Modular Gaussian Processes for Transfer Learning

Pablo **Moreno-Muñoz**
Section for Cognitive Systems, Technical University of Denmark (DTU)

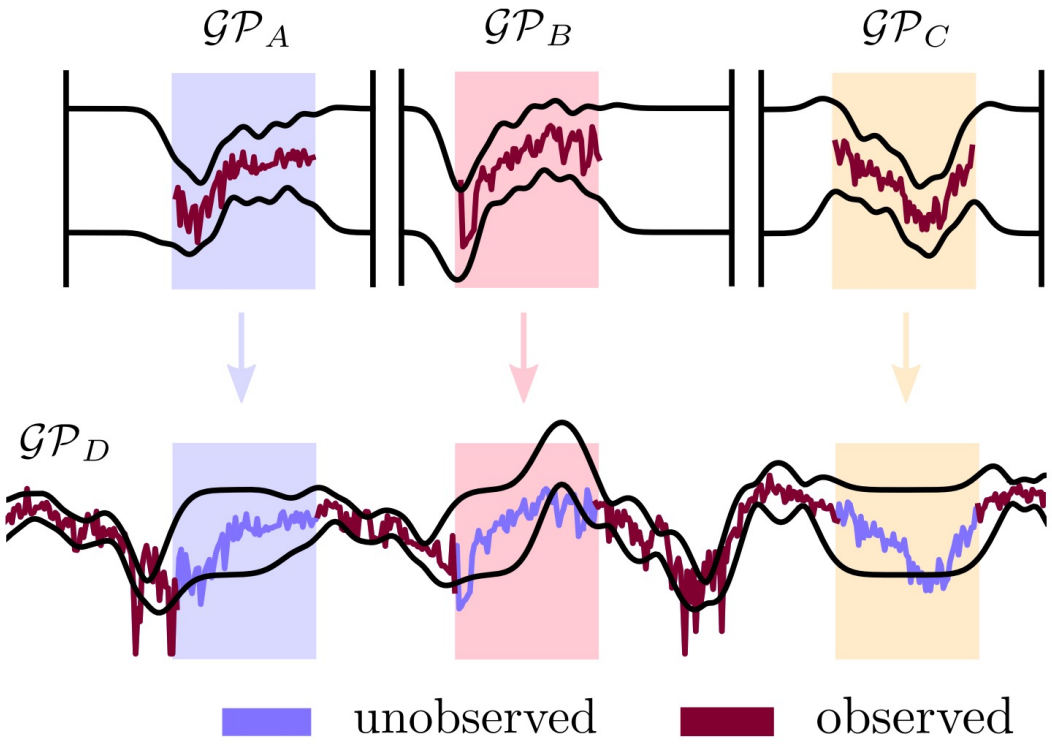
Antonio **Artés-Rodríguez**
Dept. of Signal Theory and Communications, Universidad Carlos III de Madrid, Spain
Evidence-Based Behavior (eB2)

Mauricio A. **Álvarez**
Dept. of Computer Science, University of Sheffield, UK

NEURAL INFORMATION PROCESSING SYSTEMS
2021

Problem

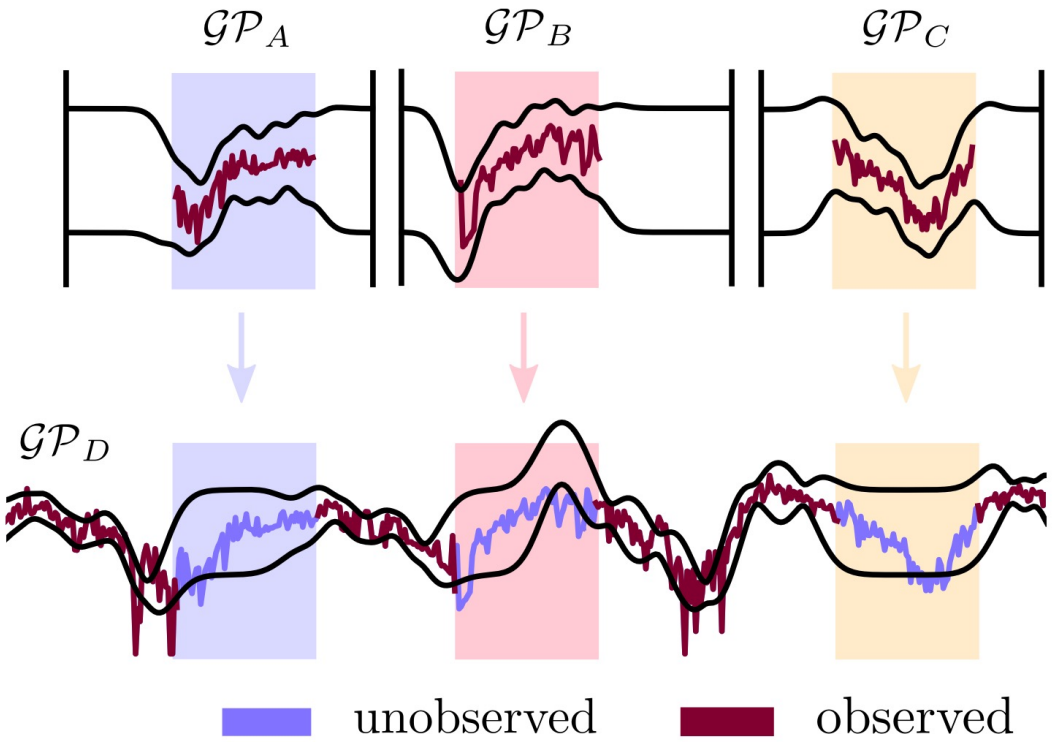
A **Gaussian process** (GP) model is trained from *N* data points, stored in our computer. At a later time, **new data** are observed. The combination of datasets for training might be **inconvenient** because of



Problem

A **Gaussian process** (GP) model is trained from *N* data points, stored in our computer. At a later time, **new data** are observed. The combination of datasets for training might be **inconvenient** because of

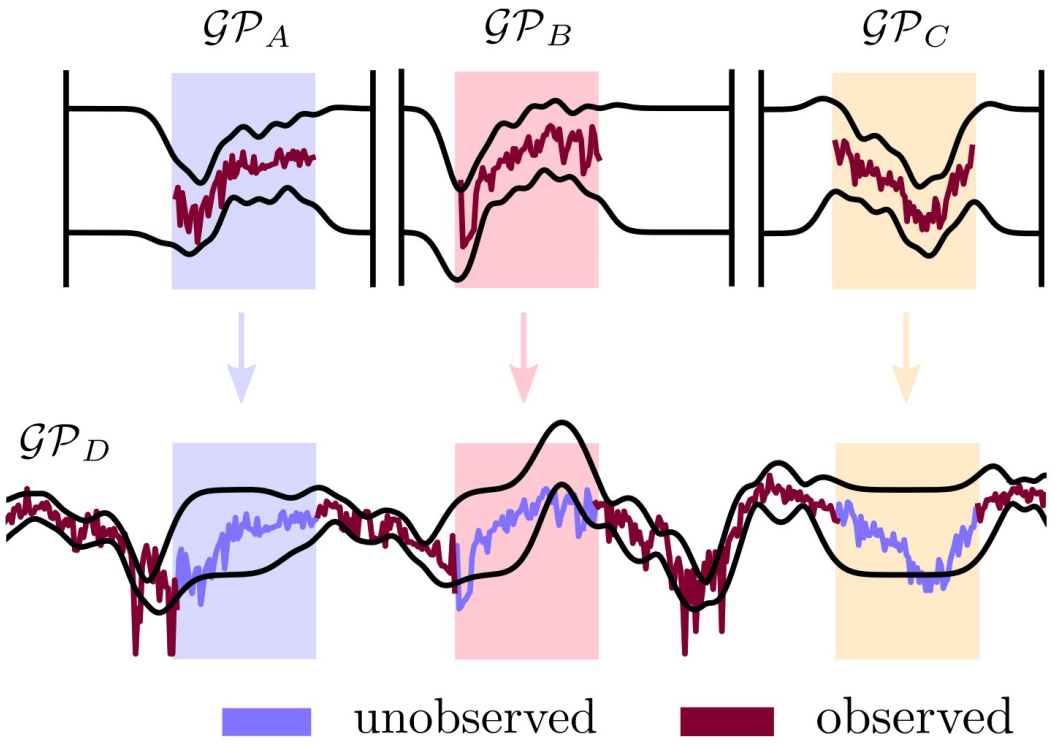
1) the need of *centralising* the data
2) the rising data-dependent *computational cost*
3) the *obsolescence* of the fitted model



Problem

A **Gaussian process** (GP) model is trained from N data points, stored in our computer. At a later time, **new data** are observed. The combination of datasets for training might be **inconvenient** because of

1) the need of *centralising* the data
2) the rising data-dependent *computational cost*
3) the *obsolescence* of the fitted model



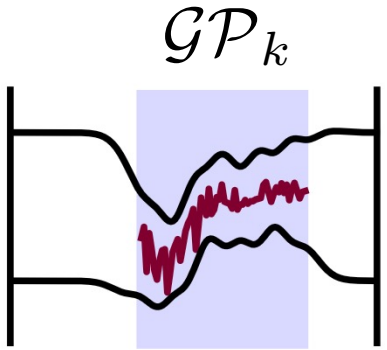
Contribution

A new framework based on **modules** of GPs. Once new data arrives, one fits a *meta-GP* using the module, but **without revisiting** any sample

Modular Gaussian Processes

For each subset of data, we adopt the *sparse variational* GP approach (Titsias, 2009)

What is a GP module?



What is a meta-GP?

Modular Gaussian Processes

For each subset of data, we adopt the *sparse variational* GP approach (Titsias, 2009)

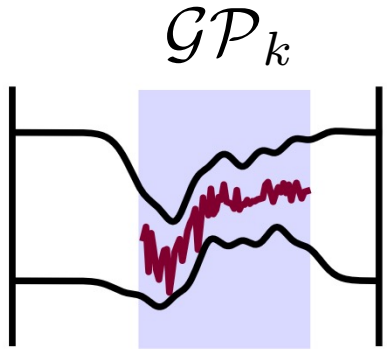
What is a GP module?

The parameters, hyperparameters and *extra* variables of the model, for a *kth module:*

$$\mathcal{M}_k = \{\boldsymbol{\phi}_k, \boldsymbol{\psi}_k, \boldsymbol{Z}_k\}$$

$\boldsymbol{\phi}_k$ _ variational parameters
$\boldsymbol{\psi}_k$ _ *kernel* hyperparameters
$\boldsymbol{Z}_k$ _ inducing points

What is a meta-GP?



$$\mathcal{M}_k = \{\boldsymbol{\phi}_k, \boldsymbol{\psi}_k, \boldsymbol{Z}_k\}$$

We use $*$ as indicator of the *new model*

Modular Gaussian Processes

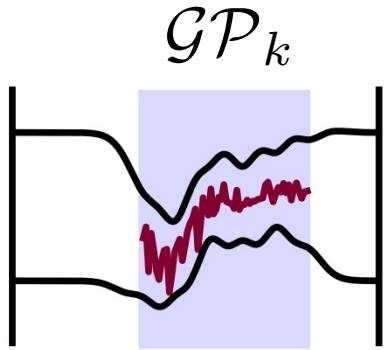
For each subset of data, we adopt the *sparse variational* GP approach (Titsias, 2009)

What is a GP module?

The parameters, hyperparameters and *extra* variables of the model, for a *kth module:*

$$\mathcal{M}_k = \{\phi_k, \psi_k, Z_k\}$$

ϕ_k _ variational parameters
ψ_k _ *kernel* hyperparameters
Z_k _ inducing points



$$\mathcal{M}_k = \{\phi_k, \psi_k, Z_k\}$$

What is a meta-GP?

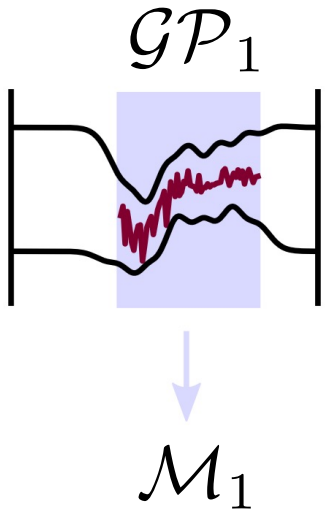
Also a *sparse variational* GP, learned from modules instead of data

$$\mathcal{M}_* = \{\phi_*, \psi_*, Z_*\}$$

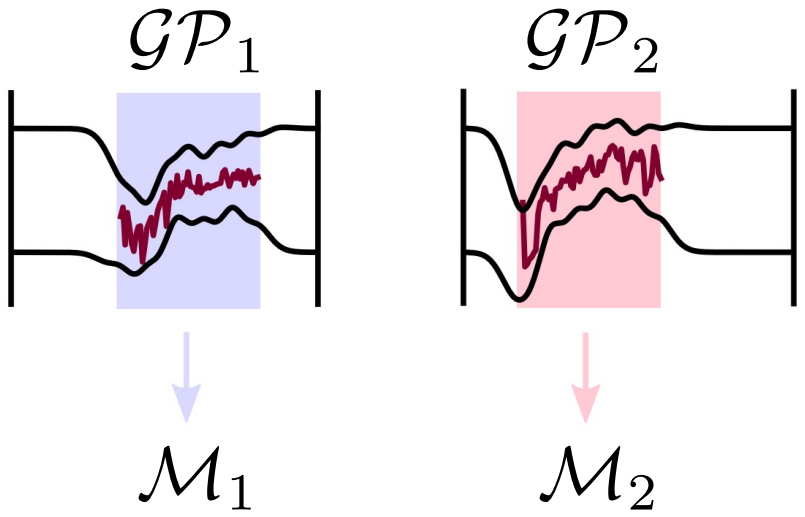
ϕ_* _ *new* variational parameters
ψ_* _ *new kernel* hyperparameters
Z_* _ *new* inducing points

We use $*$ as indicator of the *new model*

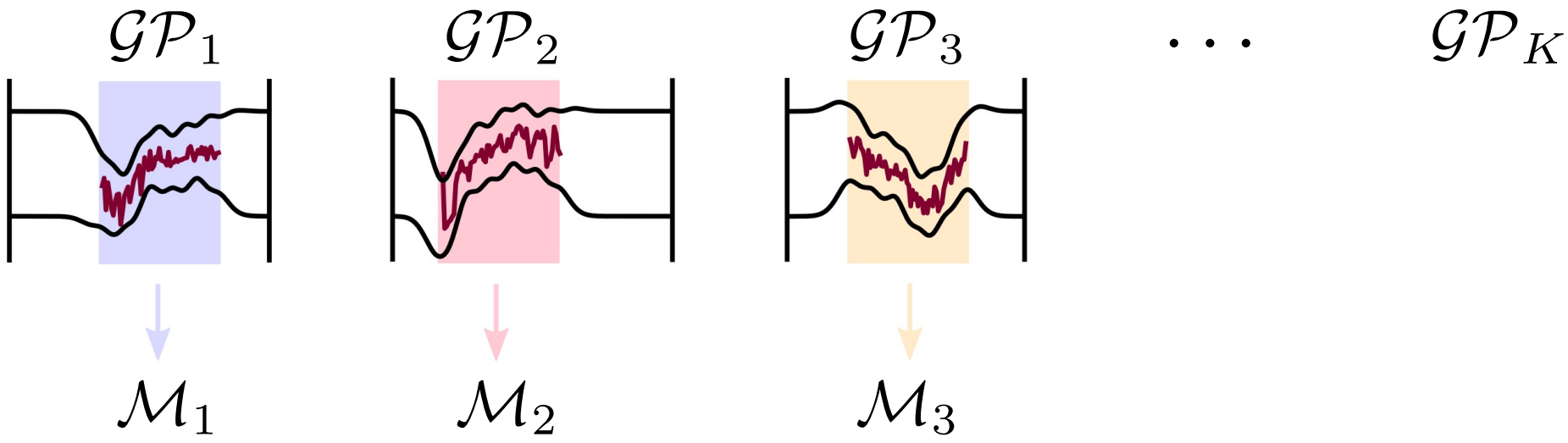
Dictionary of modules



Dictionary of modules



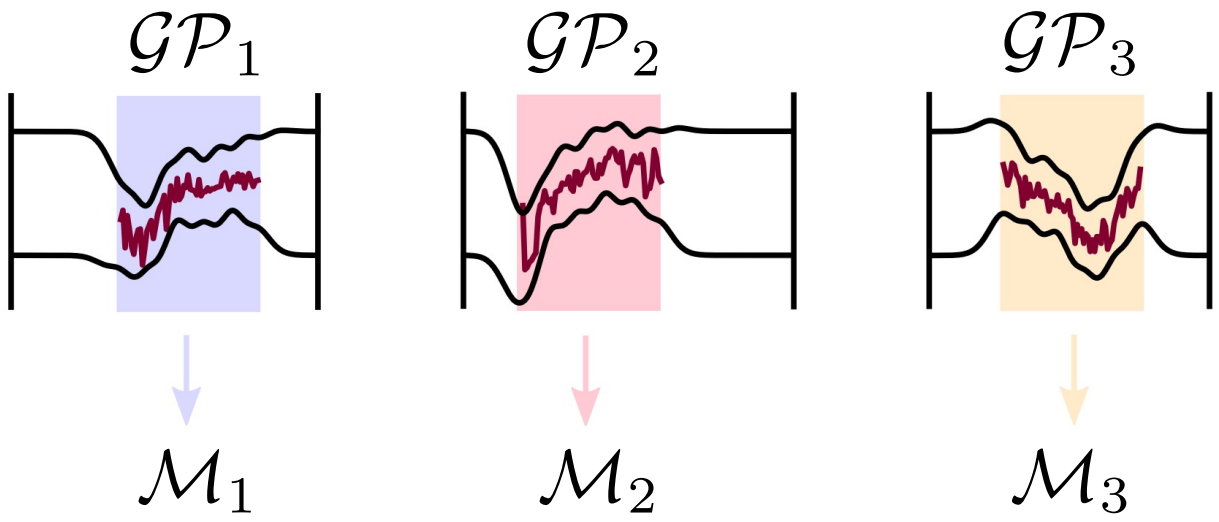
Dictionary of modules



$$\texttt{modules} = \{\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_K\}$$

The principal goal is to obtain a *dictionary*, containing the already fitted GP modules, for their later use.

Dictionary of modules



Each *k*th module is trained from a subset $\mathcal{D}_k$ of the input-output dataset

$$\mathcal{D} = \{\boldsymbol{x}_n, y_n\}_{n=1}^{N}$$

$$\texttt{modules} = \{\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_K\}$$

The principal goal is to obtain a *dictionary*, containing the already fitted GP modules, for their later use.

Global evidence objective

Ideally, to obtain a global inference solution given all GP modules, the resulting meta-GP posterior density *should be valid for all data* modules $\{\mathcal{D}_k\}_{k=1}^{K}$

$$q(f) \approx p(f|\mathcal{D})$$

Module-driven lower bound

Global evidence objective

Ideally, to obtain a global inference solution given all GP modules, the resulting meta-GP posterior density *should be valid for all data* modules $\{\mathcal{D}_k\}_{k=1}^{K}$

$$q(f) \approx p(f|\mathcal{D})$$

global log-marginal likelihood

$$\underbrace{\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_K}$$

$$\log p(\boldsymbol{y}) = \log p(\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_K)$$

Module-driven lower bound

Global evidence objective

Ideally, to obtain a global inference solution given all GP modules, the resulting meta-GP posterior density *should be valid for all data* modules $\{\mathcal{D}_k\}_{k=1}^{K}$

$$q(f) \approx p(f|\mathcal{D})$$

global log-marginal likelihood

$$\underbrace{\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_K}$$

$$\log p(\boldsymbol{y}) = \log p(\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_K)$$

Module-driven lower bound

Augment-and-reduce strategy (Ruiz et al., 2018):

1) *Augment* the GP model to be large dimensional (more function evaluations)
2) *Reduce* the bound to values of interest using properties of Gaussian marginals

Global evidence objective

Ideally, to obtain a global inference solution given all GP modules, the resulting meta-GP posterior density *should be valid for all data* modules $\{\mathcal{D}_k\}_{k=1}^K$

$$q(f) \approx p(f|\mathcal{D})$$

global log-marginal likelihood

$$\underbrace{\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_K}$$

$$\log p(\boldsymbol{y}) = \log p(\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_K)$$

Module-driven lower bound

Augment-and-reduce strategy (Ruiz et al., 2018):

1) *Augment* the GP model to be large dimensional (more function evaluations)
2) *Reduce* the bound to values of interest using properties of Gaussian marginals

f_+ contains all the function values taken by. $f(\cdot)$ at both $\{\boldsymbol{x}_n\}_{n=1}^N$ and $\{\boldsymbol{Z}_k\}_{k=1}^K$

Global evidence objective

Ideally, to obtain a global inference solution given all GP modules, the resulting meta-GP posterior density *should be valid for all data* modules $\{\mathcal{D}_k\}_{k=1}^K$

$$q(f) \approx p(f|\mathcal{D})$$

global log-marginal likelihood

$$\underbrace{\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_K}$$

$$\log p(\boldsymbol{y}) = \log p(\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_K)$$

Module-driven lower bound

$$\log p(\boldsymbol{y}) = \log p(\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_K) = \log \int p(\boldsymbol{y}, f_+) df_+$$

Global evidence objective

Ideally, to obtain a global inference solution given all GP modules, the resulting meta-GP posterior density *should be valid for all data* modules $\{\mathcal{D}_k\}_{k=1}^{K}$

$$q(f) \approx p(f|\mathcal{D})$$

global log-marginal likelihood

$$\underbrace{\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_K}$$

$$\log p(\boldsymbol{y}) = \log p(\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_K)$$

Module-driven lower bound

$$\log p(\boldsymbol{y}) = \log \iint q(\boldsymbol{u}_*) p(f_{+\neq \boldsymbol{u}_*}|\boldsymbol{u}_*) p(\boldsymbol{y}|f_+) \frac{p(\boldsymbol{u}_*)}{q(\boldsymbol{u}_*)} df_{+\neq \boldsymbol{u}_*} d\boldsymbol{u}_*$$

Global evidence objective

Ideally, to obtain a global inference solution given all GP modules, the resulting meta-GP posterior density *should be valid for all data* modules $\{\mathcal{D}_k\}_{k=1}^K$

$$q(f) \approx p(f|\mathcal{D})$$

global log-marginal likelihood

$$\underbrace{\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_K}$$

$$\log p(\boldsymbol{y}) = \log p(\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_K)$$

Module-driven lower bound

$$\log p(\boldsymbol{y}) = \log \iint q(\boldsymbol{u}_*) p(f_{+\neq \boldsymbol{u}_*}|\boldsymbol{u}_*) p(\boldsymbol{y}|f_+) \frac{p(\boldsymbol{u}_*)}{q(\boldsymbol{u}_*)} df_{+\neq \boldsymbol{u}_*} d\boldsymbol{u}_*$$

$$\geq \mathbb{E}_{q(\boldsymbol{u}_*)} \left[\mathbb{E}_{p(f_{+\neq \boldsymbol{u}_*}|\boldsymbol{u}_*)} [\log p(\boldsymbol{y}|f_+)] + \log \frac{p(\boldsymbol{u}_*)}{q(\boldsymbol{u}_*)} \right]$$

Global evidence objective

Ideally, to obtain a global inference solution given all GP modules, the resulting meta-GP posterior density *should be valid for all data* modules $\{\mathcal{D}_k\}_{k=1}^K$

$$q(f) \approx p(f|\mathcal{D})$$

global log-marginal likelihood

$$\underbrace{\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_K}$$

$$\log p(\boldsymbol{y}) = \log p(\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_K)$$

Module-driven lower bound

$$\log p(\boldsymbol{y}) = \log \iint q(\boldsymbol{u}_*) p(f_{+\neq \boldsymbol{u}_*}|\boldsymbol{u}_*) p(\boldsymbol{y}|f_+) \frac{p(\boldsymbol{u}_*)}{q(\boldsymbol{u}_*)} df_{+\neq \boldsymbol{u}_*} d\boldsymbol{u}_*$$

$$\geq \mathbb{E}_{q(\boldsymbol{u}_*)} \left[\mathbb{E}_{p(f_{+\neq \boldsymbol{u}_*}|\boldsymbol{u}_*)} [\log p(\boldsymbol{y}|f_+)] + \log \frac{p(\boldsymbol{u}_*)}{q(\boldsymbol{u}_*)} \right]$$

log-likelihood factorises across subsets

How to *avoid the re-evaluation* of likelihood terms?

$$\log p(\boldsymbol{y}|f_+) = \sum_{k=1}^{K} \log p(\boldsymbol{y}_k|f_+)$$

$$\texttt{modules} = \{\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_K\}$$

Likelihood reconstruction

How to *avoid the re-evaluation* of likelihood terms?

$$\log p(\boldsymbol{y}|f_+) = \sum_{k=1}^{K} \log p(\boldsymbol{y}_k|f_+) \quad \longleftarrow \quad \texttt{modules} = \{\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_K\}$$

Likelihood reconstruction

Using Bayes theorem for approximating GP likelihood terms (Bui et al. 2017, Matthews et al. 2016)

$$p(\boldsymbol{y}_k|f_+) \approx Z_k \frac{q_k(f_+)}{p_k(f_+)} = Z_k \frac{\cancel{p(f_{+\neq \boldsymbol{u}_k}|\boldsymbol{u}_k)} q_k(\boldsymbol{u}_k)}{\cancel{p(f_{+\neq \boldsymbol{u}_k}|\boldsymbol{u}_k)} p_k(\boldsymbol{u}_k)} = Z_k \frac{q_k(\boldsymbol{u}_k)}{p_k(\boldsymbol{u}_k)}$$

Module-driven lower bounds

$$\log p(\boldsymbol{y}) \geq \sum_{k=1}^{K} \log Z_k + \sum_{k=1}^{K} \mathbb{E}_{q_{\mathcal{C}}(\boldsymbol{u}_k)}[\log q_k(\boldsymbol{u}_k) - \log p_k(\boldsymbol{u}_k)] - \mathrm{KL}[q(\boldsymbol{u}_*)||p(\boldsymbol{u}_*)]$$

lower bound on the global log-marginal likelihood were *the evaluation of data is omitted*

Module-driven lower bounds

$$\log p(\boldsymbol{y}) \geq \sum_{k=1}^{K} \log Z_k + \sum_{k=1}^{K} \mathbb{E}_{q_{\mathcal{C}}(\boldsymbol{u}_k)}[\log q_k(\boldsymbol{u}_k) - \log p_k(\boldsymbol{u}_k)] - \mathrm{KL}[q(\boldsymbol{u}_*)||p(\boldsymbol{u}_*)]$$

$$\geq \sum_{k=1}^{K} \mathcal{L}_k^* + \sum_{k=1}^{K} \mathbb{E}_{q_{\mathcal{C}}(\boldsymbol{u}_k)}[\log q_k(\boldsymbol{u}_k) - \log p_k(\boldsymbol{u}_k)] - \mathrm{KL}[q(\boldsymbol{u}_*)||p(\boldsymbol{u}_*)]$$

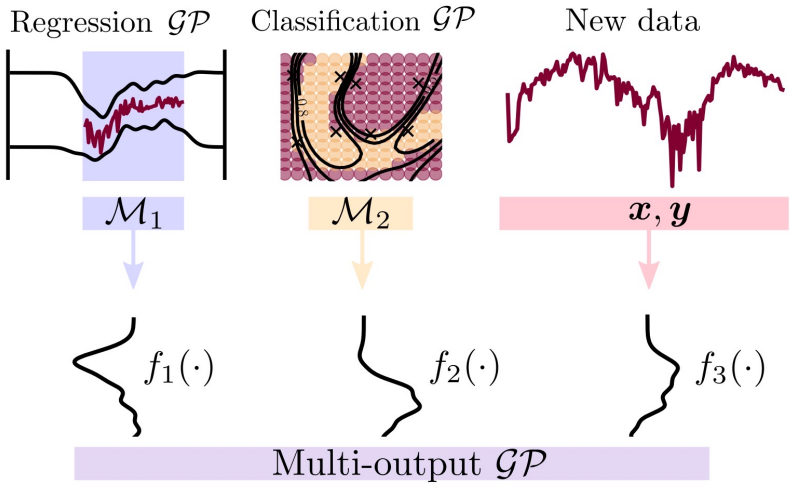
for single-output class tasks (both regression and classification)

the maximisation is w.r.t. $\mathcal{M}_* = \{\boldsymbol{\phi}_*, \boldsymbol{\psi}_*, \boldsymbol{Z}_*\}$

Transfer learning with **multi-output modules**

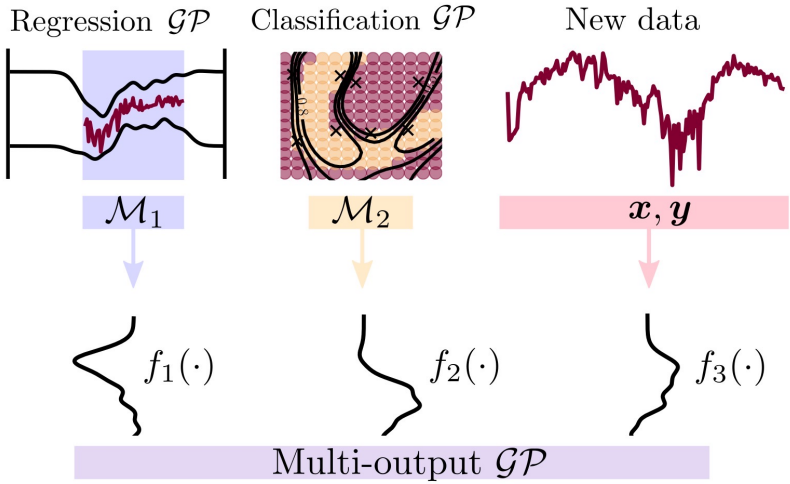
The assumption of a single GP function parameterising multiple modules might be too strong. We can relax it to *assume that modules are correlated* and *learn a multi-output GP* instead without revisiting data.

$$f_k(\boldsymbol{x}) = \sum_{q=1}^{Q} \sum_{i=1}^{R_q} a_{k,q}^i v_q^i(\boldsymbol{x}) \qquad v_q(\cdot) \sim \mathcal{GP}(0, k_q(\cdot,\cdot))$$



Transfer learning with multi-output modules

The assumption of a single GP function parameterising multiple modules might be too strong. We can relax it to *assume that modules are correlated* and *learn a multi-output GP* instead without revisiting data.



$$f_k(\boldsymbol{x}) = \sum_{q=1}^{Q} \sum_{i=1}^{R_q} a_{k,q}^i v_q^i(\boldsymbol{x}) \qquad v_q(\cdot) \sim \mathcal{GP}(0, k_q(\cdot,\cdot))$$

$$\mathcal{L}_{\mathcal{M}}^{\text{MO}} = \sum_{k=1}^{K} \mathcal{L}_k^* + \sum_{k=1}^{K} \mathbb{E}_{q_C(\boldsymbol{u}_k)}[\log q_k(\boldsymbol{u}_k) - \log p(\boldsymbol{u}_k)] - \sum_{q=1}^{Q} \text{KL}[q(\boldsymbol{v}_{*q}) || p(\boldsymbol{v}_{*q})]$$

Q variational densities $q(\boldsymbol{v}_{*q}) = \mathcal{N}(\boldsymbol{v}_{*q} | \boldsymbol{\mu}_{*q}, \boldsymbol{S}_{*q})$

Experiments / regression

Performance on regression synthetic data, results show the *robustness* of the framework to accept outlier modules

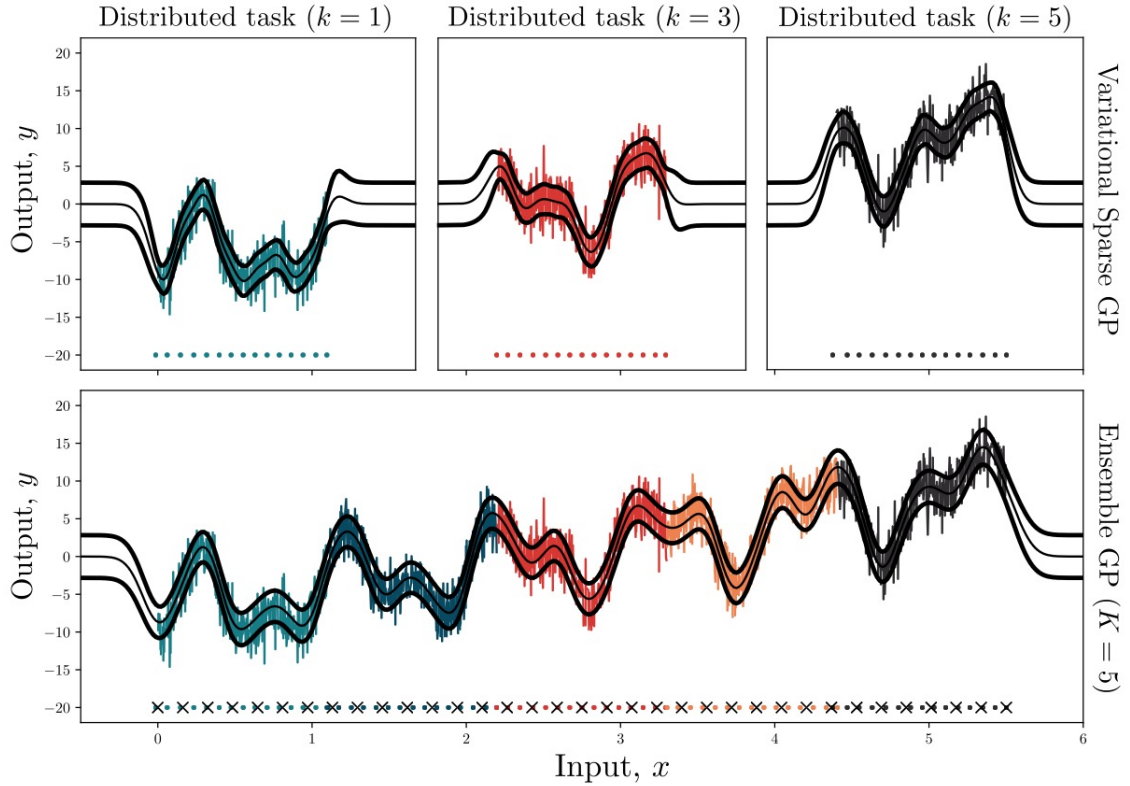


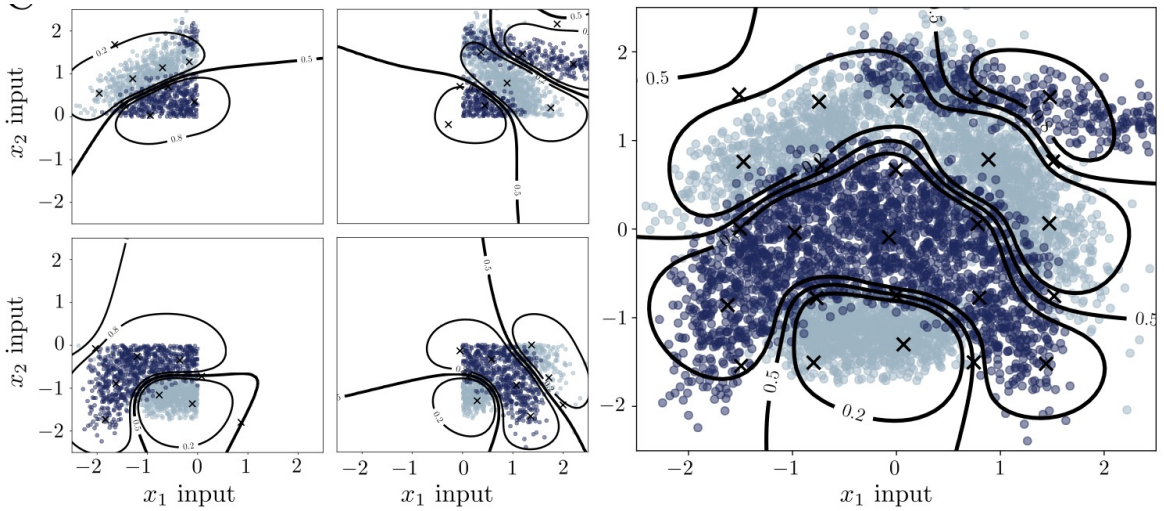
Table 2: Comparative error metrics for distributed GP models.

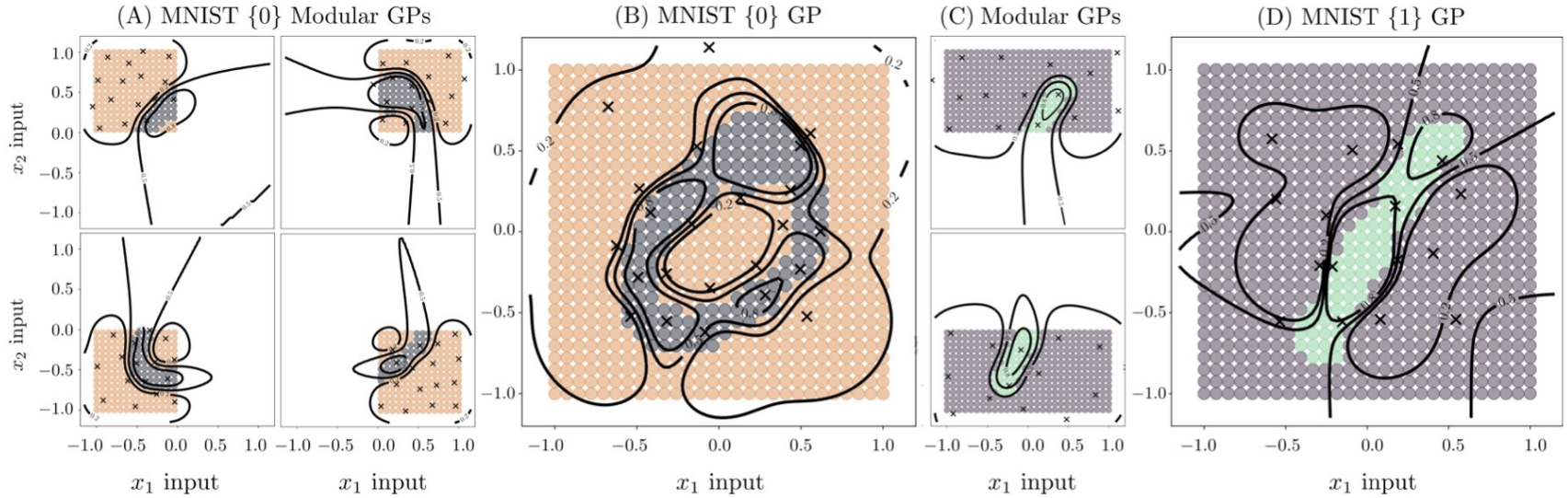
DATA SIZE →	10K			100K			1M		
MODEL	NLPD	RMSE	MAE	NLPD	RMSE	MAE	NLPD	RMSE	MAE
BCM	2.99 ± 0.94	11.94 ± 18.89	2.05 ± 1.31	3.51 ± 0.73	2.33 ± 0.96	1.34 ± 1.03	NA	NA	NA
PoE	2.79 ± 0.16	2.32 ± 0.22	1.86 ± 0.22	2.82 ± 0.67	2.19 ± 0.91	1.71 ± 0.84	2.91 ± 0.63	1.98 ± 0.61	1.32 ± 0.05
GPoE	2.79 ± 0.56	2.43 ± 0.52	1.96 ± 0.48	$\mathbf{2.73 \pm 0.72}$	2.19 ± 0.91	1.71 ± 0.84	2.72 ± 0.52	1.98 ± 0.61	$\mathbf{1.32 \pm 0.05}$
RBCM	2.96 ± 0.51	2.49 ± 0.51	2.02 ± 0.46	3.03 ± 0.86	2.51 ± 1.12	1.99 ± 1.04	$\mathbf{2.56 \pm 0.06}$	$\mathbf{1.82 \pm 0.02}$	1.37 ± 0.03
ModularGP	$\mathbf{2.71 \pm 0.11}$	$\mathbf{1.56 \pm 0.04}$	$\mathbf{0.97 \pm 0.05}$	2.89 ± 0.07	$\mathbf{1.73 \pm 0.01}$	$\mathbf{1.23 \pm 0.02}$	2.87 ± 0.09	1.87 ± 0.07	1.34 ± 0.09

Acronyms: BCM (Tresp, 2000), PoE (Ng and Deisenroth, 2014), GPoE (Cao and Fleet, 2014) and RBCM (Deisenroth and Ng, 2015).

Experiments / classification

Results on the banana dataset and pixel-wise MNIST. *Meta-GP prediction is accurate* as in the standard sparse variational GP classification setup.





Experiments / multi-output

Table 3: Comparative metrics of modular multi-output GPs for US-FLIGHT dataset.

PARTITION →	DAYS			MONTHS		
MODELS	NLPD	MAE	RMSE	NLPD	MAE	RMSE
MODULES (†)	2.36 ± 0.18	1.48 ± 0.26	2.31 ± 0.24	2.03 ± 0.02	1.53 ± 0.06	1.83 ± 0.03
MODULARGP ($Q = 2$)	2.49 ± 0.37	1.49 ± 0.26	2.31 ± 0.24	2.51 ± 0.34	$\mathbf{1.56 \pm 0.14}$	2.37 ± 0.13
MODULARGP ($Q = 3$)	2.38 ± 0.23	$\mathbf{1.49 \pm 0.25}$	2.31 ± 0.25	2.38 ± 0.13	1.57 ± 0.13	2.38 ± 0.11
MODULARGP ($Q = 4$)	$\mathbf{2.36 \pm 0.15}$	1.49 ± 0.26	2.31 ± 0.24	2.39 ± 0.03	1.57 ± 0.14	2.37 ± 0.12
MOGP ($Q = 2$)	2.49 ± 0.38	1.51 ± 0.25	2.31 ± 0.25	2.58 ± 0.42	1.61 ± 0.12	2.23 ± 0.14
MOGP ($Q = 3$)	2.39 ± 0.25	1.50 ± 0.26	2.31 ± 0.26	2.46 ± 0.38	1.61 ± 0.11	2.18 ± 0.12
MOGP ($Q = 4$)	2.37 ± 0.17	1.51 ± 0.26	2.31 ± 0.25	$\mathbf{2.34 \pm 0.28}$	1.63 ± 0.11	$\mathbf{2.14 \pm 0.13}$

PARTITION →	DAYS			MONTHS		
AVG. DIFFERENCE PER OUTPUT →	ΔNLPD	ΔMAE	ΔRMSE	ΔNLPD	ΔMAE	ΔRMSE
MODULARGP ($Q = 2$) vs. Modules	-3.91%	$\mathbf{-0.64\%}$	-0.17%	-17.69%	-1.41%	-22.73%
MODULARGP ($Q = 3$) vs. Modules	-0.51%	-0.99%	-0.16%	-14.19%	$\mathbf{-1.41\%}$	-22.81%
MODULARGP ($Q = 4$) vs. Modules	$\mathbf{+0.13\%}$	$-1,31\%$	-0.19%	-14.93%	-1.87%	-22.34%
MODULARGP ($Q = 2$) vs. MOGP	-3.71%	-0.75%	$\mathbf{-0.15\%}$	-19.49%	-4.21%	-18.04%
MODULARGP ($Q = 3$) vs. MOGP	-0.91%	-1.25%	-0.33%	-13.96%	-3.32%	-15.66%
MODULARGP ($Q = 4$) vs. MOGP	-0.11%	-1.59%	-0.27%	$\mathbf{-11.54\%}$	-4.92%	$\mathbf{-13.84\%}$

(†) Modules as the metric of reference.

Conclusions

- A new framework for *building meta-GP* models from modules
- Principled *formulation* for regression, classification and multi-output tasks
- Avoidance of *data revisiting*, extra computational cost and *robust* to module variability

Future directions

- Extension with *convolutional kernels* (Van der Wilk et al., 2017) for large-scale image processing
- Consider functional regularisation (Titsias et al., 2020; Moreno-Muñoz et al., 2019) for *continual learning* applications

Conclusions

- A new framework for *building meta-GP* models from modules
- Principled *formulation* for regression, classification and multi-output tasks
- Avoidance of *data revisiting*, extra computational cost and *robust* to module variability

Future directions

- Extension with *convolutional kernels* (Van der Wilk et al., 2017) for large-scale image processing
- Consider functional regularisation (Titsias et al., 2020; Moreno-Muñoz et al., 2019) for *continual learning* applications

Main **references**

- M. K. Titsias. Variational learning of inducing variables in sparse Gaussian processes. In *Artificial Intelligence and Statistics (AISTATS), pages 567-574, 2009*
- F. J. Ruiz, M. K. Titsias, A. B. Dieng and D. M. Blei, Augment and reduce: Stochastic inference for large scale categorical distributions. In *International Conference on Machine Learning (ICML),* 2018
- T. D. Bui, C. V. Nguyen and R. E. Turner. Streaming sparse Gaussian process approximations. In *Advances in Neural Information Processing Systems (NIPS),* pages 3299-3307, 2017
- A. G. d. G. Matthews, J. Hensman, R. E. Turner and Z. Ghahramani. On sparse variational methods and the Kullback-Leibler divergence between stochastic processes. In *Artificial Intelligence and Statistics (AISTATS),* pages 231-239, 2016
- M. Van der Wilk, C. E. Rassmussen and J. Hensman. Convolutional Gaussian processes. In *Advances in Neural Information Processing Systems (NIPS),* pages 2849-2858, 2017
- M. K. Titsias, J. Schwarz, A. G. de G. Matthews, R. Pascanu and Y. W. Teh. Functional regularisation for continual learning with Gaussian processes. In *International Conference on Learning Representations (ICLR),* 2020.
- M. Deisenroth and J. W. Ng. Distributed Gaussian processes. In *International Conference on Machine Learning (ICML),* pages 1481-1490, 2015.
- Y. Gal, M. Van der Wilk and C. E. Rassmussen. Distributed variational inference in sparse Gaussian process regression and latent variable models. In *Advances in Neural Information Processing Systems (NIPS),* pages 3257-3265, 2014