Prediction of physical fields under linear constraints

ETICS Annual Research Schools, October, 2025

NASSOURADINE MAHAMAT HAMDAN[1,2]

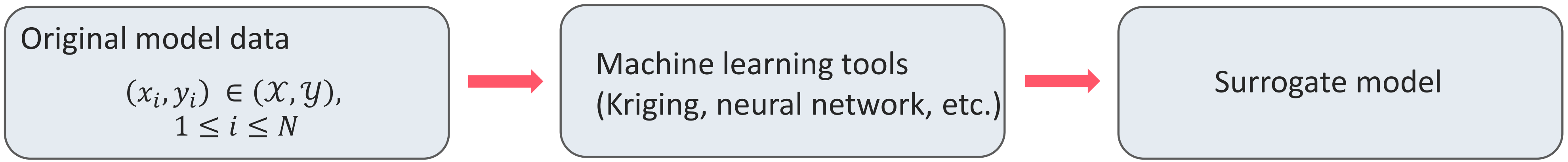
Supervisors : **Clément Gauchy**[1], **Pierre-Emmanuel Angeli**[1]
PhD Director : **Sébastien Da Veiga**[2]

mahamat-hamdan.nassouradine@cea.fr
CEA Saclay[1] ENSAI CREST[2]

Context

- Numerical simulations are widely used in engineering, especially for **uncertainty quantification** task
- Their repeated use becomes **impractical** when codes are **computationally expensive**

➡ **Metamodeling** provides a solution by approximating the full code from a limited set of high-fidelity simulations



Surrogate modeling for physical fields

- Metamodeling process

Original model data

$$(x_i, y_i) \in (\mathcal{X}, \mathcal{Y}),$$
$$1 \leq i \leq N$$

→ Machine learning tools → Surrogate model

- Input space: $\mathcal{X} \subset \mathbb{R}^D$
- Output space: $\mathcal{Y} \subset \mathbb{R}^P$

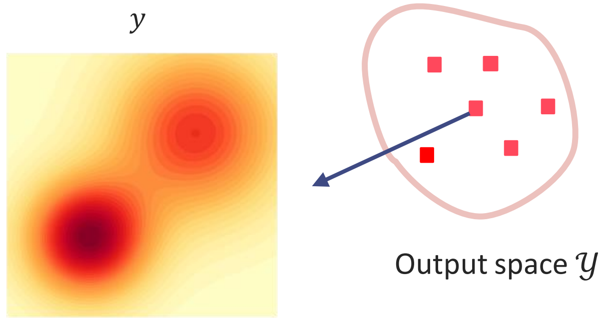
Surrogate modeling for physical fields

- Metamodeling process

Original model data $(x_i, y_i) \in (\mathcal{X}, \mathcal{Y}),$ $1 \leq i \leq N$ → Machine learning tools → Surrogate model

- Input space: $\mathcal{X} \subset \mathbb{R}^D$
- Output space: $\mathcal{Y} \subset \mathbb{R}^P$

What if P is very large ? $P \approx 10^5$ (collection of physical field outputs)



Surrogate modeling for physical fields

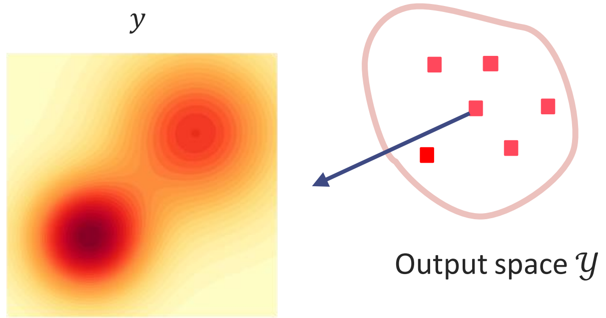
- Metamodeling process

Original model data $(x_i, y_i) \in (\mathcal{X}, \mathcal{Y}),$ $1 \leq i \leq N$ → Machine learning tools → Surrogate model

- Input space: $\mathcal{X} \subset \mathbb{R}^D$
- Output space: $\mathcal{Y} \subset \mathbb{R}^P$

What if P is very large ? $P \approx 10^5$ (collection of physical field outputs)

Solution → Dimensionality reduction techniques can be used to handle this problem before regression (PCA, KPCA, Isomap)[1]



[1] Higdon, D., Gattiker, J., Williams, B., & Rightley, M. (2008).

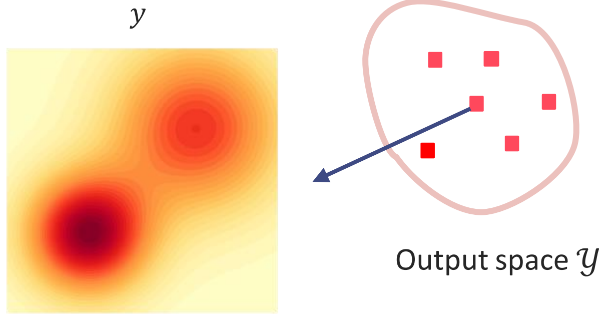
Surrogate modeling for physical fields

- Metamodeling process

Original model data $(x_i, y_i) \in (\mathcal{X}, \mathcal{Y}),\ 1 \leq i \leq N$ → Machine learning tools → Surrogate model

- Input space: $\mathcal{X} \subset \mathbb{R}^D$
- Output space: $\mathcal{Y} \subset \mathbb{R}^P$

What if P is very large ? $P \approx 10^5$ (collection of physical field outputs)



Solution → Dimensionality reduction techniques can be used to handle this problem before regression (PCA, KPCA, Isomap)[1]

- Now, what if the high dimensional fields are governed by physical constraints?

✕ Machine learning tools fail to guarantee physical constraints in prediction implicitly

[1] Higdon, D., Gattiker, J., Williams, B., & Rightley, M. (2008).

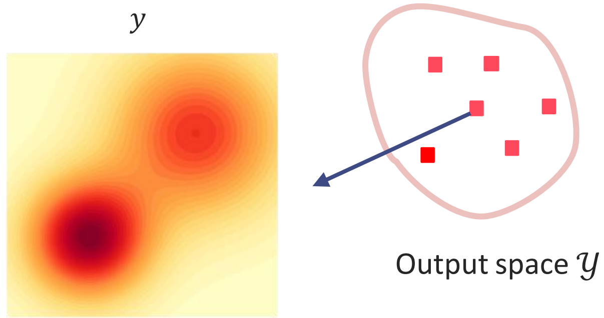
Surrogate modeling for physical fields

- Metamodeling process

Original model data $(x_i, y_i) \in (\mathcal{X}, \mathcal{Y}),\ 1 \leq i \leq N$ → Machine learning tools → Surrogate model

- Input space: $\mathcal{X} \subset \mathbb{R}^D$
- Output space: $\mathcal{Y} \subset \mathbb{R}^P$

What if P is very large ? $P \approx 10^5$ (collection of physical field outputs)



Solution → Dimensionality reduction techniques can be used to handle this problem before regression (PCA, KPCA, Isomap)[1]

- Now, what if the high dimensional fields are governed by physical constraints?

× Machine learning tools fail to guarantee physical constraints in prediction implicitly

Solution → Imposing physical constraints explicitly in the model: Physics informed machine learning!

[1] Higdon, D., Gattiker, J., Williams, B., & Rightley, M. (2008).

Linear constraints from physics

Many physics problems are governed by linear constraints, for example:

- The law of conservation of mass implies linear relationships between the concentrations or proportions of different chemical species in a given solution
- The incompressibility condition in fluid dynamics leads to linear constraints on the velocity field or, equivalently, a zero-trace stress or strain tensor

Linear constraints from physics

Many physics problems are governed by linear constraints, for example:

- ❑ The law of conservation of mass implies linear relationships between the concentrations or proportions of different chemical species in a given solution
- ❑ The incompressibility condition in fluid dynamics leads to linear constraints on the velocity field or, equivalently, a zero-trace stress or strain tensor

In this work, we are considering:

- An observed Q-dimensional vector field output on a fixed mesh of size S: $y_i = (y_i^1, \dots, y_i^Q) \in \mathbb{R}^{SQ}$,
- The vector output regression function to estimate: $f = (f^1, \dots, f^Q)$: $\mathcal{X} \longrightarrow \mathbb{R}^{SQ}$
- The class of linear constraints defined through the sum of the Q scalar fields $f^j(.)$:

$$\mathcal{F}[f(x)] = \sum_{j=1}^{Q} \alpha^j(x) f^j(x) = c(x) \qquad (1)$$

where $\alpha^j(.)$ are input-dependent scalar weights, $f^j(.)$ and $c(.)$ are scalar fields. We assume α^j and c **are known**.

Problem statement

Simultaneous modeling of linearly constrained fields

- $\mathcal{X} \subset \mathbb{R}^D$: Input space (PDEs parameters)
- $\mathcal{Y} \subset \mathbb{R}^{S^Q}$: Output space (collection of spatial fields)
- $\left(x_i, y_i = (y_i^1, \dots, y_i^Q)\right)_{\{i=1,\dots,N\}} \in \mathcal{X} \times \mathcal{Y}$: iid samples from Q scalar fields in fixed discretization (mesh, time discretization, etc.)
- Note $P = Q * S$: $y \in \mathbb{R}^P$

Problem statement

Simultaneous modeling of linearly constrained fields

- $\mathcal{X} \subset \mathbb{R}^D$: Input space (PDEs parameters)
- $\mathcal{Y} \subset \mathbb{R}^{S^Q}$: Output space (collection of spatial fields)
- $\left(x_i, y_i = (y_i^1, \ldots, y_i^Q)\right)_{\{i=1,\ldots,N\}} \in \mathcal{X} \times \mathcal{Y}$: iid samples from Q scalar fields in fixed discretization (mesh, time discretization, etc.)
- Note $P = Q * S$: $y \in \mathbb{R}^P$

We are considering the multi-output regression task of finding $f = (f_1, \ldots, f_Q): \mathcal{X} \longrightarrow \mathcal{Y}$, such that:

$$\begin{cases} y = f(x), \qquad y \text{ is a high dimensional tensor, i.e. } P \approx 10^5 \\ \mathcal{F}[f(x)] = \sum_{j=1}^{Q} \alpha^j(x) f^j(x) = c(x) \end{cases}$$

Problem statement

Simultaneous modeling of linearly constrained fields

- $\mathcal{X} \subset \mathbb{R}^D$: Input space (PDEs parameters)
- $\mathcal{Y} \subset \mathbb{R}^{S^Q}$: Output space (collection of spatial fields)
- $\left(x_i, y_i = (y_i^1, \dots, y_i^Q)\right)_{\{i=1,\dots,N\}} \in \mathcal{X} \times \mathcal{Y}$: iid samples from Q scalar fields in fixed discretization (mesh, time discretization, etc.)
- Note $P = Q * S$: $y \in \mathbb{R}^P$

We are considering the multi-output regression task of finding $f = (f_1, \dots, f_Q)$: $\mathcal{X} \longrightarrow \mathcal{Y}$, such that:

$$\begin{cases} y = f(x), \quad y \text{ is a high dimensional tensor, i.e. } P \approx 10^5 \\ \mathcal{F}[f(x)] = \sum_{j=1}^{Q} \alpha^j(x) f^j(x) = c(x) \end{cases}$$

Main mathematical tools:

- **Constrained Gaussian processes:**
  ⟹ Can deal with constrained problems
  × Not suited for high dimensional output
- **Multi-output GP[2] :**
  ⟹ Well suited for multitask regression
  × Complexity $\mathcal{O}(N^3 P^3)$ makes optimization impracticable when output dimension is large
- MOGP with **dimensionality reduction:**
  ⟹ handles high-dimensional outputs and is well known in spatial fields modeling
  × Does not respect linear constraints

[2] Alvarez , M. A., Rosasco, L., & Lawrence, N. D. (2012)

Problem statement

Simultaneous modeling of linearly constrained fields

- $\mathcal{X} \subset \mathbb{R}^D$: Input space (PDEs parameters)
- $\mathcal{Y} \subset \mathbb{R}^{S^Q}$: Output space (collection of spatial fields)
- $\left(x_i, y_i = (y_i^1, \dots, y_i^Q)\right)_{\{i=1,\dots,N\}} \in \mathcal{X} \times \mathcal{Y}$: iid samples from Q scalar fields in fixed discretization (mesh, time discretization, etc.)
- Note $P = Q * S$: $y \in \mathbb{R}^P$

We are considering the multi-output regression task of finding $f = (f_1, \dots, f_Q)$: $\mathcal{X} \longrightarrow \mathcal{Y}$, such that:

$$\begin{cases} y = f(x), \quad y \text{ is a high dimensional tensor, i.e. } P \approx 10^5 \\ \mathcal{F}[f(x)] = \sum_{j=1}^{Q} \alpha^j(x) f^j(x) = c(x) \end{cases}$$

Main mathematical tools:

- **Constrained Gaussian processes:**
  $\Longrightarrow$ Can deal with constrained problems
  × Not suited for high dimensional output
- **Multi-output GP[2] :**
  $\Longrightarrow$ Well suited for multitask regression
  × Complexity $\mathcal{O}(N^3P^3)$ makes optimization impracticable when output dimension is large
- MOGP with **dimensionality reduction:**
  $\Longrightarrow$ handles high-dimensional outputs and is well known in spatial fields modeling
  × Does not respect linear constraints

How can we combine constrained Multi-output GP regression with dimensionality reduction ?

[2] Alvarez , M. A., Rosasco, L., & Lawrence, N. D. (2012)

Constrained Gaussian Process regression

Given a linear constraint on f: $\mathcal{L}[f] = 0$, $\mathcal{L}$ linear operator

Parametrization approach[4] :

- Suppose that: $f(x) = \mathcal{G}_x[g]$
- Then we impose the constraint on the operator $\mathcal{G}_x$:

$$\mathcal{L}[\mathcal{G}_x] = 0$$

- Gaussian processes are closed under linear transformation

$$g \sim \mathcal{GP}(\mu_g(x), k_g(x, x'))$$

$$\mu_f(x) = \mathcal{G}_x[\mu_g], \qquad k_f = \mathcal{G}_x k_g \mathcal{G}_x^T$$

[1] Jidling, C., Wahlström, N., Wills, A., & Schön, T. B. (2017)

Constrained Gaussian Process regression

Given a linear constraint on f: $\mathcal{L}[f] = 0$, $\mathcal{L}$ linear operator

Parametrization approach[3] :

- Suppose that: $f(x) = \mathcal{G}_x[g]$
- Then we impose the constraint on the operator $\mathcal{G}_x$:

$$\mathcal{L}[\mathcal{G}_x] = 0$$

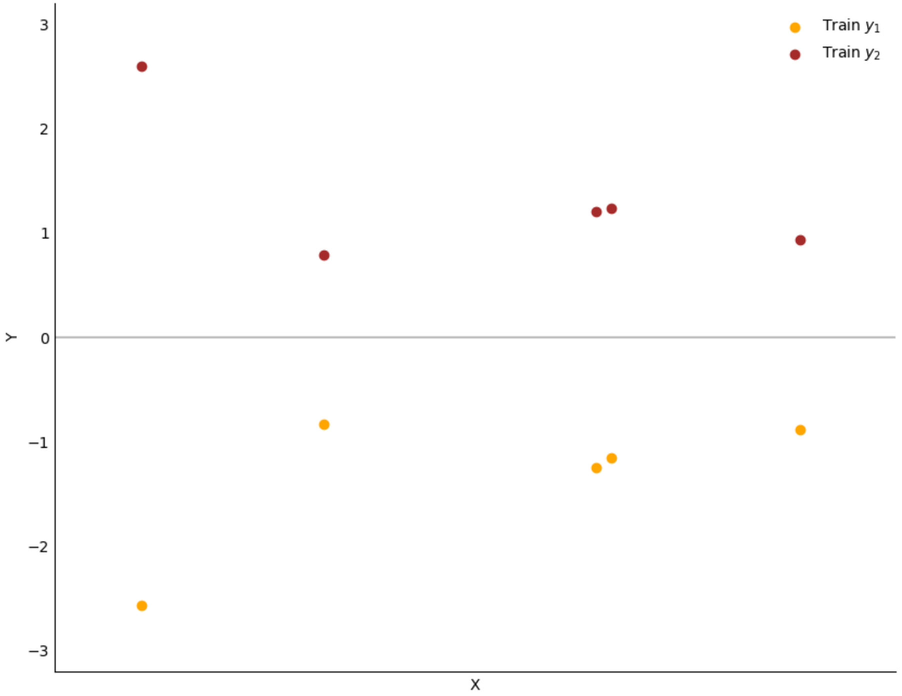
- Gaussian processes are closed under linear transformation

$$g \sim \mathcal{GP}(\mu_g(x), k_g(x, x'))$$

$$\mu_f(x) = \mathcal{G}_x[\mu_g], \qquad k_f = \mathcal{G}_x k_g \mathcal{G}_x^T$$

Illustration: $\mathcal{L}[f] = f_1 + f_2 = 0$

Constrained GP illustration



[3] Jidling, C., Wahlström, N., Wills, A., & Schön, T. B. (2017)

Constrained Gaussian Process regression

Given a linear constraint on f: $\mathcal{L}[f] = 0$, $\mathcal{L}$ linear operator

Parametrization approach[3] :

- Suppose that: $f(x) = \mathcal{G}_x[g]$
- Then we impose the constraint on the operator $\mathcal{G}_x$:

$$\mathcal{L}[\mathcal{G}_x] = 0$$

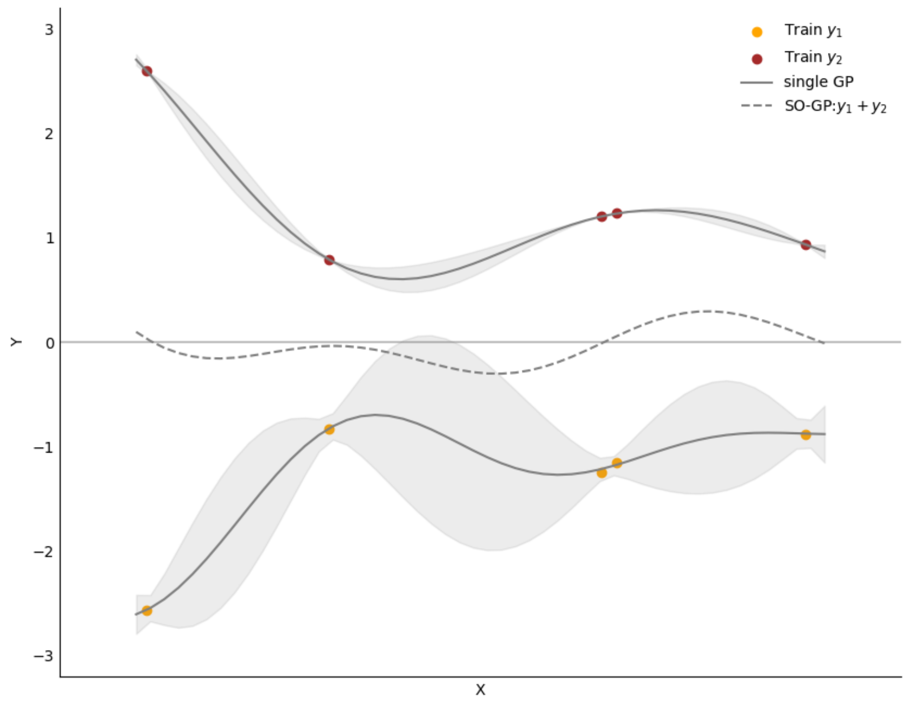
- Gaussian processes are closed under linear transformation

$$g \sim \mathcal{GP}(\mu_g(x), k_g(x, x'))$$

$$\mu_f(x) = \mathcal{G}_x[\mu_g], \qquad k_f = \mathcal{G}_x k_g \mathcal{G}_x^T$$

Illustration: $\mathcal{L}[f] = f_1 + f_2 = 0$

Constrained GP illustration



[3] Jidling, C., Wahlström, N., Wills, A., & Schön, T. B. (2017)

Constrained Gaussian Process regression

Given a linear constraint on f: $\mathcal{L}[f] = 0$, $\mathcal{L}$ linear operator

Parametrization approach[3] :

- Suppose that: $f(x) = \mathcal{G}_x[g]$
- Then we impose the constraint on the operator $\mathcal{G}_x$:

$$\mathcal{L}[\mathcal{G}_x] = 0$$

- Gaussian processes are closed under linear transformation

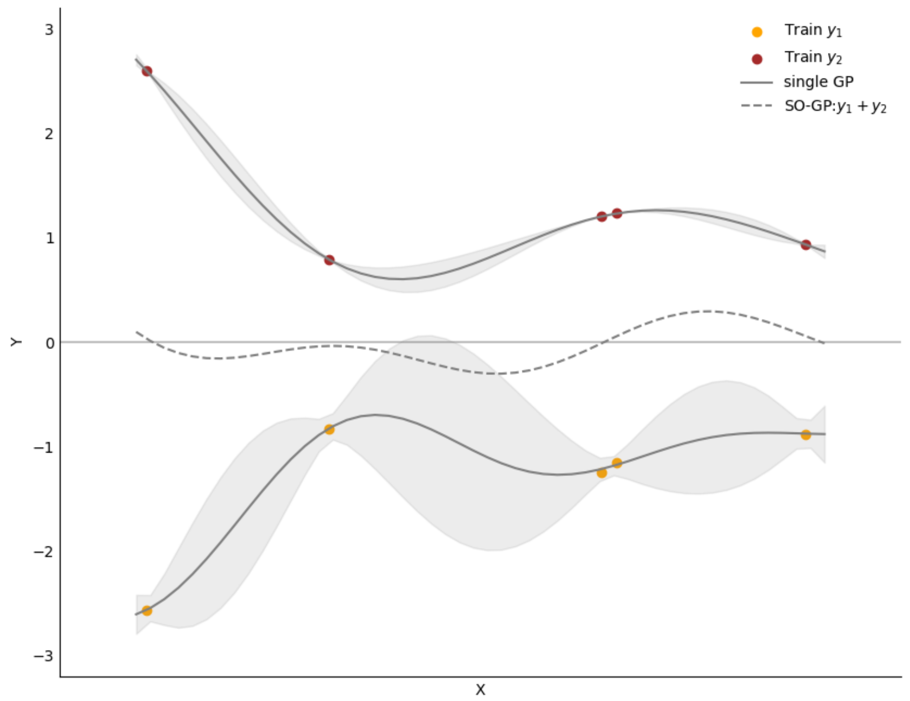
$$g \sim \mathcal{GP}(\mu_g(x), k_g(x, x'))$$

$$\mu_f(x) = \mathcal{G}_x[\mu_g], \qquad k_f = \mathcal{G}_x k_g \mathcal{G}_x^T$$

Illustration: $\mathcal{L}[f] = f_1 + f_2 = 0$

- $\mathcal{G}_x = [1, -1]$
- $\mu_f(x) = [\mu_g, -\mu_g]^T$, $k_f = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} k_g$

Constrained GP illustration



[3] Jidling, C., Wahlström, N., Wills, A., & Schön, T. B. (2017)

Constrained Gaussian Process regression

Given a linear constraint on f: $\mathcal{L}[f] = 0$, $\mathcal{L}$ linear operator

Parametrization approach[3] :

- Suppose that: $f(x) = \mathcal{G}_x[g]$
- Then we impose the constraint on the operator $\mathcal{G}_x$:

$$\mathcal{L}[\mathcal{G}_x] = 0$$

- Gaussian processes are closed under linear transformation

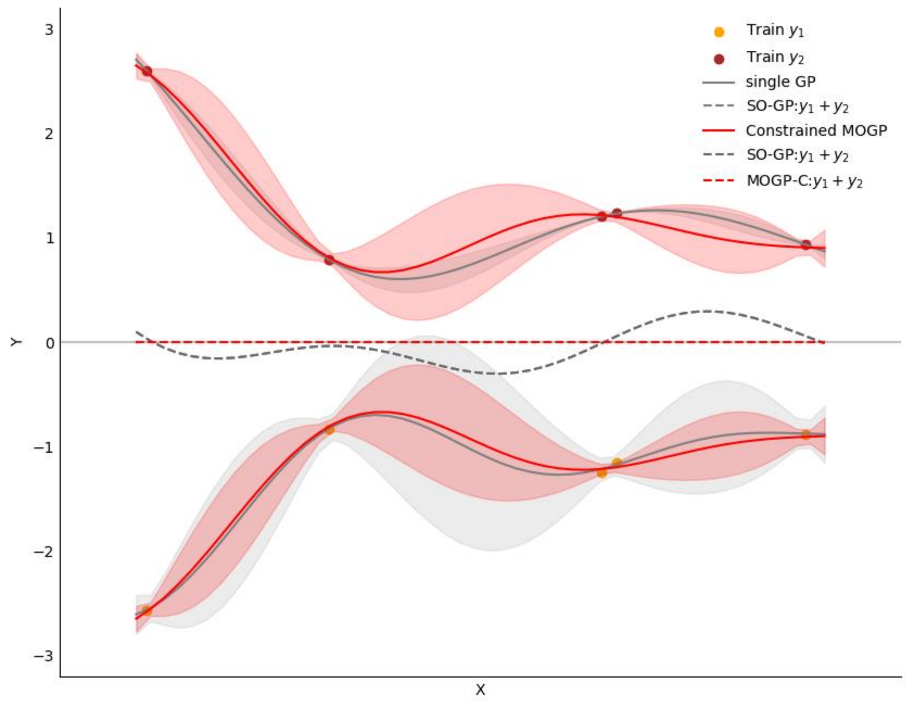
$$g \sim \mathcal{GP}(\mu_g(x), k_g(x, x'))$$

$$\mu_f(x) = \mathcal{G}_x[\mu_g], \qquad k_f = \mathcal{G}_x k_g \mathcal{G}_x^T$$

Illustration: $\mathcal{L}[f] = f_1 + f_2 = 0$

- $\mathcal{G}_x = [1, -1]$
- $\mu_f(x) = [\mu_g, -\mu_g]^T, \ k_f = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} k_g$

Constrained GP illustration



[3] Jidling, C., Wahlström, N., Wills, A., & Schön, T. B. (2017)

Constrained Gaussian Process regression

Given a linear constraint on f: $\mathcal{L}[f] = 0$, $\mathcal{L}$ linear operator

Parametrization approach[3] :

- Suppose that: $f(x) = \mathcal{G}_x[g]$
- Then we impose the constraint on the operator $\mathcal{G}_x$:

$$\mathcal{L}[\mathcal{G}_x] = 0$$

- Gaussian processes are closed under linear transformation

$$g \sim \mathcal{GP}(\mu_g(x), k_g(x, x'))$$

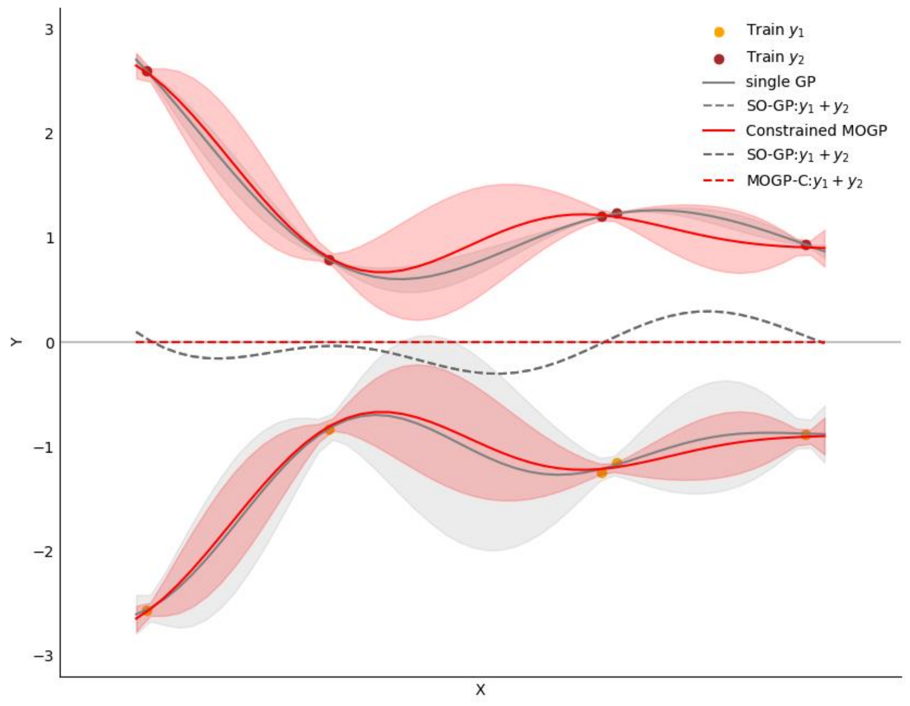
$$\mu_f(x) = \mathcal{G}_x[\mu_g], \qquad k_f = \mathcal{G}_x k_g \mathcal{G}_x^T$$

Illustration: $\mathcal{L}[f] = f_1 + f_2 = 0$

- $\mathcal{G}_x = [1, -1]$
- $\mu_f(x) = [\mu_g, -\mu_g]^T, \ k_f = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} k_g$

[3] Jidling, C., Wahlström, N., Wills, A., & Schön, T. B. (2017)

Constrained GP illustration



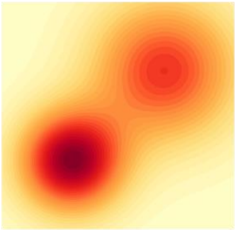
Problem: $\mathcal{O}(N^3P^3)$ complexity !

We need to combine it with a dimensionality reduction techniques,

but how, and which techniques ?

Space-modulation form (SMF) PCA for Vector fields [4]

- A 1-dimensional spatial field $y_i^1 \in \mathbb{R}^s$



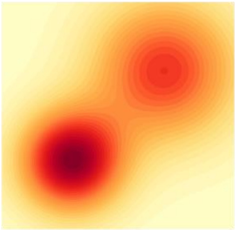
vector representation

$$y_i^1 = \begin{bmatrix} y_{i,1}^1 \\ \vdots \\ \vdots \\ \vdots \\ y_{i,s}^1 \end{bmatrix}$$

[4] R, W, Preisendorfer and C. D. Mobley. (1988)

Space-modulation form (SMF) PCA for Vector fields [4]

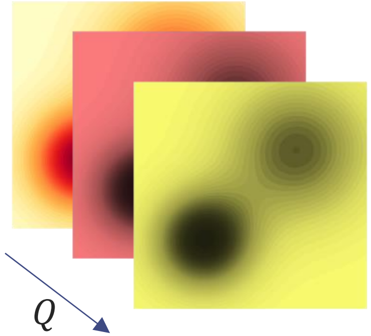
- A 1-dimensional spatial field $y_i^1 \in \mathbb{R}^s$



vector representation →

$$y_{\mathrm{i}}^1 = \begin{bmatrix} y_{i,1}^1 \\ \vdots \\ \vdots \\ \vdots \\ y_{i,S}^1 \end{bmatrix}$$

- A Q-dimensional spatial field $y_i \in \mathbb{R}^S \times \mathbb{R}^Q$



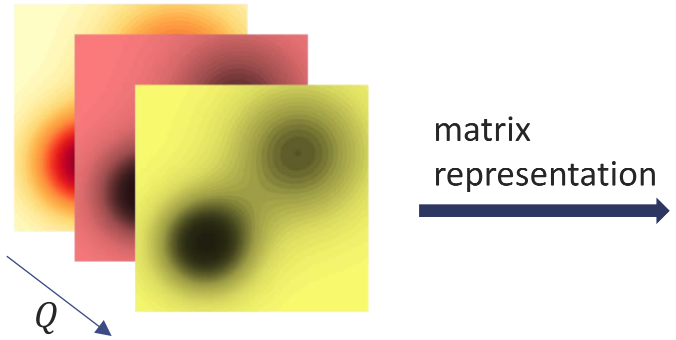
matrix representation →

$$y_{\mathrm{i}} = \begin{bmatrix} y_{i,1}^1 & & y_{i,1}^Q \\ \vdots & & \vdots \\ \vdots & \cdots & \vdots \\ \vdots & & \vdots \\ y_{i,S}^1 & & y_{i,S}^Q \end{bmatrix}$$

[4] R, W, Preisendorfer and C. D. Mobley. (1988)

Space-modulation form (SMF) PCA for Vector fields [4]

- A Q-dimensional spatial field $y_i \in \mathbb{R}^S \times \mathbb{R}^Q$

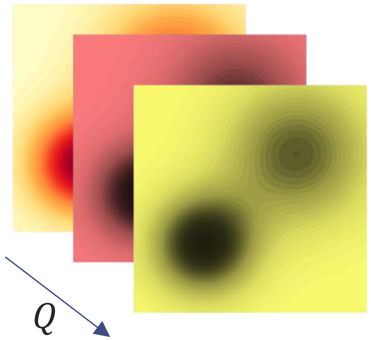


$$y_i = \begin{bmatrix} y_{i,1}^1 & & y_{i,1}^Q \\ \vdots & & \vdots \\ \vdots & \cdots & \vdots \\ \vdots & & \vdots \\ y_{i,S}^1 & & y_{i,S}^Q \end{bmatrix}$$

[4] R, W, Preisendorfer and C. D. Mobley. (1988)

Space-modulation form (SMF) PCA for Vector fields [4]

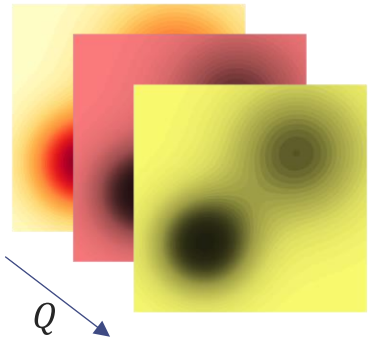
- A Q-dimensional spatial field $y_i \in \mathbb{R}^S \times \mathbb{R}^Q$



matrix representation

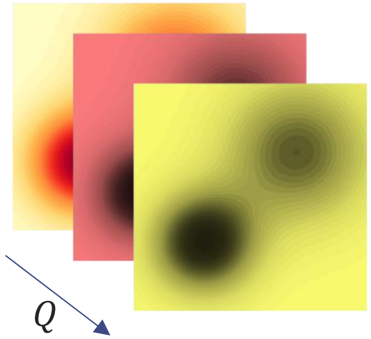
$$y_\mathrm{i} = \begin{bmatrix} y_{i,1}^{1} & & y_{i,1}^{Q} \\ \vdots & & \vdots \\ \vdots & \cdots & \vdots \\ \vdots & & \vdots \\ y_{i,S}^{1} & & y_{i,S}^{Q} \end{bmatrix}$$

- A Q-dimensional spatial field data $\{y_i \in \mathbb{R}^S \times \mathbb{R}^Q, i = 1, \ldots, N\}$



N iid samples

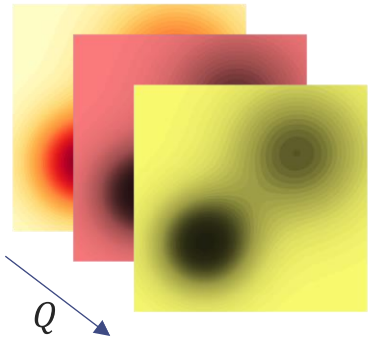
…



[4] R, W, Preisendorfer and C. D. Mobley. (1988)

Space-modulation form (SMF) PCA for Vector fields [4]

- A Q-dimensional spatial field $y_i \in \mathbb{R}^S \times \mathbb{R}^Q$



matrix representation →

$$y_{\mathrm{i}} = \begin{bmatrix} y_{i,1}^{1} & & y_{i,1}^{Q} \\ \vdots & & \vdots \\ \vdots & \cdots & \vdots \\ \vdots & & \vdots \\ y_{i,S}^{1} & & y_{i,S}^{Q} \end{bmatrix}$$

- A Q-dimensional spatial field data $\{y_i \in \mathbb{R}^S \times \mathbb{R}^Q, i = 1, \dots, N\}$

$$y_1 = \begin{bmatrix} y_{1,1}^{1} & & y_{1,1}^{Q} \\ \vdots & & \vdots \\ \vdots & \cdots & \vdots \\ \vdots & & \vdots \\ y_{1,S}^{1} & & y_{1,S}^{Q} \end{bmatrix}$$

N iid samples

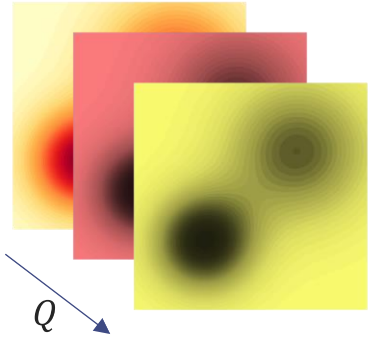
…

$$y_N = \begin{bmatrix} y_{N,1}^{1} & & y_{N,1}^{Q} \\ \vdots & & \vdots \\ \vdots & \cdots & \vdots \\ \vdots & & \vdots \\ y_{N,S}^{1} & & y_{N,S}^{Q} \end{bmatrix}$$

[4] R, W, Preisendorfer and C. D. Mobley. (1988)

Space-modulation form (SMF) PCA for Vector fields [4]

- A Q-dimensional spatial field $y_i \in \mathbb{R}^S \times \mathbb{R}^Q$



matrix representation →

$$y_\text{i} = \begin{bmatrix} y_{i,1}^1 & \cdots & y_{i,1}^Q \\ \vdots & & \vdots \\ y_{i,S}^1 & & y_{i,S}^Q \end{bmatrix}$$

- A Q-dimensional spatial field data $\{y_i \in \mathbb{R}^S \times \mathbb{R}^Q, i = 1, \dots, N\}$

$$y_1 = \begin{bmatrix} y_{1,1}^1 & \cdots & y_{1,1}^Q \\ \vdots & & \vdots \\ y_{1,S}^1 & & y_{1,S}^Q \end{bmatrix}$$

N iid samples

…

$$y_N = \begin{bmatrix} y_{N,1}^1 & \cdots & y_{N,1}^Q \\ \vdots & & \vdots \\ y_{N,S}^1 & & y_{N,S}^Q \end{bmatrix}$$

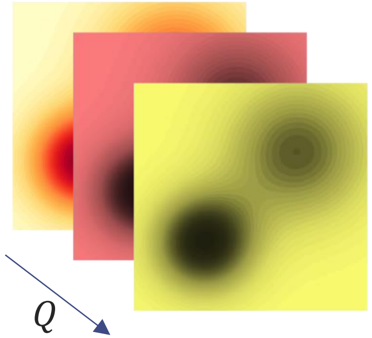
- Given a spatial vector $e \in \mathbb{R}^S$

$$e = \begin{bmatrix} e_1 \\ \vdots \\ e_S \end{bmatrix}$$

[4] R, W, Preisendorfer and C. D. Mobley. (1988)

Space-modulation form (SMF) PCA for Vector fields [4]

- A Q-dimensional spatial field $y_i \in \mathbb{R}^S \times \mathbb{R}^Q$



matrix representation

$$y_i = \begin{bmatrix} y_{i,1}^1 & & y_{i,1}^Q \\ \vdots & & \vdots \\ \vdots & \cdots & \vdots \\ \vdots & & \vdots \\ y_{i,S}^1 & & y_{i,S}^Q \end{bmatrix}$$

- A Q-dimensional spatial field data $\{y_i \in \mathbb{R}^S \times \mathbb{R}^Q, i = 1, \dots, N\}$

$$y_1 = \begin{bmatrix} y_{1,1}^1 & & y_{1,1}^Q \\ \vdots & & \vdots \\ \vdots & \cdots & \vdots \\ \vdots & & \vdots \\ y_{1,S}^1 & & y_{1,S}^Q \end{bmatrix}$$

N iid samples

$\cdots$

$$y_N = \begin{bmatrix} y_{N,1}^1 & & y_{N,1}^Q \\ \vdots & & \vdots \\ \vdots & \cdots & \vdots \\ \vdots & & \vdots \\ y_{N,S}^1 & & y_{N,S}^Q \end{bmatrix}$$

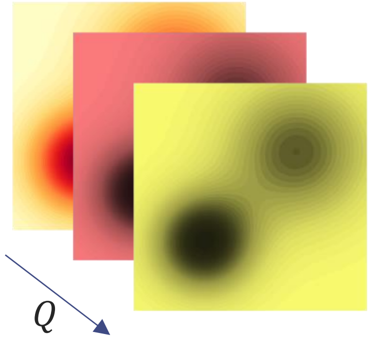
- Given a spatial vector $e \in \mathbb{R}^S$
- The projection of y_i in the direction $\boldsymbol{e}$ is:

$$\boldsymbol{\pi_e}(\boldsymbol{y_i}) = \boldsymbol{y_i^T e} = \begin{bmatrix} y_{i,1}^1 & \cdots & \cdots & \cdots & y_{i,S}^1 \\ & & \vdots & & \\ y_{i,1}^Q & \cdots & \cdots & \cdots & y_{i,S}^Q \end{bmatrix} \begin{bmatrix} e_1 \\ \vdots \\ \vdots \\ \vdots \\ e_S \end{bmatrix} = \begin{bmatrix} w_i^1 \\ \vdots \\ w_i^Q \end{bmatrix} \in \mathbb{R}^Q$$

[4] R, W, Preisendorfer and C. D. Mobley. (1988)

Space-modulation form (SMF) PCA for Vector fields [4]

- A Q-dimensional spatial field $y_i \in \mathbb{R}^S \times \mathbb{R}^Q$



matrix representation

$$y_i = \begin{bmatrix} y_{i,1}^1 & & y_{i,1}^Q \\ \vdots & & \vdots \\ \vdots & \cdots & \vdots \\ \vdots & & \vdots \\ y_{i,S}^1 & & y_{i,S}^Q \end{bmatrix}$$

- A Q-dimensional spatial field data $\{y_i \in \mathbb{R}^S \times \mathbb{R}^Q, i = 1, \ldots, N\}$

$$y_1 = \begin{bmatrix} y_{1,1}^1 & & y_{1,1}^Q \\ \vdots & & \vdots \\ \vdots & \cdots & \vdots \\ \vdots & & \vdots \\ y_{1,S}^1 & & y_{1,S}^Q \end{bmatrix}$$

N iid samples

…

$$y_N = \begin{bmatrix} y_{N,1}^1 & & y_{N,1}^Q \\ \vdots & & \vdots \\ \vdots & \cdots & \vdots \\ \vdots & & \vdots \\ y_{N,S}^1 & & y_{N,S}^Q \end{bmatrix}$$

- Given a spatial vector $\boldsymbol{e} \in \mathbb{R}^S$
- The projection of y_i in the direction $\boldsymbol{e}$ is:

$$\boldsymbol{\pi_e(y_i) = y_i^T e} = \begin{bmatrix} y_{i,1}^1 & \cdots & \cdots & \cdots & y_{i,S}^1 \\ & & \vdots & & \\ y_{i,1}^Q & \cdots & \cdots & \cdots & y_{i,S}^Q \end{bmatrix} \begin{bmatrix} e_1 \\ \vdots \\ \vdots \\ \vdots \\ e_S \end{bmatrix} = \begin{bmatrix} w_i^1 \\ \vdots \\ w_i^Q \end{bmatrix} \in \mathbb{R}^Q$$

- We define the **magnitude** of $\pi_e(y_i)$ by its L^2-norm in $\mathbb{R}^Q$:

$$\left|\left|y_i^T e\right|\right|^2$$

[4] R, W, Preisendorfer and C. D. Mobley. (1988)

Space-modulation form (SMF) PCA for Vector fields [4]

- A Q-dimensional spatial field data $\{y_i \in \mathbb{R}^S \times \mathbb{R}^Q, i = 1, \dots, N\}$

$$\boldsymbol{Y} = \begin{vmatrix} y_{1,1}^1 & \cdots & \cdots & \cdots & y_{1,S}^1 \\ & & \vdots & & \\ y_{1,1}^Q & \cdots & \cdots & \cdots & y_{1,S}^Q \\ \vdots & & \vdots & & \vdots \\ y_{N,1}^1 & \cdots & \cdots & \cdots & y_{N,S}^1 \\ & & \vdots & & \\ y_{N,1}^Q & \cdots & \cdots & \cdots & y_{N,S}^Q \end{vmatrix}$$

$\boldsymbol{Y} \in \mathbb{R}^{QN} \times \mathbb{R}^S$ the dataset matrix

- Given a spatial vector $\boldsymbol{e} \in \mathbb{R}^S$
- The projection of y_i in the direction $\boldsymbol{e}$ is:

$$\boldsymbol{\pi_e(y_i) = y_i^T e} = \begin{bmatrix} y_{i,1}^1 & \cdots & \cdots & \cdots & y_{i,s}^1 \\ & & \vdots & & \\ y_{i,1}^Q & \cdots & \cdots & \cdots & y_{i,s}^Q \end{bmatrix} \begin{bmatrix} e_1 \\ \vdots \\ \vdots \\ \vdots \\ e_S \end{bmatrix} = \begin{bmatrix} w_i^1 \\ \vdots \\ w_i^Q \end{bmatrix} \in \mathbb{R}^Q$$

- We define the **magnitude** of $\pi_e(y_i)$ by its L^2-norm in $\mathbb{R}^Q$:

$$\left|\left|y_i^T e\right|\right|^2$$

- Hence, the **variance** of projection of the Q-dimensional field along $\boldsymbol{e}$ is

$$\frac{1}{N}\sum_{i=1}^N \left|\left|y_i^T e\right|\right|^2 = \mathrm{tr}(\frac{1}{N} Y^T Y)$$

[4] R, W, Preisendorfer and C. D. Mobley. (1988)

Space-modulation form (SMF) PCA for Vector fields [4]

- A Q-dimensional spatial field data $\{y_i \in \mathbb{R}^S \times \mathbb{R}^Q, i = 1, \ldots, N\}$

$$\boldsymbol{Y} = \begin{vmatrix} y_{1,1}^1 & \cdots & \cdots & \cdots & y_{1,S}^1 \\ & & \vdots & & \\ y_{1,1}^Q & \cdots & \cdots & \cdots & y_{1,S}^Q \\ \vdots & & \vdots & & \vdots \\ y_{N,1}^1 & \cdots & \cdots & \cdots & y_{N,S}^1 \\ & & \vdots & & \\ y_{N,1}^Q & \cdots & \cdots & \cdots & y_{N,S}^Q \end{vmatrix}$$

$\boldsymbol{Y} \in \mathbb{R}^{QN} \times \mathbb{R}^S$ the dataset matrix

The goal of space-modulation form PCA[4] is to find the subspace direction $\boldsymbol{e}$ such that the projection of the Q-dimensional field has maximum variance:

$$\underset{||\boldsymbol{e}||=1}{\mathbf{argmax}} \sum_{x=1}^{N} \left|\left|\boldsymbol{y}^T \boldsymbol{e}\right|\right|^2$$

⟹ As for classical PCA, this problem can be solved by performing SVD on the dataset matrix $\boldsymbol{Y}$

Dimensionality reduction : retain only a few $M \ll S$ eigenvectors that capture a large proportion of the total variance

$$\pi_M(y) = (\boldsymbol{w}_1, \ldots, \boldsymbol{w}_M) \in \mathbb{R}^{Q^M}$$

[4] R, W, Preisendorfer and C. D. Mobley. (1988)

PCA-Constrained GP

Space-modulation form PCA for vector fields [4] :

- Linear dimensionality reduction technique
- Explicitly data reconstruction

✓ **Preserves linear constraints on data: the PCA weights follow the linear constraint (1)**

$$\sum_{j=1}^{Q} \alpha^j(x_i) w_i^j = c(x_i)$$

[4] R, W, Preisendorfer and C. D. Mobley. (1988)

PCA-Constrained GP

Space-modulation form PCA for vector fields [4] :

- Linear dimensionality reduction technique
- Explicitly data reconstruction

✓ **Preserves linear constraints on data: the PCA weights follow the linear constraint (1)**

$$\sum_{j=1}^{Q} \alpha^{j}(x_i) w_i^{j} = c(x_i)$$

Proposed model

$x, y = (y_1, \dots, y_Q) \in \left(\mathbb{R}^D, \mathbb{R}^{Q^S}\right)$

Train

[4] R, W, Preisendorfer and C. D. Mobley. (1988)

PCA-Constrained GP

Space-modulation form PCA for vector fields [4] :

- Linear dimensionality reduction technique
- Explicitly data reconstruction

✓ **Preserves linear constraints on data: the PCA weights follow the linear constraint (1)**

$$\sum_{j=1}^{Q} \alpha^j(x_i) w_i^j = c(x_i)$$

Proposed model

$x, y = (y_1, \ldots, y_Q) \in \left(\mathbb{R}^D, \mathbb{R}^{Q^S}\right)$

↓ SMF PCA projection

$x, \pi_M(y) = (w_1, \ldots, w_Q) \in \left(\mathbb{R}^D, \mathbb{R}^{Q^M}\right), \; M \ll S$

Train

[4] R, W, Preisendorfer and C. D. Mobley. (1988)

PCA-Constrained GP

Space-modulation form PCA for vector fields [4] :

- Linear dimensionality reduction technique
- Explicitly data reconstruction

✓ **Preserves linear constraints on data: the PCA weights follow the linear constraint (1)**

$$\sum_{j=1}^{Q} \alpha^{j}(x_i) w_i^{j} = c(x_i)$$

$$\widehat{f_{\pi_M}}(\cdot): \mathcal{X} \longrightarrow \mathbb{R}^{Q^M}$$

Proposed model

$x, y = (y_1, \dots, y_Q) \in \left(\mathbb{R}^D, \mathbb{R}^{Q^S}\right)$

↓ SMF PCA projection

$x, \pi_M(y) = (w_1, \dots, w_Q) \in \left(\mathbb{R}^D, \mathbb{R}^{Q^M}\right), \ M \ll S$

↓

$\pi_M(y) = \widehat{f_{\pi_M}(x)}$

Train

[4] R, W, Preisendorfer and C. D. Mobley. (1988)

PCA-Constrained GP

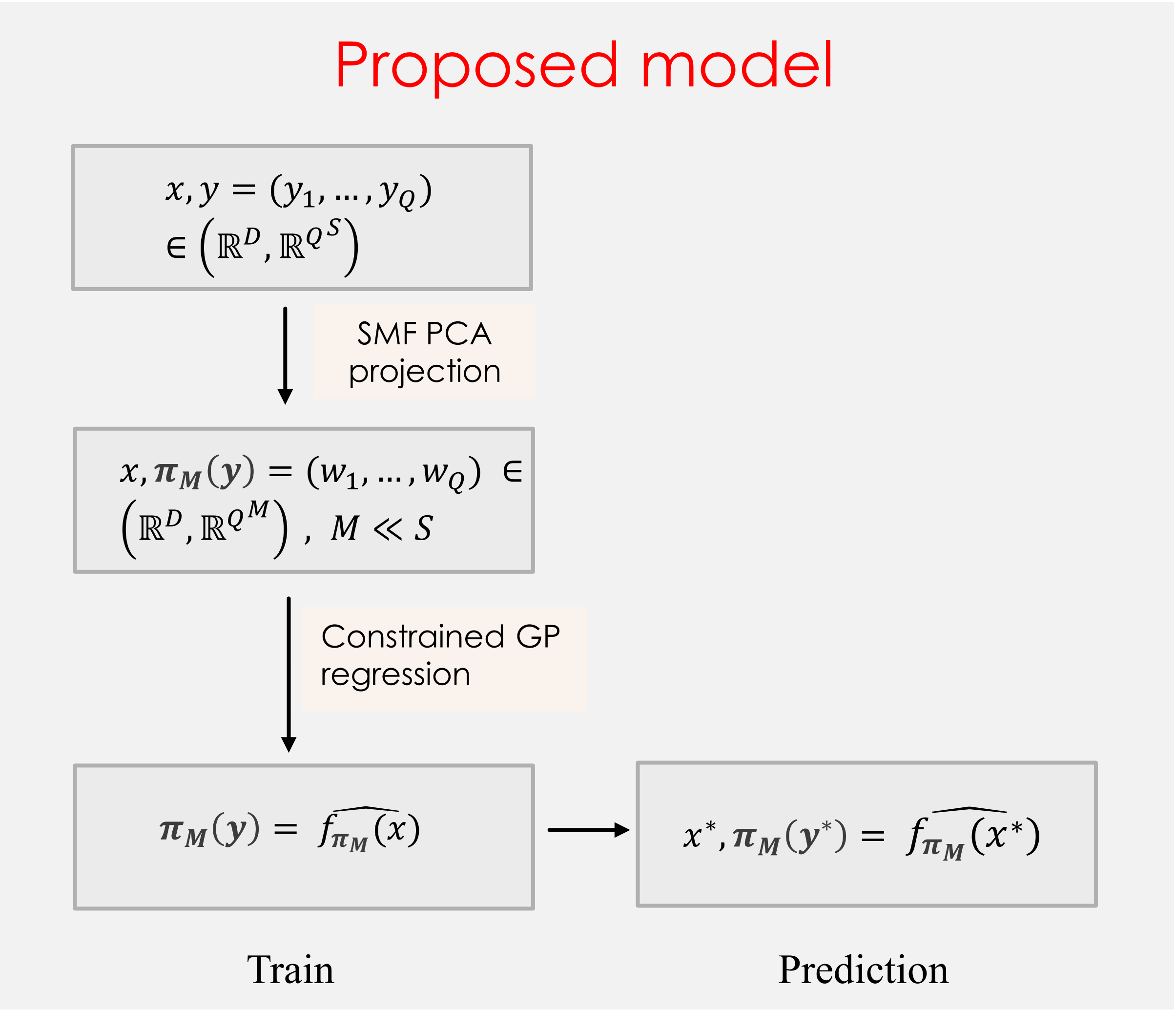
Space-modulation form PCA for vector fields [4] :

- Linear dimensionality reduction technique
- Explicitly data reconstruction

✓ **Preserves linear constraints on data: the PCA weights follow the linear constraint (1)**

$$\sum_{j=1}^{Q} \alpha^j(x_i) w_i^j = c(x_i)$$

$$\widehat{f_{\pi_M}}(\cdot) \colon \mathcal{X} \longrightarrow \mathbb{R}^{Q^M}$$



[4] R, W, Preisendorfer and C. D. Mobley. (1988)

PCA-Constrained GP

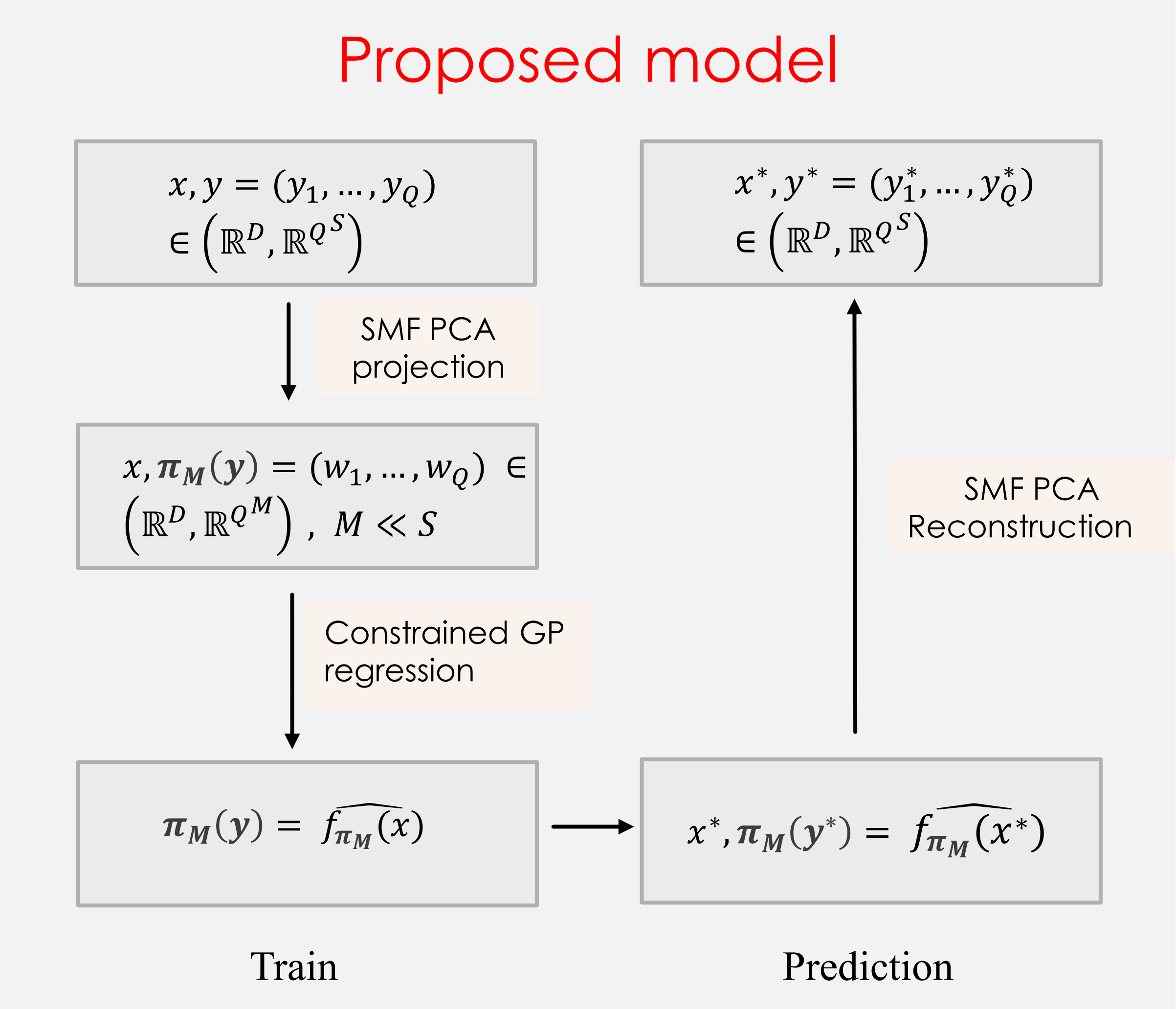
Space-modulation form PCA for vector fields [4] :

- Linear dimensionality reduction technique
- Explicitly data reconstruction

✓ **Preserves linear constraints on data: the PCA weights follow the linear constraint (1)**

$$\sum_{j=1}^{Q} \alpha^j(x_i) w_i^j = c(x_i)$$

$$\widehat{f_{\pi_M}}(\cdot): \mathcal{X} \longrightarrow \mathbb{R}^{Q^M}$$



[4] R, W, Preisendorfer and C. D. Mobley. (1988)

Metamodel for population dynamics

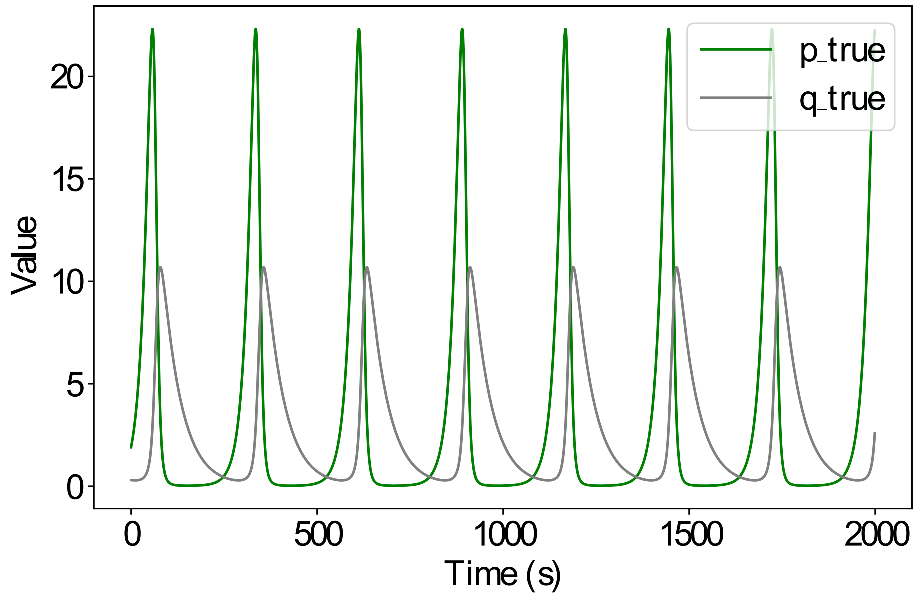
Lotka-Volterra equations:

p: prey density population
q: predator density population
(p_0, q_0): initial condition

$$\begin{cases} p'(t) = ap(t) - bp(t)q(t) \\ q'(t) = -cq(t) + dp(t)q(t) \end{cases} \quad (1)$$

Metamodeling problem:

- Inputs: $x = (b, d)$
- Outputs: $y(t, x) = [p(t, x), q(t, x), r(t, x), s(t, x)]^T$
  with $s = \log(p), r = \log(q)$



Metamodel for population dynamics

Lotka-Volterra equations:

p: prey density population
q: predator density population
(p_0, q_0): initial condition

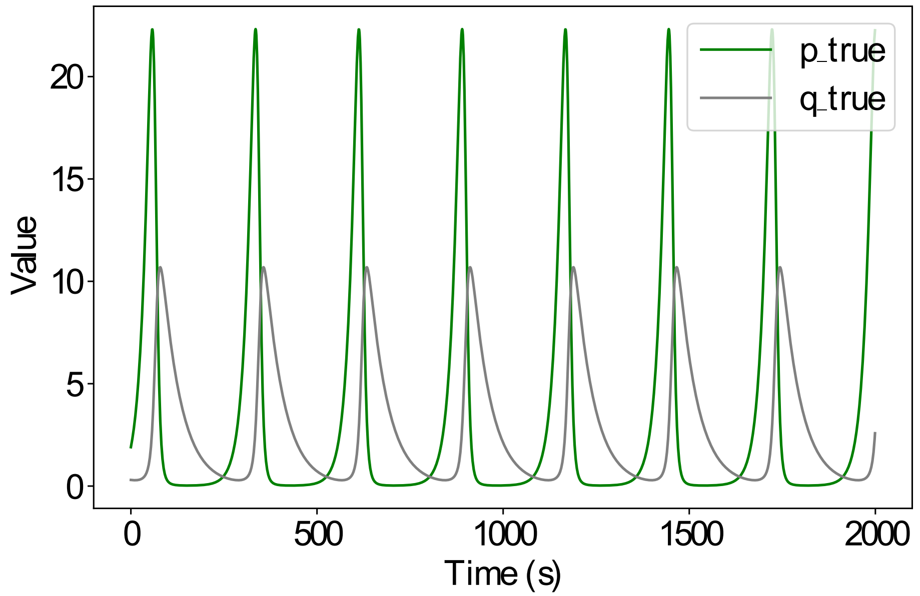
$$\begin{cases} p'(t) = ap(t) - bp(t)q(t) \\ q'(t) = -cq(t) + dp(t)q(t) \end{cases} \quad (1)$$

Metamodeling problem:

- Inputs: $x = (b, d)$
- Outputs: $y(t,x) = [p(t,x), q(t,x), r(t,x), s(t,x)]^T$
  with $s = \log(p), r = \log(q)$
- Input dependent linear constraint from the Hamiltonian structure [5] of (1):

$$\boldsymbol{dp(t) - cs(t) + bq(t) - ar(t) - C_0 = 0}$$

- Dataset : $\left(x_i, y(t_1, x_i), \dots, y(t_s, x_i)\right)_{1 \le i \le N}$



[5] Nutku, Y. (1990)

Metamodel for population dynamics

Lotka-Volterra equations:

p: prey density population
q: predator density population
(p_0, q_0): initial condition

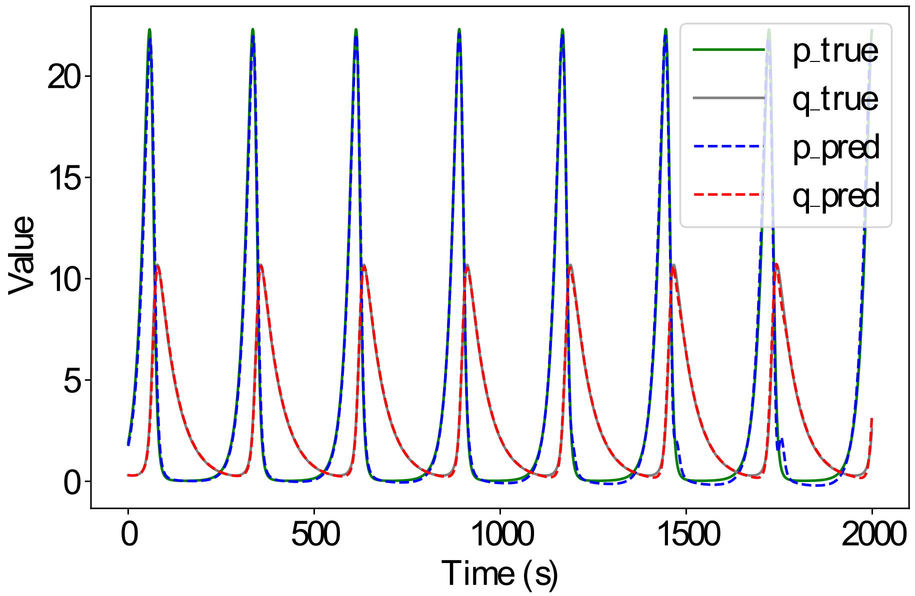
$$\begin{cases} p'(t) = ap(t) - bp(t)q(t) \\ q'(t) = -cq(t) + dp(t)q(t) \end{cases} \quad (1)$$

Metamodeling problem:

- Inputs: $x = (b, d)$
- Outputs: $y(t,x) = [p(t,x), q(t,x), r(t,x), s(t,x)]^T$ with $s = \log(p), r = \log(q)$
- Input dependent linear constraint from the Hamiltonian structure [5] of (1):

$$\boldsymbol{dp(t) - cs(t) + bq(t) - ar(t) - C_0 = 0}$$

- Dataset : $\left(x_i, y(t_1, x_i), \dots, y(t_s, x_i)\right)_{1 \le i \le N}$



[5] Nutku, Y. (1990)

Metamodel for population dynamics

Lotka-Volterra equations:

p: prey density population
q: predator density population
(p_0, q_0): initial condition

$$\begin{cases} p'(t) = ap(t) - bp(t)q(t) \\ q'(t) = -cq(t) + dp(t)q(t) \end{cases} \quad (1)$$

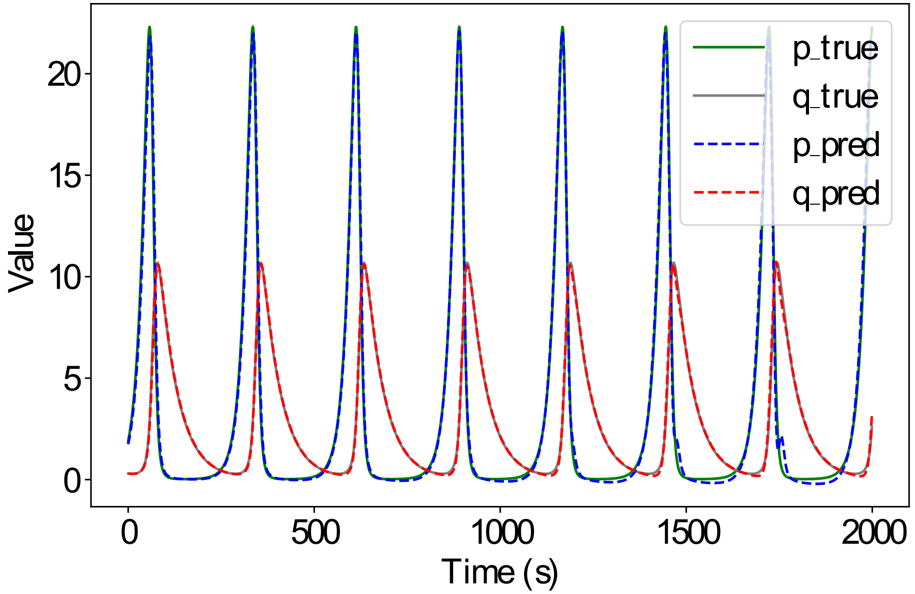
Metamodeling problem:

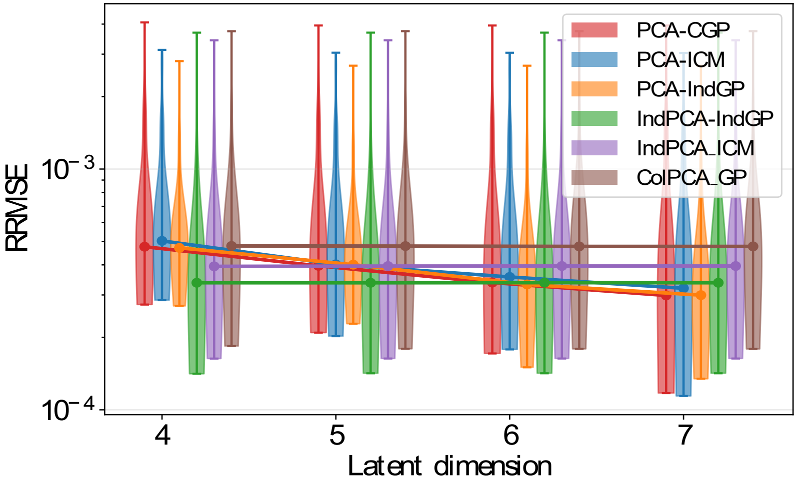
- Inputs: $x = (b, d)$
- Outputs: $y(t,x) = [p(t,x), q(t,x), r(t,x), s(t,x)]^T$
  with $s = \log(p), r = \log(q)$
- Input dependent linear constraint from the Hamiltonian structure [5] of (1):

$$\boldsymbol{dp(t) - cs(t) + bq(t) - ar(t) - C_0 = 0}$$

- Dataset : $\left(x_i, y(t_1, x_i), \dots, y(t_s, x_i)\right)_{1 \le i \le N}$

[5] Nutku, Y. (1990)



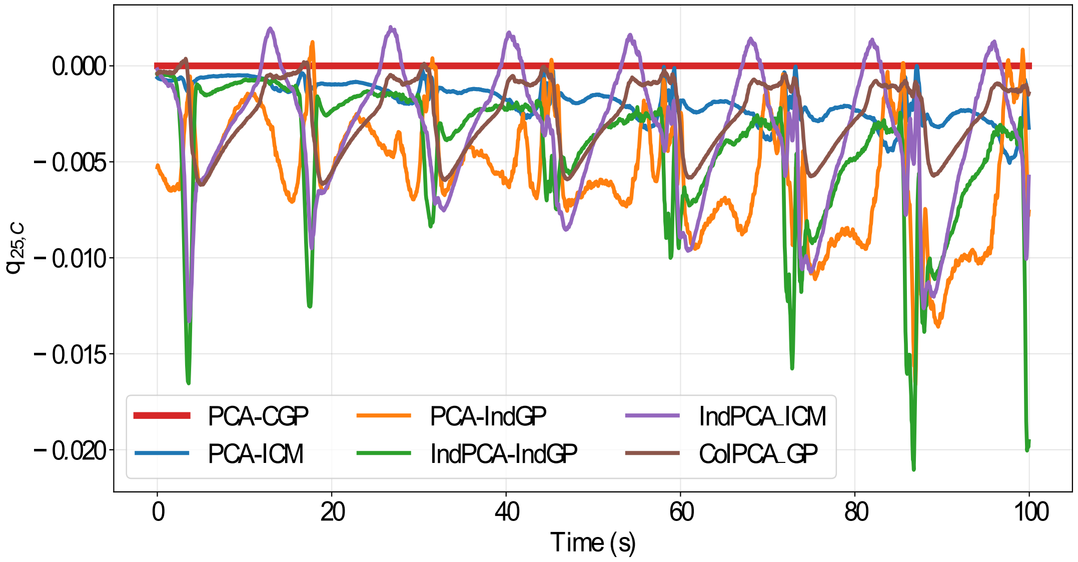


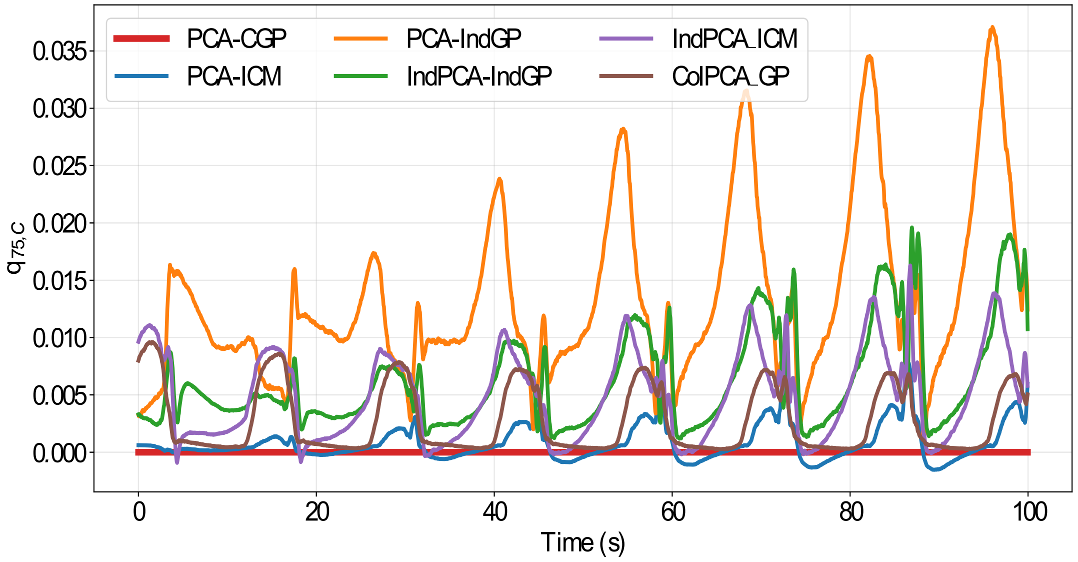
Constraint violation results

- $C(\nu, x_*) = \frac{\mathcal{L}(f(\nu,x_*))}{||f(\nu,\cdot)||_\infty}$,
- Quantile of level α of $\left\{C\left(\nu, x_*^{(i)}\right)\right\}_{i=1}^{N_*}$: $q_{\alpha,C} = C_{([N_*\alpha])}(\nu)$

Constraint violation results

- $C(\nu, x_*) = \frac{\mathcal{L}(f(\nu, x_*))}{||f(\nu,\cdot)||_\infty}$,
- Quantile of level α of $\left\{C\left(\nu, x_*^{(i)}\right)\right\}_{i=1}^{N_*}$: $q_{\alpha,C} = C_{([N_*\alpha])}(\nu)$





Standard PCA-GP does not satisfy the constraint at any time while our PCA-CGP satisfies the linear constraint all the time

Metamodel for the turbulent Reynolds stress tensor

- **Context**: Uncertainty propagation study in RANS modeling
- Turbulence closure model: standard $k-\varepsilon$
- **Inputs**: closure parameters of the standard $k-\varepsilon$ turbulence model

$$\boldsymbol{x} = (C_\mu, \sigma_k, \sigma_\varepsilon, C_{\varepsilon_1}, C_{\varepsilon_2})$$

- **Output**: Reynolds stress tensor $\tau_{ij} = \overline{u_i' u_j'}$ and the turbulent kinetic energy k

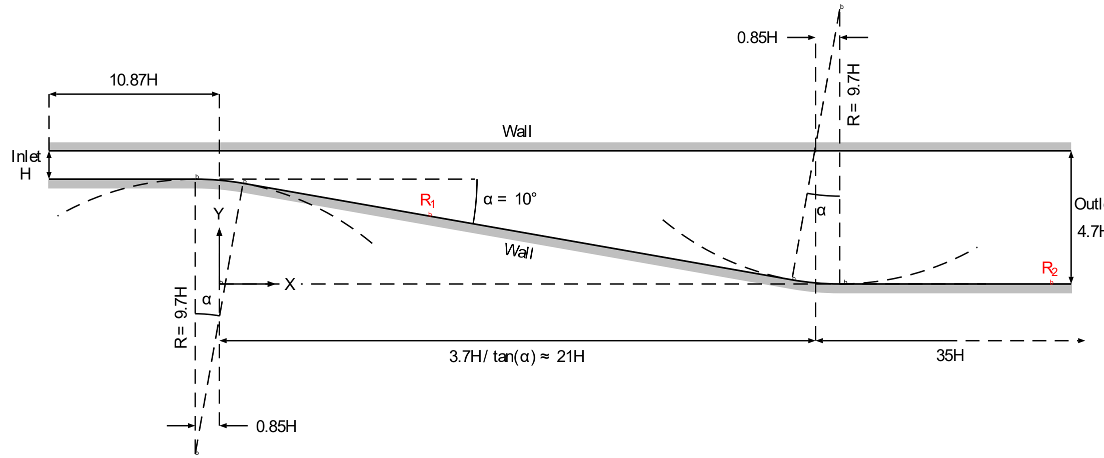
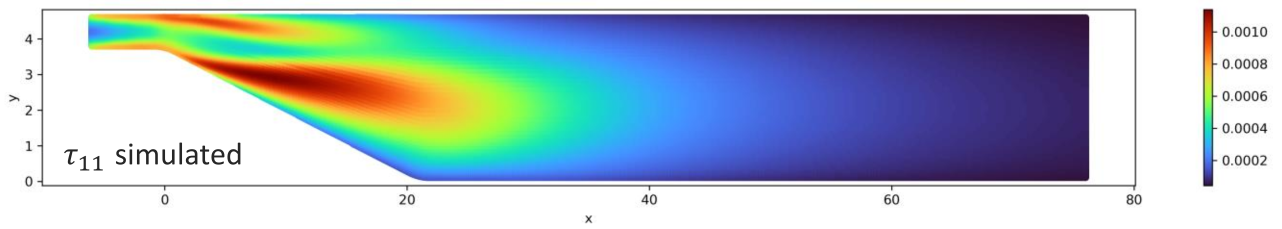


Figure: Scheme of the experiment carried out in Buice and Eaton, 2000. The wall of the channel are defined by straight lines and circular arcs of centers O_1 and O_2



Metamodel for the turbulent Reynolds stress tensor

- **Context**: Uncertainty propagation study in RANS modeling
- Turbulence closure model: standard $k - \varepsilon$
- **Inputs**: closure parameters of the standard $k - \varepsilon$ turbulence model

$$\boldsymbol{x} = (C_\mu, \sigma_k, \sigma_\varepsilon, C_{\varepsilon_1}, C_{\varepsilon_2})$$

- **Output**: Reynolds stress tensor $\tau_{ij} = \overline{u_i' u_j'}$ and the turbulent kinetic energy k
- The incompressibility condition is translated by:

$$\boldsymbol{\tau_{11} + \tau_{22} - \frac{4}{3}k = 0}$$

Datasets generation details:

- Mesh composed of 200 000 triangular elements
- CEA/DES CFD solver TrioCFD
- Reynolds number = 18000
- N_{train} = 50 and $N_{test} = 100$

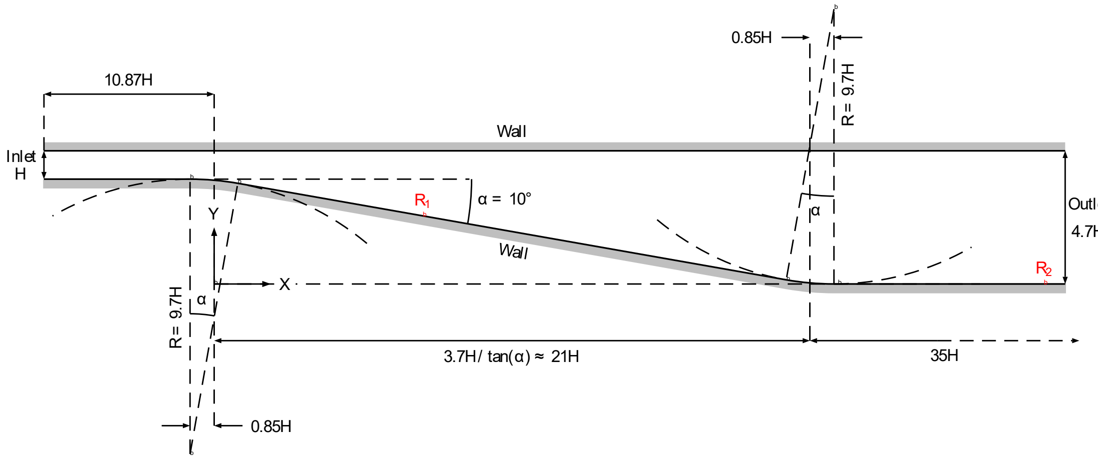
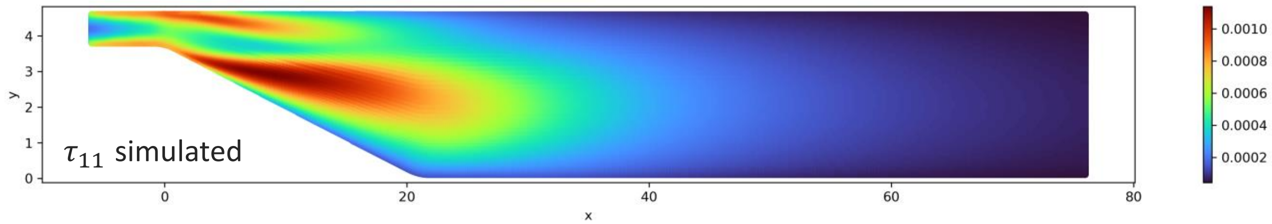


Figure: Scheme of the experiment carried out in Buice and Eaton, 2000. The wall of the channel are defined by straight lines and circular arcs of centers O_1 and O_2.



Metamodel for the turbulent Reynolds stress tensor

- **Context**: Uncertainty propagation study in RANS modeling
- Turbulence closure model: standard $k - \varepsilon$
- **Inputs**: closure parameters of the standard $k - \varepsilon$ turbulence model

$$\boldsymbol{x} = \left(C_\mu, \sigma_k, \sigma_\varepsilon, C_{\varepsilon_1}, C_{\varepsilon_2}\right)$$

- **Output**: Reynolds stress tensor $\tau_{ij} = \overline{u_i' u_j'}$ and the turbulent kinetic energy k
- The incompressibility condition is translated by:

$$\boldsymbol{\tau_{11} + \tau_{22} - \frac{4}{3}k = 0}$$

Datasets generation details:

- Mesh composed of 200 000 triangular elements
- CEA/DES CFD solver TrioCFD
- Reynolds number = 18000
- N_{train} = 50 and $N_{test} = 100$

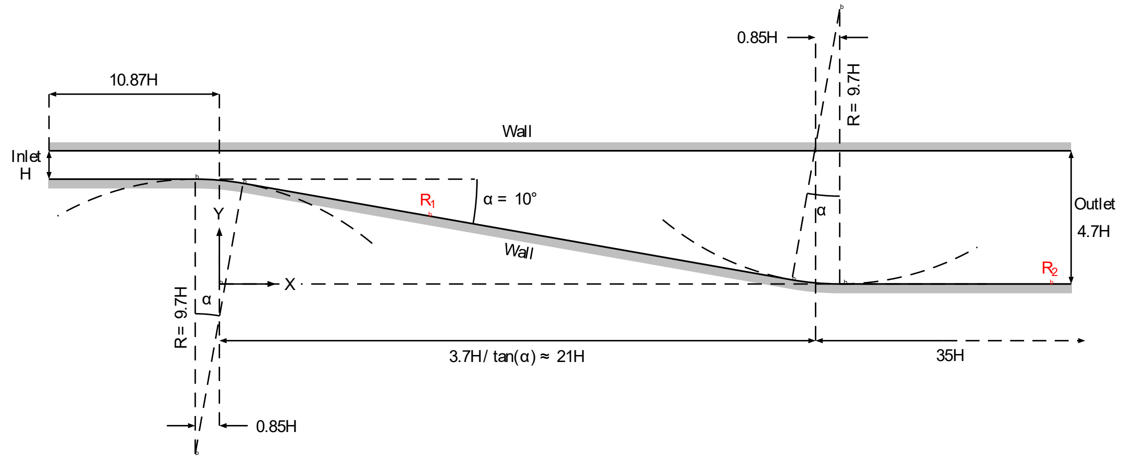


Figure: Scheme of the experiment carried out in Buice and Eaton, 2000. The wall of the channel are defined by straight lines and circular arcs of centers O_1 and O_2.

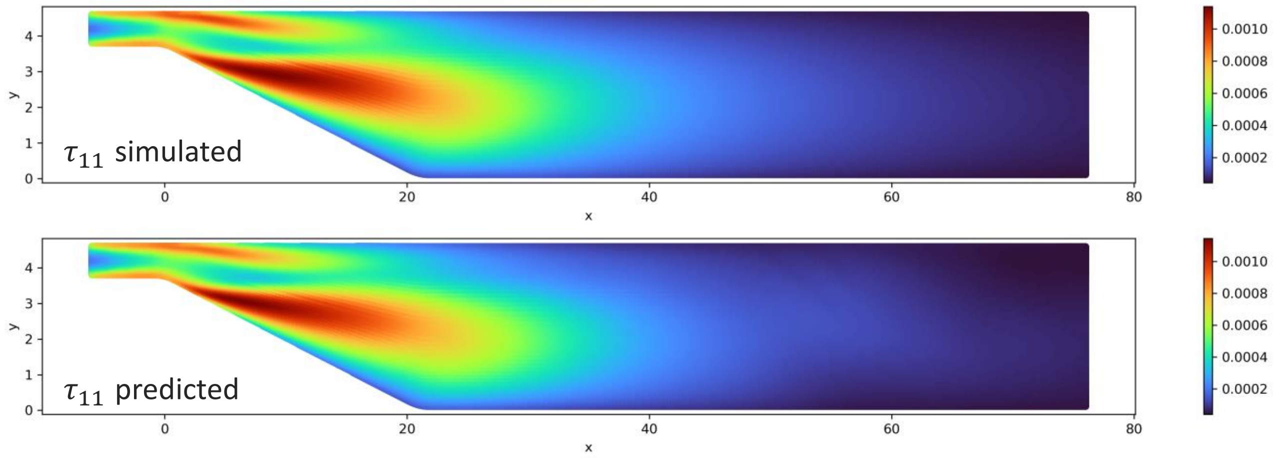


Figure: Simulation vs prediction results of the τ_{11} component on the fine mesh with 107,493 points.

Results on the LEVM Reynolds stress tensor

Model	$\left\| \tau_{11} + \tau_{22} - \frac{4k}{3} \right\|_{\infty}$	R^2	RRMSE
PCA-CGP	**4.33e-15 ± 1.66e-15**	0.9963 ± 0.0294	0.0002 ± 0.0018
PCA-ICM	1.08e-02 ± 2.88e-03	0.9961 ± 0.0294	0.0002 ± 0.0018
PCA-IndGP	2.00e-01 ± 6.44e-02	0.9977 ± 0.0298	0.0001 ± 0.0018
IndPCA-GP	1.69e-01 ± 7.31e-02	**0.9978 ± 0.0041**	**0.0001 ± 0.0003**
IndPCA-ICM	1.16e-02 ± 3.33e-03	0.9952 ± 0.0034	0.0003 ± 0.0002
ColPCA -GP	5.37e-04 ± 3.80e-05	**0.9978 ± 0.0042**	**0.0001 ± 0.0003**

The results on the quality of prediction are comparable between all methods while only our PCA-CGP strictly satisfies the linear constraint

Conclusion & Perspective

We proposed PCA-Constrained GP model to handle metamodeling problem with linear constraint in high dimensional context

- ✓ Gaussian process regression is preferred for its probabilistic framework and small data context
- ✓ Our framework guarantees constraint satisfaction globally and not just on set of collocation points
- ✓ A comparison against state-of-the-art physical fields models illustrates the efficiency of the PCA-CGP model

Perspective :

- ➢ Extend the combination of Constrained GP with non linear dimensionality reduction techniques (Autoencoder, isomap, kernel PCA)
- ➢ Extend the method to problems with a non-fixed mesh

⟹ A **paper** on the prediction of physical fields under linear constraints is currently being finalized. **Submission** is planned for October

References

[1] Higdon, D., Gattiker, J., Williams, B., & Rightley, M. (2008). Computer model calibration using high- dimensional output. Journal of the American Statistical Association.

[2] Alvarez, M. A., Rosasco, L., & Lawrence, N. D. (2012). Kernels for vector-valued functions: A review. Foundations and Trends® in Machine Learning.

[3] Jidling, C., Wahlström, N., Wills, A., & Schön, T. B. (2017). Linearly constrained Gaussian processes. Advances in neural information processing systems.

[4] R. W. Preisendorfer and C. D. Mobley. (1988). Principal component analysis in meteorology and oceanography. Elsevier.

[5] Nutku, Y. (1990). Hamiltonian structure of the Lotka-Volterra equations. Physics Letters A.

Acknowledgments:

This work has been funded by the French *Agence Nationale de la Recherche* ExaMA project, under the France 2030 initiative, with reference ANR-22-EXNU-0002.

Thanks !

Any questions ?

PROGRAMME DE RECHERCHE

NUMÉRIQUE POUR L'EXASCALE

Appendix : Gaussian process regression

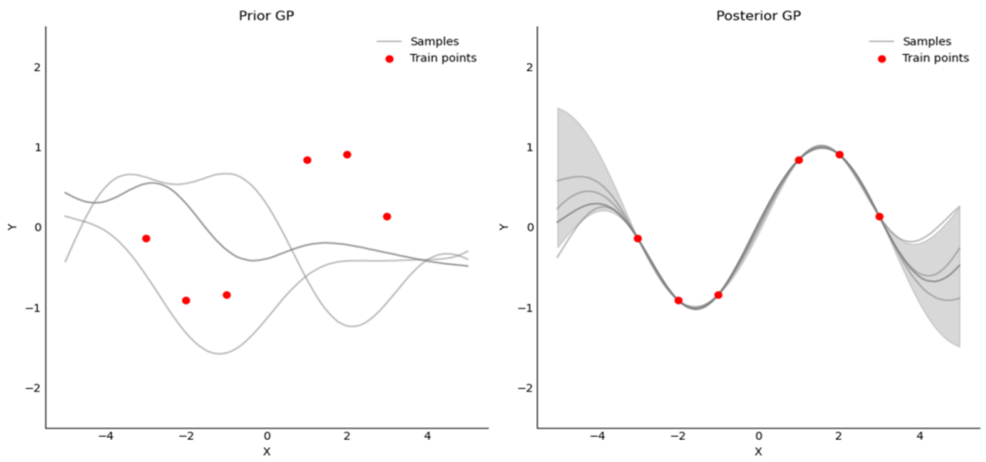
- **Single-output GP regression**[2]

$(x_i, y_i)_{\{i=1,\ldots,N\}} \in \mathcal{X} \times \mathbb{R}, \quad y_i = f(x_i) + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0, \sigma^2)$

Prior: $f(x) \sim \mathcal{GP}(0, k(x, x'))$, Kernel: $k(\cdot,\cdot) : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$

Posterior: $f(x_*)|y \sim \mathcal{N}(f_*, \text{cov}(f_*))$

- $f_* = k_*^T (k(X,X) + \sigma^2 I)^{-1}$
- $\text{cov}(f_*) = k(x_*, x_*) - k_*^T (k(X,X) + \sigma^2 I)^{-1} k_*$



- **Multi-Output GP (MOGP)** [3]

$(x_i, \boldsymbol{y}_i)_{\{i=1,\ldots,N\}} \in \mathcal{X} \times \mathbb{R}^P$

$\boldsymbol{y}_i = \boldsymbol{f}(x_i) + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0, \Sigma)$

Prior: $\boldsymbol{f}(x) \sim \mathcal{GP}\big(0, \boldsymbol{k}(x, x')\big)$

Kernel: $\boldsymbol{k}(\cdot,\cdot) : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}^P \times \mathbb{R}^P$

→ Matrix valued kernel

→ Symmetric positive definite

Kernel models:

- Intrinsic coregionalization model

$$\boldsymbol{k}(X,X) = B \otimes k(X,X)$$

- Linear model of coregionalization

$$\boldsymbol{k}(X,X) = \sum_{q=1}^{Q} B_q \otimes k_q(X,X),$$

$B_q \in \mathbb{R}^{P \times P}, : k_q(\cdot,\cdot) : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$

[2] Rasmussen, C. E. (2003)
[3] Alvarez , M. A., Rosasco, L., & Lawrence, N. D. (2012)

Appendix B

Lotka-Volterra Constraint :

- Constraint given by the Hamiltonian structure[5] of (1):

$$H(p,q) = dp - c\log(p) + bq - a\log(q) = C_0$$

- Reparametrized Hamiltonian structure:

$$\boldsymbol{dp - cs + bq - ar = C_0}$$

with $s = \log(p), r = \log(q)$

Constraint evaluation metric:

- $h(t) = \frac{1}{n_{test}} \sum_{i=1}^{n_{test}} h_i(t),$
- $h_i(t) = d\hat{p}_i(t) - c\hat{s}_i(t) + b\hat{q}_i(t) - a\hat{r}_i(t) - c_i(t)$

RRMSE metric :

$$RRMSE^2 = \frac{1}{N} \sum_{i=1}^{N} RRMSE^2(y_{sim}^i, y_{pred}^i)$$

$$RRMSE^2\left(y_{sim}^i, y_{pred}^i\right) = \frac{\left|\left|y_{sim}^i - y_{pred}^i\right|\right|_2^2}{S\left|\left|y_{sim}^i\right|\right|_\infty^2}$$

Appendix B

Uncertainty propagation

Parameters of the $k - \varepsilon$ standard turbulence model calibrated on experimental data :

- C_μ : constant
- σ_ε: The Prandtl turbulent dissipation number
- C_{ε_1}: The source term coefficient
- C_{ε_2}: The well term coefficient
- σ_k :

C_μ	σ_ε	$\sigma_{\boldsymbol{k}}$	C_{ε_1}	C_{ε_2}
0.09	1.30	1.00	1.44	1.92

Table: The usual values of the calibrated parameters