For brevity, we will expand θ to also include the projection weights $\{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_M\}$.

To begin, Figure 1 illustrates a simple example to help clarify the Projected GP structure. It can be seen from the figure that the Additive GP is a special case of the Projected GP with unitary projections. Also, notice that when transforming the multidimensional Full GP to a GMP model, the input space was separated and sorted for each dimension.

The core PPGPR algorithm is detailed in Algorithm 1. As an instructive example, we give here the typical structure of these matrices and show how to construct them in the PPGPR algorithm. We use the Mateŕn(3/2) kernel (e.g., (Rasmussen & Williams, 2006), and we show how to derive the key algorithmic steps in Algorithm 1.

In order to construct the GMP model, we must connect the SDE of Equation (5) (main text) to a GMP model. This involves calculating transition and observation matrices $\mathbf{\Phi}$ and $\mathbf{Q}$.

$$\textbf{Initial state :}\quad p(\mathbf{z}(x_1)) = \mathcal{N}(\mathbf{z}(x_1); \boldsymbol{\mu}, \mathbf{V}). \tag{S-1}$$

$$\textbf{State update :}\quad p(\mathbf{z}(x_t)|\mathbf{z}(x_{t-1})) = \mathcal{N}(\mathbf{z}(x_t); \mathbf{\Phi}_{t-1}\mathbf{z}(x_{t-1}), \mathbf{Q}_{t-1}). \tag{S-2}$$

$$\textbf{Emission :}\quad p(y(x_t)|\mathbf{z}(x_t)) = \mathcal{N}(y(x_t); \mathbf{h}^\top \mathbf{z}(x_t), \sigma_n^2). \tag{S-3}$$

where the z vector is the state vector defined in Eq. (6). The vector $\mathbf{h}$ simply picks out the first element of the vector $\mathbf{z}(x_t)$ which corresponds to the latent function value inferred at location x_t. Deriving the $\mathbf{\Phi}$, and $\mathbf{Q}$

matrices involves finding the $\mathbf{A}$ (main text Eq. (5)) matrix using the Fourier transform of the covariance function, and solving the SDE of Eq. (1). Earlier works (Hartikainen & Särkkä, 2010; Saatci, 2011) derived these terms for different kernels families. Extending these matrices for projected inputs, as in PPGPR, is straight forward since each projection will result in a new GMP as in Eqs. (S-1)-(S-3). As an example, the $\mathbf{\Phi}$ and $\mathbf{Q}$ matrices for the m-th projection will result in a GMP with matrices:

$$\mathbf{\Phi}_{m_{t-1}} = \frac{1}{\exp\left(\lambda_m \delta_{m_t}\right)} \begin{bmatrix} (\lambda_m \delta_{m_t} + 1) & \delta_{m_t} \\ -(\lambda_m^2 \delta_{m_t}) & (1 - \lambda_m \delta_{m_t}) \end{bmatrix}, \tag{S-4}$$

$$\mathbf{Q}_{m_{t-1}} = \begin{bmatrix} \frac{1}{4\lambda_m^3} - \frac{4\delta_{m_t}^2 \lambda_m^2 + 4\delta_{m_t}\lambda_m + 2}{8\lambda_m^3 \exp(2\delta_{m_t}\lambda_m)} & \frac{\delta_{m_t}^2}{2\exp(2\delta_{m_t}\lambda_m)} \\ \frac{\delta_{m_t}^2}{2\exp(2\delta_{m_t}\lambda_m)} & \frac{1}{4\lambda_m} - \frac{2\delta_{m_t}^2\lambda_m^2 - 2\delta_{m_t}\lambda_m + 1}{4\lambda_m \exp(2\delta_{m_t}\lambda_m)} \end{bmatrix}, \tag{S-5}$$

where $\delta_{m_t} = \mathbf{w}_m \left(\mathbf{x}_t - \mathbf{x}_{t-1}\right)$ is the m-th linear projection of the input space, and λ_m is the m-th covariance lengthscale hyperparameter. Notice, that in PPGPR, the projections are chosen sequentially in a greedy form, and it is never necessary to consider all the projections simultaneously. For brevity, we will omit the m subscript notation from now on, understanding that the $\mathbf{\Phi}$, and $\mathbf{Q}$ matrices correspond to the current projection.

In order to learn the optimal hyperparameters of the covariance function we calculate the negative log marginal likelihood (NLML) and its derivatives with respect to the hyperparameters. Finding the NLML of a GMP is simple as the Markov property induce conditional independence between the links of the chain. Hence, the NLML can be written as

$$-\log\left(p(z(x_1), z(x_2), \ldots, z(x_N)|\theta)\right) = -\log \prod_{i=1}^{N} p(z(x_i)|z(x_{i-1}), \theta) = -\sum_{i=1}^{N} \log p(z(x_i)|z(x_{i-1}), \theta). \tag{S-6}$$

For GMP, the terms in the sum can be efficiently calculated by running a Kalman filter on the chain.

The log marginal likelihood ($\log Z(\theta)$) can be written in closed form as

$$L(\mathbf{y}(i(t)), \theta) = -\frac{1}{2}\left[\log 2\pi + \log\left(\mathbf{h}^\top \mathbf{P}_{t-1}(\theta)\mathbf{h} + \sigma_n^2\right) + \frac{\left(y(i(t)) - \mathbf{h}^\top \mathbf{\Phi}_{t-1}(\theta)\boldsymbol{\mu}_{t-1}^{(f)}\right)^2}{\mathbf{h}^\top \mathbf{P}_{t-1}(\theta)\mathbf{h} + \sigma_n^2}\right] \tag{S-7}$$

where the matrix $\mathbf{P}_t$ is the estimated covariance, and $\boldsymbol{\mu}_t^{(f)}$ is the estimated state, of the forward pass Kalman filter. The $i(t)$ function sorts the observations $\mathbf{y}$ according to the new projected scalar input. The derivatives ($\frac{d\log Z(\theta)}{d\theta_i}$) are calculated in the same manner and are also summed following the Kalman forward pass.

The introduction of the projection weights in PPGPR will require the to calculate NLML derivatives with respect to a weight components of the projection vector ($\frac{d\log Z(\theta)}{dw_i}, i = 1, \ldots, D$). Since the projections weights are only in the δ_{m_t} term, and since these δ_m terms only appear in $\mathbf{\Phi}$ and $\mathbf{Q}$ (Eqs. (S-4) and (S-5)), the log marginal likelihood derivative can be written as

$$\frac{dL(\mathbf{y}(i(t)), \theta)}{dw_i} = \left(\frac{dL(\mathbf{y}(i(t)), \theta)}{d\mathbf{P}_{t-1}}\frac{d\mathbf{P}_{t-1}}{d\delta_{m_t}} + \frac{dL(\mathbf{y}(i(t)), \theta)}{d\mathbf{\Phi}_{t-1}}\frac{d\mathbf{\Phi}_{t-1}}{d\delta_{m_t}}\right)\left(\mathbf{x}_{t_i} - \mathbf{x}_{t-1_i}\right) \tag{S-8}$$

where,

$$\frac{d\mathbf{P}_{t-1}}{d\delta_{m_t}} = \frac{d\mathbf{P}_{t-1}}{d\mathbf{\Phi}_{t-1}}\frac{d\mathbf{\Phi}_{t-1}}{d\delta_{m_t}} + \frac{d\mathbf{P}_{t-1}}{d\mathbf{Q}_{t-1}}\frac{d\mathbf{Q}_{t-1}}{d\delta_{m_t}}. \tag{S-9}$$

Full details of the development, including important proofs, can be found in our preliminary work (Saatci, 2011).

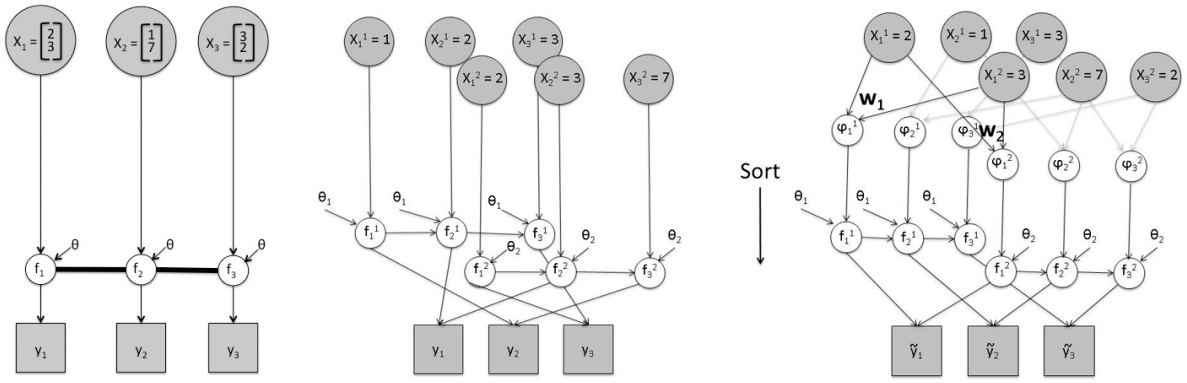

Figure 1. A simple example to illustrate the different models: Full GP, Additive GP, and Projected GP. The Full GP is shown on the left for a two dimensional input space. The bold line represents a fully connected graph. The Additive GP, and projected GP are shown on the middle, and right, respectively. Notice that in the projected GP a sort step is performed after the projections to make it a Gauss-Markov process. The sorted outputs are written with a tilde.

References

Barber, D., Cemgil, A. T., and Chiappa, S. *Bayesian Time Series Models*. Cambridge University Press, 2011.

Hartikainen, J. and Särkkä, S. Kalman filtering and smoothing solutions to temporal Gaussian process regression models. In *Machine Learning for Signal Processing (MLSP)*, pp. 379–384, Kittilä, Finland, August 2010. IEEE.

Minka, T. Divergence measures and message passing. Technical report, Microsoft Research, 2005.

Rasmussen, C.E. and Williams, C.K.I. *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

Saatci, Y. *Scalable Inference for Structured Gaussian Process Models*. PhD thesis, University of Cambridge, 2011.

Algorithm 1 Gaussian Process Regression using SSMs

Input: Jointly sorted training and test input locations $\mathbf{x}$. Targets $\mathbf{y}$ associated with training inputs. State transition function stfunc that returns $\mathbf{\Phi}$ and $\mathbf{Q}$ matrices. Hyperparameters θ.

outputs: Log-marginal likelihood $\log Z(\theta)$ and its derivatives. Predictive means $\boldsymbol{\mu}_\star$ and variances $\mathbf{v}_\star$. E-step moments: $\mathbb{E}(\mathbf{z}_t\mathbf{z}_t^\top)$, $\mathbb{E}(\mathbf{z}_t\mathbf{z}_{t-1}^\top)$

$\boldsymbol{\mu}_0^{(f)} \leftarrow \boldsymbol{\mu}$; $\mathbf{V}_0^{(f)} \leftarrow \mathbf{V}$; $Z(\theta) = 0$
for $t \leftarrow 1 \dots K$ **do**
  if $t > 1$ **then**
    $[\mathbf{\Phi}_{t-1}, \mathbf{Q}_{t-1}] \leftarrow \text{stfunc}(\theta, \mathbf{x}(t) - \mathbf{x}(t-1))$
  else
    $[\mathbf{\Phi}_{t-1}, \mathbf{Q}_{t-1}] \leftarrow \text{stfunc}(\theta, \infty)$
  end if
  $\mathbf{P}_{t-1} \leftarrow \mathbf{\Phi}_{t-1}\mathbf{V}_{t-1}^{(f)}\mathbf{\Phi}_{t-1}^\top + \mathbf{Q}_{t-1}$
  $\mathbf{G}_t = \mathbf{P}_{t-1}\mathbf{h}\left(\mathbf{h}^\top\mathbf{P}_{t-1}\mathbf{h} + \sigma_n^2\right)^{-1}$
  $L(\mathbf{y}(i(t)), \theta) = \log\left(P\left(\mathbf{y}(i(t))|\mathbf{h}^\top\mathbf{\Phi}_{t-1}\boldsymbol{\mu}_{t-1}^{(f)}, \mathbf{h}^\top\mathbf{P}_{t-1}\mathbf{h} + \sigma_n^2\right)\right)$
  $\log Z(\theta) \leftarrow \log Z(\theta) + L(\mathbf{y}(i(t)), \theta)$
  $\frac{d\log Z(\theta)}{d\theta_i} \leftarrow \frac{d\log Z(\theta)}{d\theta_i} + \frac{dL(\mathbf{y}(i(t)),\theta)}{d\theta_i}$
  $\boldsymbol{\mu}_t^{(f)} = \mathbf{\Phi}_{t-1}\boldsymbol{\mu}_{t-1}^{(f)} + \mathbf{G}_t[\mathbf{y}(i(t)) - \mathbf{h}^\top\mathbf{\Phi}_{t-1}\boldsymbol{\mu}_{t-1}^{(f)}]$
  $\mathbf{V}_t^{(f)} = \mathbf{P}_{t-1} - \mathbf{G}_t\mathbf{h}^\top\mathbf{P}_{t-1}$
end for
$\boldsymbol{\mu}_K \leftarrow \boldsymbol{\mu}_K^{(f)}$; $\mathbf{V}_K \leftarrow \mathbf{V}_K^{(f)}$; $\boldsymbol{\mu}_\star(K) \leftarrow \mathbf{h}^\top\boldsymbol{\mu}_K$; $\mathbf{v}_\star(K) \leftarrow \mathbf{h}^\top\mathbf{V}_K\mathbf{h}$
$\mathbb{E}(\mathbf{z}_K\mathbf{z}_K^\top) \leftarrow \mathbf{V}_K + \boldsymbol{\mu}_K\boldsymbol{\mu}_K^\top$
$\mathbb{E}(\mathbf{z}_K\mathbf{z}_{K-1}^\top) \leftarrow (I_D - \mathbf{G}_K\mathbf{h})\mathbf{\Phi}_{K-1}\mathbf{V}_{K-1}$
for $t \leftarrow K-1 \dots 1$ **do**
  $\mathbf{L}_t \leftarrow \mathbf{V}_t\mathbf{\Phi}_t^\top\mathbf{P}_t^{-1}$
  $\boldsymbol{\mu}_t \leftarrow \boldsymbol{\mu}_t^{(f)} + \mathbf{L}_t\left(\boldsymbol{\mu}_{t+1} - \mathbf{\Phi}_t\boldsymbol{\mu}_t^{(f)}\right)$; $\boldsymbol{\mu}_\star(t) \leftarrow \mathbf{h}^\top\boldsymbol{\mu}_t$
  $\mathbf{V}_t \leftarrow \mathbf{V}_t^{(f)} + \mathbf{L}_t\left(\mathbf{V}_{t+1} - \mathbf{P}_t\right)\mathbf{L}_t^\top$; $\mathbf{v}_\star(t) \leftarrow \mathbf{h}^\top\mathbf{V}_t\mathbf{h}$
  $\mathbb{E}(\mathbf{z}_t\mathbf{z}_t^\top) \leftarrow \mathbf{V}_t + \boldsymbol{\mu}_t\boldsymbol{\mu}_t^\top$
  if $t < K-1$ **then**
    $\mathbb{E}(\mathbf{z}_t\mathbf{z}_{t-1}^\top) \leftarrow \mathbf{V}_{t+1}^{(f)}\mathbf{L}_t^\top + \mathbf{L}_{t+1}\left(\mathbb{E}(\mathbf{z}_{t+1}\mathbf{z}_t^\top) - \mathbf{\Phi}_{t+1}\mathbf{V}_{t+1}\right)\mathbf{L}_t^\top$
  end if
end for