Orthogonal Bases for Multi-Output Gaussian Processes

Wessel Bruinsma

University of Cambridge, CBL

20 December 2019

Contents

Introduction

Gaussian Processes

- A powerful and popular probabilistic modelling framework for nonlinear functions.
- Definition: $f \sim \mathcal{GP}(m, k)$ if, for all $(t_1, \ldots, t_n) \in \mathcal{T}^n$,

$$\begin{bmatrix} f(t_1) \\ \vdots \\ f(t_n) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} m(t_1) \\ \vdots \\ m(t_n) \end{bmatrix}, \begin{bmatrix} k(t_1, t_1) & \cdots & k(t_1, t_n) \\ \vdots & \ddots & \vdots \\ k(t_n, t_1) & \cdots & k(t_n, t_n) \end{bmatrix} \right).$$

- Inference and learning: $O(n^3)$ time and $O(n^2)$ memory.

Multi-Output Gaussian Processes

- Multi-output GPs go long way back (Matheron, 1969).
- Vector-valued mean function m and matrix-valued kernel K:

$$m\colon \mathcal{T} \to \mathbb{R}^p, \quad K\colon \mathcal{T}^2 \to \mathbb{R}^{p\times p},$$

number of outputs

input space (time)

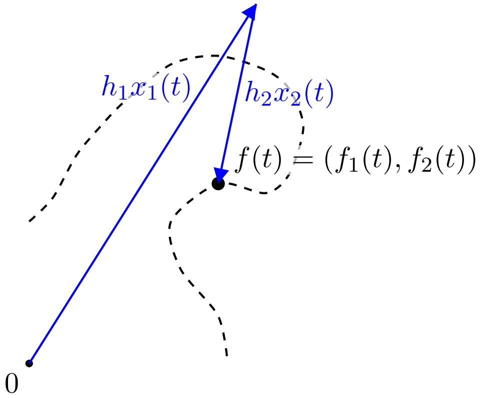
$$m(t) = \begin{bmatrix} \mathbb{E}[f_1(t)] \\ \vdots \\ \mathbb{E}[f_p(t)] \end{bmatrix},$$

$$K(t,t') = \begin{bmatrix} \operatorname{cov}(f_1(t), f_1(t')) & \cdots & \operatorname{cov}(f_1(t), f_p(t')) \\ \vdots & \ddots & \vdots \\ \operatorname{cov}(f_p(t), f_1(t')) & \cdots & \operatorname{cov}(f_p(t), f_p(t')) \end{bmatrix}.$$

- Inference and learning: $O(n^3 p^3)$ time and $O(n^2 p^2)$ memory.
  - Often alleviated by exploiting structure in K.

The Linear Mixing Model

Fixed Basis with Varying Coefficients



$$f(t) = h_1 x_1(t) + h_2 x_2(t) = \underbrace{\begin{bmatrix} h_1 & h_2 \end{bmatrix}}_{H} \underbrace{\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}}_{x(t)}.$$

The Linear Mixing Model

“mixing matrix”

Definition (Linear Mixing Model)

$$x \sim \mathcal{GP}(0, K(t,t')), \quad f(t) \,|\, H, x = Hx(t), \quad y \,|\, f \sim \mathcal{GP}(f(t), \Lambda).$$

“latent processes”

- f is p-dimensional, x is m-dimensional, and H is $p \times m$.
  - Often $p \gg m$.
- Equivalently, $y \sim \mathcal{GP}(0, HK(t,t')H^\mathsf{T} + \Lambda)$.
- Generalisation of FA to time series setting.
- Fixed spatial correlation: $\mathbb{E}[f(t)f^\mathsf{T}(t)] = HH^\mathsf{T}$ if $K(t,t) = I$.
- Instantaneous mixing: $f(t)$ depends on $x(t')$ only for $t = t'$.
- Inference and learning: $O(n^3m^3)$ time and $O(n^2m^2)$ memory.

Exploiting the Low-Rank Structure

Proposition

Let T be the $(m \times p)$-matrix $(H^\mathsf{T}\Lambda^{-1}H)^{-1}H^\mathsf{T}\Lambda^{-1}$. Then

$$\text{conditioning } y \,|\, f \sim \mathcal{GP}(f(t), \Lambda) \text{ on data } Y\text{: } O(n^3 p^3)$$

$$\iff$$

$$\text{conditioning } \underbrace{Ty}_{u} \,|\, f \sim \mathcal{GP}(\underbrace{Tf(t)}_{x(t)}, T\Lambda T^\mathsf{T}) \text{ on data } TY\text{: } O(n^3 m^3).$$

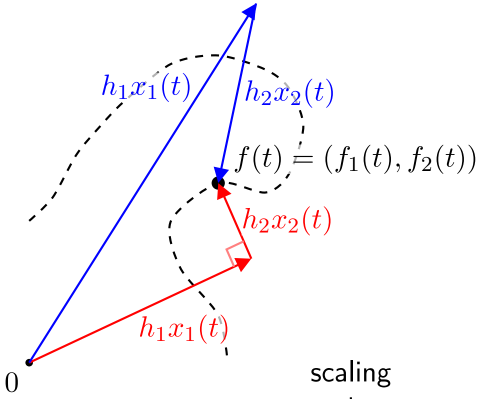
- T: "y-space" $\to$ "x-space".
- $Ty = \arg\min_x \|\Lambda^{\frac{1}{2}}(y - Hx)\|_2$.

$$\begin{tikzcd}
Y \arrow[r, "\text{inference}", "O(n^3p^3)"'] \arrow[d] & p(y \,|\, Y) \\
TY \arrow[r, "\text{inference}", "O(n^3m^3)"'] & p(u \,|\, TY) \arrow[u]
\end{tikzcd}$$

What if $T\Lambda T^\mathsf{T}$ were diagonal? Then inference decouples into independent problems!

The Orthogonal Linear Mixing Model

Fixed Orthogonal Basis with Varying Coefficients



$$\underbrace{\begin{bmatrix} h_1 & h_2 \end{bmatrix} = H = \overset{\text{scaling}}{\overset{\downarrow}{U S^{\frac{1}{2}}}}}_{}.$$

$$\underset{\text{orthogonal}}{\underset{\uparrow}{U}}$$

The Orthogonal Linear Mixing Model

Definition (Orthogonal Linear Mixing Model)

With $K(t,t) = I$, $H = US^{\frac{1}{2}}$, and $\Lambda = \sigma^2 I + HDH^{\mathsf{T}}$,

$$x \sim \mathcal{GP}(0, K(t,t')), \quad f(t) \,|\, H, x = Hx(t), \quad y \,|\, f \sim \mathcal{GP}(f(t), \Lambda).$$

- Generalisation of PPCA (Tipping and Bishop, 1999) to time series setting.
- Like GPFA (Yu et al., 2009), but orthogonality built in.
- General spatial correlation: $\mathbb{E}[f(t)f^{\mathsf{T}}(t)] = USU^{\mathsf{T}}$.
  - $\Rightarrow$ Suggests way to initialise U and S.

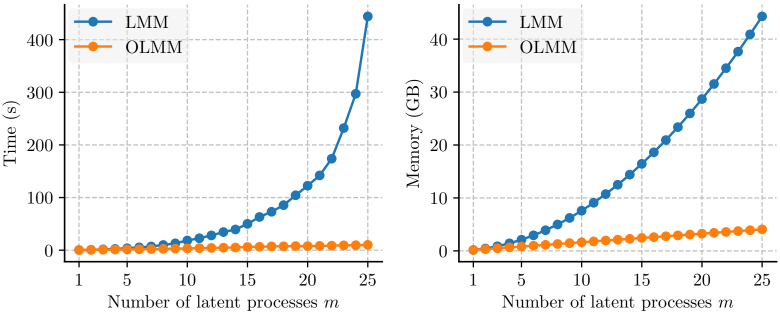
Inference and Learning

- Image of noise: $T\Lambda T^\mathsf{T} = \sigma^2 S^{-1} + D$. Diagonal!

$$\begin{tikzcd}
Y \arrow[r, "\text{inference}", "O(n^3 p^3)"'] \arrow[dd] & p(y \,|\, Y) \\
 & p(u_1 \,|\, (TY)_{1:}) \arrow[u, bend right] \\
TY \arrow[ur, "\text{inference}", "O(n^3)"'] \arrow[dr, "\text{inference}", "O(n^3)"'] & \vdots \\
 & p(u_m \,|\, (TY)_{m:}) \arrow[uuu, bend right]
\end{tikzcd}$$

- Inference and learning: $O(n^3 m)$ time and $O(n^2 m)$ memory.
  - $\Rightarrow$ Linear scaling in the number of degrees of freedom!
- Trivially compatible with one-dimensional scaling techniques.

Inference and Learning (2)



Proposition

The evidence $\log p(Y)$ is convex in U.

Arbitrary Likelihoods

Recapitulation of the OLMM

- $+$ Computationally efficient
  - $+$ Linear scaling in number of degrees of freedom m
  - $+$ Trivially compatibly with one-dimensional scaling techniques
  - $+$ Convex in U
- $+$ Easy to implement
- $\pm$ Expressivity
  - $\pm$ Restricted to orthogonal bases
  - $-$ Linear correlations
- $-$ Cannot handle missing data
- $-$ Cannot handle imhomogeneous observation noise

Goal: arbitrary $p(y \mid f)$ whilst retaining computational efficiency.

Application: Neural Networks with Time-Varying Weights

Task: predict mapping $z \mapsto \phi(z, t)$ that slowly varies with time.

- ???
- Economics?

Generative model:

$$w \sim \mathrm{OLMM}, \quad \phi(z, t) \,|\, w = \mathrm{NN}_{w(t)}(z).$$

Inference: VI with an OLMM as computationally efficient q.

Variational Inference

Prior:

$$p(y, f) = \underbrace{p(y \,|\, f)}_{\text{arbitrary likelihood}} p(f).$$

Approximate posterior (Johnson et al., 2016):

$$q(f) = \frac{1}{Z} p(f) \underbrace{p(\hat{y} \,|\, f, \hat{\Lambda})}_{\text{likelihood ``conjugate'' to the OLMM}}.$$

ELBO:

$$\mathcal{L}(\hat{y}, \hat{\Lambda}) = \underbrace{\log Z}_{\text{evidence of pseudo-observations}} + \overbrace{\mathbb{E}_{q(f)}[\underbrace{\log p(y \,|\, f)}_{\text{tractable / Monte Carlo}} - \underbrace{\log p(\hat{y} \,|\, f, \hat{\Lambda})}_{\text{tractable}}]}^{\text{likelihood correction}}.$$

- \+ Pseudo-evidence $\log Z$ and approximate posterior $q(f)$ cheap
- \+ Easy to implement
- ± Prior $p(f)$ shared with $q(f)$
- − For k pseudo-points, $O(kp)$ parameters

Variational Inference, Take 2

Approximate posterior:

$$
\begin{aligned}
q(f) &= \frac{1}{Z} p(f) p(\hat{y} \mid f, \hat{\Lambda}) \\
&= \frac{1}{Z} p(f) p(T\hat{y}\ \underbrace{T\hat{y}}_{\hat{u}} \mid f, T\hat{\Lambda}T^\top\ \underbrace{T\hat{\Lambda}T^\top}_{\hat{D}}) = \frac{1}{Z} p(f) p(\hat{u} \mid f, \hat{D}).
\end{aligned}
$$

ELBO:

$$
\mathcal{L}(\hat{u}, \hat{D}) = \log Z + \mathbb{E}_{q(f)}[\log p(y \mid f) - \log p(\hat{u} \mid f, \hat{D})].
$$

+ For k pseudo-points, $O(km)$ parameters

Naturally Mean Field

- $q(f) \propto p(f)p(\hat{u} \,|\, f, \hat{D})$: EP-style approximation in VI setting.
- Consider traditional pseudo-point method (Titsias, 2009):

$$
\begin{aligned}
q(f) &= \int p(f \,|\, \hat{f}) q(\hat{f}) \,\mathrm{d}\hat{f} \\
&= \int p(f \,|\, T\hat{f}\underbrace{T\hat{f}}_{\hat{x}}) q(T\hat{f}) \,\mathrm{d}T\hat{f} = \int p(f \,|\, \hat{x}) q(\hat{x}) \,\mathrm{d}\hat{x}.
\end{aligned}
$$

- Equivalent if $q(\hat{x}_i) = \mathcal{N}(\hat{u}_i, (K_{x_i}^{-1} + \hat{D}_i^{-1})^{-1})$.
- When does $q^*(\hat{x})$ factorise over the latent processes?

$$
\prod_{i=1}^{m} q^*(\hat{x}_i) \stackrel{?}{=} q^*(\hat{x}) = e^{-\mathcal{L}^*} p(\hat{x}) \exp\langle \log p(y \,|\, x) \rangle_{p(x \,|\, \hat{x})}.
$$

1. OLMM!
2. $y \,|\, x \sim \mathcal{GP}(H\phi(x(t)), \Lambda)$ with ϕ a pointwise nonlinearity.

(Preliminary) Conclusions

(Preliminary) Conclusions

- Orthogonal basis decouples inference into independent problems.
- OLMM can (maybe) function as a computationally efficient prior / approximate posterior in larger spatio–temporal models.

- Experiments on real-world data!

Outstanding Questions

- Fixing a computational budget, how does OLMM compare?
- How restrictive is the orthogonality assumption?
  - For a given LMM, how close is the closest OLMM?
- When is H identifiable? Connection to ICA?
- How does learned basis compare to other methods, e.g. PCA?
  - Can pointwise nonlinearity ϕ improve learned basis?
- Can orthogonality alleviate downsides of mean-field VI?

These slides: `https://wessel.page.link/olmm`.

Appendix

References

Johnson, M. J., Duvenaud, D., Wiltschko, A. B., Datta, S. R., & Adams, R. P. (2016). Composing graphical models with neural networks for structured representations and fast inference. *arXiv preprint arXiv:1603.06277*. eprint: `https://arxiv.org/abs/1603.06277`

Matheron, G. (1969). Le krigeage universel. In *Cahiers du centre de morphologie mathématique de fontainebleau* (Vol. 1). École nationale supérieure des mines de Paris.

Tipping, M. E., & Bishop, C. M. (1999). Probabilistic principal component analysis. *Journal of the Royal Statistical Society, Series B*, *61*(3), 611–622.

Titsias, M. K. (2009). Variational learning of inducing variables in sparse Gaussian processes. *Artificial Intelligence and Statistics*, *12*, 567–574.

References (2)

Yu, B. M., Cunningham, J. P., Santhanam, G., Ryu, S. I., Shenoy, K. V., & Sahani, M. (2009). Gaussian-Process factor analysis for low-dimensional single-trial analysis of neural population activity. *Advances in Neural Information Processing Systems*, *21*, 1881–1888. Retrieved from `http://papers.nips.cc/paper/3494-gaussian-process-factor-analysis-for-low-dimensional-single-trial-analysis-of-neural-population-activity.pdf`