Optimal Kernel Learning for Gaussian Process Models with High-Dimensional Input

Lulu Kang, Minshen Xu

Department of Mathematics and Statistics

Joint Research Conference, June 2024

University of Massachusetts Amherst

Outline

Motivation

Dimension Reduction for GP

- Situation:
  - Gaussian Process (GP) is a popular tool for computer experiments which often contains non-linear relationships.
  - Many computer simulation models in engineering and scientific domains involve a large number of input variables and limited sample size for experiments.

Dimension Reduction for GP

- Situation:
  - Gaussian Process (GP) is a popular tool for computer experiments which often contains non-linear relationships.
  - Many computer simulation models in engineering and scientific domains involve a large number of input variables and limited sample size for experiments.
- Challenges:
  - Prediction accuracy: curse of dimensionality.
  - Computation: optimization in high dimensional variable space, nonconvex, matrix inversion.

Dimension Reduction for GP

- Situation:
  - Gaussian Process (GP) is a popular tool for computer experiments which often contains non-linear relationships.
  - Many computer simulation models in engineering and scientific domains involve a large number of input variables and limited sample size for experiments.
- Challenges:
  - Prediction accuracy: curse of dimensionality.
  - Computation: optimization in high dimensional variable space, nonconvex, matrix inversion.
- If the underlying system is only varied in a low dimensional input space of a few essential variables, then reducing the dimension of the input variables can help:
  - Alleviate the curse of dimensionality issue.
  - Computation involved in the estimation should be should be reduced.
  - Better understand the underlying system.

Existing Literature: identifying the active input variables

- Sensitivity analysis: can only identify the influential variables after fitting the GP in the original high dimensional space.

Existing Literature: identifying the active input variables

- Sensitivity analysis: can only identify the influential variables after fitting the GP in the original high dimensional space.
- Active Subspace: $y(\boldsymbol{x}) = Z(\boldsymbol{B}'\boldsymbol{x}) + \epsilon$, where $\boldsymbol{B}$ is the matrix of size $p \times d$ that projecting the input space from p to d dimension. So $\boldsymbol{B}$ is full-column-rank and $d < p$. Sometimes there is a constraint on $\boldsymbol{B}$: $\boldsymbol{B}'\boldsymbol{B} = \boldsymbol{I}_d$.
  - Bayesian approach: Single-Index or Multi-Index GP [Gramacy and Lian, 2012, Tripathy et al., 2016].
  - Active subspace for kriging[Constantine et al., 2014]: based on the gradient of the computer model.
  - Gradient-based kernel dimension reduction [Fukumizu and Leng, 2014] is used in Liu and Guillas [2017].

  Remark: it is challenging to estimate $\boldsymbol{B}$, either gradients or large number of simulations are needed.

Existing Literature: identifying the active input variables

- Sensitivity analysis: can only identify the influential variables after fitting the GP in the original high dimensional space.
- Active Subspace: $y(\boldsymbol{x}) = Z(\boldsymbol{B}'\boldsymbol{x}) + \epsilon$, where $\boldsymbol{B}$ is the matrix of size $p \times d$ that projecting the input space from p to d dimension. So $\boldsymbol{B}$ is full-column-rank and $d < p$. Sometimes there is a constraint on $\boldsymbol{B}$: $\boldsymbol{B}'\boldsymbol{B} = \boldsymbol{I}_d$.
  - Bayesian approach: Single-Index or Multi-Index GP [Gramacy and Lian, 2012, Tripathy et al., 2016].
  - Active subspace for kriging[Constantine et al., 2014]: based on the gradient of the computer model.
  - Gradient-based kernel dimension reduction [Fukumizu and Leng, 2014] is used in Liu and Guillas [2017].

  Remark: it is challenging to estimate $\boldsymbol{B}$, either gradients or large number of simulations are needed.
- Functional ANOVA decomposition: Borgonovo et al. [2018] and Sung et al. [2019].

GP Models

Gaussian Process Models

The GP model assumes the following probabilistic distribution for the response $y(\boldsymbol{x})$:

$$y(\boldsymbol{x}) = Z(\boldsymbol{x}) + \epsilon, \tag{1}$$

where

$$Z(\boldsymbol{x}) \sim GP(0, \tau^2 K(\cdot,\cdot)), \quad \text{and } \epsilon \sim^{iid} N(0, \sigma^2). \tag{2}$$

- Correlation function=a kernel function $K(\cdot,\cdot,\boldsymbol{\theta})$, and $\boldsymbol{\theta} \in \mathbb{R}_+^p$ are the correlation parameters. Gaussian kernel: $\exp\left(-\sum_{i=1}^p \theta_i (x_{1,i} - x_{2,i})^2\right)$ or $\exp(-\theta||\boldsymbol{x}_1 - \boldsymbol{x}_2||^2)$.
- Unknown parameters $\boldsymbol{\theta}$, τ^2, σ^2.

Estimation and Prediction

- Data $\{\boldsymbol{x}_i, y_i\}_{i=1}^n$. When $\sigma^2 > 0$, there should be replications at some design points $\boldsymbol{x}_i$. Otherwise, σ^2 is user specified.

Estimation and Prediction

- Data $\{\boldsymbol{x}_i, y_i\}_{i=1}^n$. When $\sigma^2 > 0$, there should be replications at some design points $\boldsymbol{x}_i$. Otherwise, σ^2 is user specified.
- MLE

$$\min_{\boldsymbol{\theta}\in\mathbb{R}_+} n\log\tau^2 + \log\det(K+\eta\boldsymbol{I}) + \frac{\boldsymbol{y}'(\boldsymbol{K}+\eta\boldsymbol{I})^{-1}\boldsymbol{y}}{\tau^2},$$

where $\eta = \sigma^2/\tau^2$ is the noise to signal ratio or *nugget effect* if $\sigma^2 = 0$. The MLE of τ^2 is $\frac{1}{n}(\boldsymbol{y}'(\boldsymbol{K}+\eta\boldsymbol{I})^{-1}\boldsymbol{y})$.

Estimation and Prediction

- Data $\{\boldsymbol{x}_i, y_i\}_{i=1}^n$. When $\sigma^2 > 0$, there should be replications at some design points $\boldsymbol{x}_i$. Otherwise, σ^2 is user specified.
- MLE
$$\min_{\boldsymbol{\theta}\in\mathbb{R}_+} n\log\tau^2 + \log\det(K+\eta\boldsymbol{I}) + \frac{\boldsymbol{y}'(\boldsymbol{K}+\eta\boldsymbol{I})^{-1}\boldsymbol{y}}{\tau^2},$$
where $\eta = \sigma^2/\tau^2$ is the noise to signal ratio or *nugget effect* if $\sigma^2 = 0$. The MLE of τ^2 is $\frac{1}{n}(\boldsymbol{y}'(\boldsymbol{K}+\eta\boldsymbol{I})^{-1}\boldsymbol{y})$.
- Prediction formula:
$$\hat{Y}(\boldsymbol{x}) = \boldsymbol{k}(\boldsymbol{x})(\boldsymbol{K}+\eta\boldsymbol{I})^{-1}\boldsymbol{y},$$
where $\boldsymbol{k}(\boldsymbol{x}) = [K(\boldsymbol{x},\boldsymbol{x}_1),\ldots,K(\boldsymbol{x},\boldsymbol{x}_n)]'$

GP model as a regularized RKHS regression

- $K(\cdot,\cdot)$: a positive definite kernel function on $\Omega \subset \mathbb{R}^p$.

GP model as a regularized RKHS regression

- $K(\cdot, \cdot)$: a positive definite kernel function on $\Omega \subset \mathbb{R}^p$.
- $\mathcal{H}_K$: reproducing kernel Hilbert space induced by K.

GP model as a regularized RKHS regression

- $K(\cdot,\cdot)$: a positive definite kernel function on $\Omega \subset \mathbb{R}^p$.
- $\mathcal{H}_K$: reproducing kernel Hilbert space induced by K.
- Given data $\{\boldsymbol{x}_i, y_i\}_{i=1}^n$, for any kernel function $K(\cdot,\cdot)$ and $f \in \mathcal{H}_K$, define the regularized loss function

$$Q_\eta(f, K, \mathcal{X}, \boldsymbol{y}) = Q(f, K, \mathcal{X}, \boldsymbol{y}) + \eta \|f\|^2_{\mathcal{H}_K}, \tag{3}$$

where $Q(f, K, \boldsymbol{X}, \boldsymbol{y})$ is a user-specified loss function measuring the goodness-of-fit and $\eta > 0$ is the regularization parameter.

GP model as a regularized RKHS regression

- $K(\cdot,\cdot)$: a positive definite kernel function on $\Omega \subset \mathbb{R}^p$.
- $\mathcal{H}_K$: reproducing kernel Hilbert space induced by K.
- Given data $\{\boldsymbol{x}_i, y_i\}_{i=1}^n$, for any kernel function $K(\cdot,\cdot)$ and $f \in \mathcal{H}_K$, define the regularized loss function

$$Q_\eta(f, K, \mathcal{X}, \boldsymbol{y}) = Q(f, K, \mathcal{X}, \boldsymbol{y}) + \eta \|f\|^2_{\mathcal{H}_K}, \tag{3}$$

  where $Q(f, K, \boldsymbol{X}, \boldsymbol{y})$ is a user-specified loss function measuring the goodness-of-fit and $\eta > 0$ is the regularization parameter.
- The penalized regression problem is to solve the following minimization problem

$$\min_{f \in \mathcal{H}_K} Q_\eta(f, K, \mathcal{X}, \boldsymbol{y}). \tag{4}$$

GP model as a regularized RKHS regression

- Since $f \in \mathcal{H}_K$, $f(\boldsymbol{x}) = \sum_{i=1}^{n} c_i K(\boldsymbol{x}, \boldsymbol{x}_i)$, it is equivalent to

$$\min_{\boldsymbol{c} \in \mathbb{R}^n} Q_\eta(\boldsymbol{c}, K) = Q(\boldsymbol{c}, K) + \eta \boldsymbol{c}^\top \boldsymbol{K} \boldsymbol{c}, \tag{5}$$

- Quadratic loss $Q(f, K) = \|\boldsymbol{y} - \boldsymbol{f}\|_2^2$ leads to optimal $\boldsymbol{c}^* = (\boldsymbol{K} + \eta \boldsymbol{I}_n)^{-1} \boldsymbol{y}$.
- Problem: kernel function is fixed, how to find this?

MKL Problem

Multiple kernel learning (MKL) problem

- MKL problem [Gönen and Alpaydın, 2011]: given data, how to find the optimal kernel function K^* from a space of kernel functions $\mathcal{K}$ for a specific kernel learning method, such as GP regression or the SVM?

Multiple kernel learning (MKL) problem

- MKL problem [Gönen and Alpaydın, 2011]: given data, how to find the optimal kernel function K^* from a space of kernel functions $\mathcal{K}$ for a specific kernel learning method, such as GP regression or the SVM?
- For GP regression,

$$Q_\eta(\mathcal{K}) = \min_{K \in \mathcal{K}} Q_\eta(K), \tag{6}$$

where $Q_\eta(K) = \min_{f \in \mathcal{H}_K} Q_\eta(f, K)$.

Multiple kernel learning (MKL) problem

- MKL problem [Gönen and Alpaydın, 2011]: given data, how to find the optimal kernel function K^* from a space of kernel functions $\mathcal{K}$ for a specific kernel learning method, such as GP regression or the SVM?
- For GP regression,

$$Q_\eta(\mathcal{K}) = \min_{K \in \mathcal{K}} Q_\eta(K), \tag{6}$$

  where $Q_\eta(K) = \min_{f \in \mathcal{H}_K} Q_\eta(f, K)$.
- Consider the squared-error loss function $Q(f, K) = \|\boldsymbol{y} - \boldsymbol{f}\|_2^2$. The minimization problem to find optimal kernel is

$$Q_\eta(\mathcal{K}) = \min_{K \in \mathcal{K}} \left\{ (\boldsymbol{y} - \boldsymbol{K}\boldsymbol{c}^*)^\top (\boldsymbol{y} - \boldsymbol{K}\boldsymbol{c}^*) + \mu {\boldsymbol{c}^*}^\top \boldsymbol{K}\boldsymbol{c}^* \right\}. \tag{7}$$

Discrete nature

- $\mathcal{A}_+(\Omega)$: the set of kernel functions such that for any set of design points in Ω the resulted kernel matrix $\boldsymbol{K}$ is positive definite.

Discrete nature

- $\mathcal{A}_+(\Omega)$: the set of kernel functions such that for any set of design points in Ω the resulted kernel matrix $\boldsymbol{K}$ is positive definite.
- If $\mathcal{K}$ is a compact and convex subset of $\mathcal{A}_+(\Omega)$ and $Q : \mathbb{R}^n \to \mathbb{R}$ is continuous, then the solution of (6) exists (Lemma 2 of Micchelli and Pontil [2005]).

Discrete nature

- $\mathcal{A}_+(\Omega)$: the set of kernel functions such that for any set of design points in Ω the resulted kernel matrix $\boldsymbol{K}$ is positive definite.
- If $\mathcal{K}$ is a compact and convex subset of $\mathcal{A}_+(\Omega)$ and $Q : \mathbb{R}^n \to \mathbb{R}$ is continuous, then the solution of (6) exists (Lemma 2 of Micchelli and Pontil [2005]).
- If $\mathcal{G} \subset \mathcal{A}_+(\Omega)$ is a compact set of basic kernels, $\mathcal{K}$ is the closure of the convex hall of $\mathcal{G}$, denoted by $\overline{\text{conv}(\mathcal{G})}$, the loss function $Q : \mathbb{R}^n \to \mathbb{R}$ is continuous and η is positive, then there exists a subset $\mathcal{T} \subset \mathcal{G}$ containing at most $n+2$ basic kernels such that $\mathcal{Q}_\eta(\mathcal{K})$ admits a minimizer $K \in \text{conv}(\mathcal{T})$ and $Q_\eta(\text{conv}(\mathcal{T})) = Q_\eta(\mathcal{K})$.

Discrete nature

- $\mathcal{A}_+(\Omega)$: the set of kernel functions such that for any set of design points in Ω the resulted kernel matrix $\boldsymbol{K}$ is positive definite.
- If $\mathcal{K}$ is a compact and convex subset of $\mathcal{A}_+(\Omega)$ and $Q : \mathbb{R}^n \to \mathbb{R}$ is continuous, then the solution of (6) exists (Lemma 2 of Micchelli and Pontil [2005]).
- If $\mathcal{G} \subset \mathcal{A}_+(\Omega)$ is a compact set of basic kernels, $\mathcal{K}$ is the closure of the convex hall of $\mathcal{G}$, denoted by $\overline{\text{conv}(\mathcal{G})}$, the loss function $Q : \mathbb{R}^n \to \mathbb{R}$ is continuous and η is positive, then there exists a subset $\mathcal{T} \subset \mathcal{G}$ containing at most $n+2$ basic kernels such that $\mathcal{Q}_\eta(\mathcal{K})$ admits a minimizer $K \in \text{conv}(\mathcal{T})$ and $Q_\eta(\text{conv}(\mathcal{T})) = Q_\eta(\mathcal{K})$.

General Message

It implies that the optimal kernel K^* solving $\min_{K\in\mathcal{K}} Q_\eta(K)$ is a convex combination of **at most** $n+2$ basic kernels, when $\mathcal{K}$ is a closed convex hull of the basic kernels. The unqiueness of the solution is achieved if Q is a strict convex function of $\mathbb{R}^n$.

Learning the Optimal Kernel

Optimal Design

- Approximate design [Atkinson, 2014, Kiefer, 1974]: a design ξ belongs to a class Ξ of probability measures on a compact design space $\mathcal{X} \in \mathbb{R}^d$, and Ξ includes all discrete measures.

Optimal Design

- Approximate design [Atkinson, 2014, Kiefer, 1974]: a design ξ belongs to a class Ξ of probability measures on a compact design space $\mathcal{X} \in \mathbb{R}^d$, and Ξ includes all discrete measures.
- $M(\xi)$: information matrix. Defined as $M(\xi) = \int_{\mathcal{X}} M(\boldsymbol{x})\xi(\mathrm{d}\,\boldsymbol{x})$, where $M(\boldsymbol{x})$ is the information matrix at a design point $\boldsymbol{x}$.

Optimal Design

- Approximate design [Atkinson, 2014, Kiefer, 1974]: a design ξ belongs to a class Ξ of probability measures on a compact design space $\mathcal{X} \in \mathbb{R}^d$, and Ξ includes all discrete measures.
- $M(\xi)$: information matrix. Defined as $M(\xi) = \int_{\mathcal{X}} M(\boldsymbol{x})\xi(\mathrm{d}\,\boldsymbol{x})$, where $M(\boldsymbol{x})$ is the information matrix at a design point $\boldsymbol{x}$.
- Design criteria, such as D- and I-optimal criteria, are convex in the information matrix M, are also convex in ξ [Kiefer, 1974].

Optimal Design

- Approximate design [Atkinson, 2014, Kiefer, 1974]: a design ξ belongs to a class Ξ of probability measures on a compact design space $\mathcal{X} \in \mathbb{R}^d$, and Ξ includes all discrete measures.
- $M(\xi)$: information matrix. Defined as $M(\xi) = \int_{\mathcal{X}} M(\boldsymbol{x})\xi(\mathrm{d}\,\boldsymbol{x})$, where $M(\boldsymbol{x})$ is the information matrix at a design point $\boldsymbol{x}$.
- Design criteria, such as D- and I-optimal criteria, are convex in the information matrix M, are also convex in ξ [Kiefer, 1974].
- The optimal design ξ^* minimizing such a design criterion consists of m support points $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m\} \subset \mathcal{X}$ and the optimal weights $\boldsymbol{\lambda}^*$, where $0 < \lambda_i^* \leq 1$ and $\sum_{i=1}^m \lambda_i^* = 1$. Thus, λ_i^* is the optimal probabilty mass allocated to each support point $\boldsymbol{x}_i^*$.

Design and Kernel

- We extend the concept of *design* to optimal kernel learning.

Design and Kernel

- We extend the concept of *design* to optimal kernel learning.
- Design ξ: a probability measure $\xi \in \Xi$.
- Ξ is a class of probability measures on the compact set of basic kernels $\mathcal{G} \subset \mathcal{A}_+(\Omega)$ including all discrete measures.

Design and Kernel

- We extend the concept of *design* to optimal kernel learning.
- Design ξ: a probability measure $\xi \in \Xi$.
- Ξ is a class of probability measures on the compact set of basic kernels $\mathcal{G} \subset \mathcal{A}_+(\Omega)$ including all discrete measures.
- $\mathcal{K} = \overline{\text{conv}(\mathcal{G})}$.

Design and Kernel

- We extend the concept of *design* to optimal kernel learning.
- Design ξ: a probability measure $\xi \in \Xi$.
- Ξ is a class of probability measures on the compact set of basic kernels $\mathcal{G} \subset \mathcal{A}_+(\Omega)$ including all discrete measures.
- $\mathcal{K} = \overline{\text{conv}(\mathcal{G})}$.
- For any $K \in \mathcal{K}$, there exists a $\xi \in \Xi$, such that $K = \int G\xi(\mathrm{d}\, G)$, where G is the notation for any kernel in $\mathcal{G}$, and vice versa.
- If $\mathcal{G}$ is a countable and compact set, i.e., $\mathcal{G} = \{G_1, G_2, \ldots\}$, then $K = \sum_{i=1} \xi_i G_i$, where $0 \leq \xi_i \leq 1$ is the probability mass for G_i and $\sum \xi_i = 1$.

Design and Kernel

- We extend the concept of *design* to optimal kernel learning.
- Design ξ: a probability measure $\xi \in \Xi$.
- Ξ is a class of probability measures on the compact set of basic kernels $\mathcal{G} \subset \mathcal{A}_+(\Omega)$ including all discrete measures.
- $\mathcal{K} = \overline{\text{conv}(\mathcal{G})}$.
- For any $K \in \mathcal{K}$, there exists a $\xi \in \Xi$, such that $K = \int G\xi(\mathrm{d}\, G)$, where G is the notation for any kernel in $\mathcal{G}$, and vice versa.
- If $\mathcal{G}$ is a countable and compact set, i.e., $\mathcal{G} = \{G_1, G_2, \ldots\}$, then $K = \sum_{i=1} \xi_i G_i$, where $0 \leq \xi_i \leq 1$ is the probability mass for G_i and $\sum \xi_i = 1$.
- A kernel function K is then a function of ξ, i.e., $K(\xi)$.

Optimal Kernel

- Finding the optimal kernel $\Longleftrightarrow$ finding the optimal design ξ^* with m **support kernels** $\{K_1, \ldots, K_m\}$ selected from $\mathcal{G}$, borrowing the term **support points**, and the optimal weights $\boldsymbol{\lambda}^*$ corresponding to the support kernels. Here $0 < \lambda_i \leq 1$ for $i = 1, \ldots, m$, and $\sum_{i=1}^m \lambda_i = 1$.

Optimal Kernel

- Finding the optimal kernel $\Longleftrightarrow$ finding the optimal design ξ^* with m **support kernels** $\{K_1, \dots, K_m\}$ selected from $\mathcal{G}$, borrowing the term **support points**, and the optimal weights $\boldsymbol{\lambda}^*$ corresponding to the support kernels. Here $0 < \lambda_i \leq 1$ for $i = 1, \dots, m$, and $\sum_{i=1}^m \lambda_i = 1$.
- The optimal kernel then can be expressed by $K(\xi^*) = \sum_{i=1}^m \lambda_i^* K_i$.

Optimal Kernel

- Finding the optimal kernel $\Longleftrightarrow$ finding the optimal design ξ^* with m **support kernels** $\{K_1, \dots, K_m\}$ selected from $\mathcal{G}$, borrowing the term **support points**, and the optimal weights $\boldsymbol{\lambda}^*$ corresponding to the support kernels. Here $0 < \lambda_i \leq 1$ for $i = 1, \dots, m$, and $\sum_{i=1}^{m} \lambda_i = 1$.
- The optimal kernel then can be expressed by $K(\xi^*) = \sum_{i=1}^{m} \lambda_i^* K_i$.
- Recall previous result from MKL problem, $m \leq \min\{n + 2, |\mathcal{G}|\}$.

General Equivalence Theorem (1)

The effort invested in connecting the two is worthwhile because the theories and algorithms for solving optimal design can also be adapted to for optimal kernel learning.

Definition (Directional Direvative w.r.t. Design)

Given a compact set of kernel functions $\mathcal{G} \subset \mathcal{A}_+(\Omega)$, let ξ and ξ' be two probability measures in Ξ on $\mathcal{G}$, including all discrete measures. As a function of ξ, the directional derivative of $Q_\eta(\xi)$ in the direction of ξ' is

$$\phi(\xi', \xi) := \bigtriangledown_{\xi'} Q_\eta(\xi) = \lim_{\alpha \to 0^+} \frac{Q_\eta((1-\alpha)\xi + \alpha\xi') - Q_\eta(\xi)}{\alpha}. \qquad (8)$$

General Equivalence Theorem (2)

Proposition

The directional derivative of $Q_\eta(\xi)$ in the direction of ξ' is given as,

$$\phi(\xi', \xi) = \frac{\partial Q_\eta(\xi)}{\partial \alpha}\bigg|_{\alpha=0} = -\eta \boldsymbol{y}^\top((\boldsymbol{K}_\xi + \eta \boldsymbol{I}_n)^{-1}(\boldsymbol{K}_{\xi'} - \boldsymbol{K}_\xi)(\boldsymbol{K}_\xi + \eta \boldsymbol{I}_n)^{-1})\boldsymbol{y}, \quad (9)$$

where $\boldsymbol{K}_\xi$ and $\boldsymbol{K}_{\xi'}$ are the $n \times n$ kernel matrix computed by evaluating $K(\xi)$ and $K(\xi')$ on $\mathcal{X} = \{\boldsymbol{x}_i\}_{i=1}^n$.

Theorem (General Equivalence Theorem)

Assume the same definition of Ξ, $\mathcal{G}$, $\mathcal{K}$, and $Q_\eta(\cdot)$ as above. The following conditions of a design $\xi \in \Xi$ are equivalent:

(1) *The design $\xi^* \in \Xi$ minimizes $Q_\eta(\xi)$;*

(2) *$\phi(\xi', \xi^*) \geq 0$ holds for any $\xi' \in \Xi$;*

(3) *$\phi(G, \xi^*) \geq 0$ holds for any $G \in \mathcal{G}$, and the inequality become equality if G is a support kernel of ξ^*. Here, the derivative $\phi(G, \xi)$ is a simplified notation for $\phi(\xi_G, \xi)$, and ξ_G is a probability measure assigning unit probability to the single kernel G in $\mathcal{G}$*

Algorithm 1 Forward Stepwise Optimal Kernel Learning

- The General Equivalence Theorem provides insight on how to select the support kernels sequentially.
- In each iteration, we check the sign of $\phi(G, \xi^r)$ for any G that has not been selected into the current design ξ^r. If it is non-negative for all G, then ξ^r reaches the optimal. But if $\phi(G, \xi^r) < 0$ for some G, it indicates that G is a potential support kernel and should be added into the design. To achieve the maximum reduction of the loss function $Q_\eta(\xi^r)$, we add the kernel $K_{r+1} = \arg\min_G \phi(G, \xi^r) < 0$ into the current set of support kernels for ξ^r.

Algorithm 1

Algorithm 1 is a Fedorov-Wynn type of algorithm that iteratively forward select a basic kernel into the design as a support kernel and update the weights using **Algorithm 2** to the optimal weights.

Algorithm 2 Optimal-Weight Procedure

Corollary (Conditions of Optimal Weights)

Restrict the set of basic kernel $\mathcal{G}$ to be a finite set, $\mathcal{G} = \{K_1, \ldots, K_M\}$ and Ξ is the class of discrete measure on $\mathcal{G}$. For any $\xi \in \Xi$, the corresponding weight vector $\boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_M]^\top$ with $0 \leq \lambda_i \leq 1$ becomes the only variable that decides $Q_\eta(\xi)$. The following two conditions on the optimal design ξ^* and its weight vector $\boldsymbol{\lambda}^*$ are equivalent.

1. *The weight vector $\boldsymbol{\lambda}^*$ minimizes $Q_\eta(\xi)$;*
2. *For all K_i with $\lambda_i^* > 0$, $\phi(K_i, \xi^*) = 0$; for all K_i with $\lambda_i^* = 0$, $\phi(K_i, \xi^*) \geq 0$.*

Algorithm 2

Based on the Corollary, we can develop **Algorithm 2** that returns the optimal weights for a set of support kernels. It is a type of multiplicative algorithm.

Convergence

Theorem

Assume the optimal weight procedure in **Algorithm 2** *converges to the optimal solution. Given the compact set of basic kernels* $\mathcal{G} \subset \mathcal{A}_+(\Omega)$ *and let* $\mathcal{K} = \overline{conv(\mathcal{G})}$, *the design constructed by* **Algorithm 1** *(without the optional delete step at the end) converges to* ξ^* *that minimizes* $Q_\eta(\xi)$, *i.e.,*

$$\lim_{r \to \infty} Q_\eta(\xi^r) = Q_\eta(\xi).$$

Low-Dimensional Approximation

Construct $\mathcal{G}$ of lower dimension variables

- Lower Dimension Kernel Space: K_j is the kernel function on x_j; K_{ij} is the kernel function on (x_i, x_j); K_{ijk} is the kernel function on (x_i, x_j, x_k);...
- All kernels are radial basis functions, i.e., isotropic.
- For each K_j or K_{ij}, we can specify the possible correlation parameter $\theta_l \in [\theta_{\min}, \theta_{\max}]$.

Functional ANOVA

Consider the ANOVA (upto the second order) decomposition [Sung et al., 2017] of GP:

$$Z(\boldsymbol{x}) \approx \sum_{j=1}^{p} \sum_{m=1}^{M_j} \beta_j^m Z_j^m(x_j) + \sum_{j=1}^{p-1} \sum_{k=j+1}^{p} \sum_{m=1}^{M_{jk}} \beta_{jk}^m Z_{jk}^m(x_j, x_k) + \epsilon.$$

which is equivalent to approximate the kernel $Z(\boldsymbol{x})$ by

$$K(\cdot,\cdot) \approx \sum_{j=1}^{p} \sum_{m=1}^{M_j} \lambda_j^m K_j^m(\cdot,\cdot) + \sum_{j=1}^{p-1} \sum_{k=j+1}^{p} \sum_{m=1}^{M_{jk}} \lambda_{jk}^m K_{jk}^m(\cdot,\cdot).$$

Algorithm 3: Forward+Backward Construction + Heredity Principle

1. Construct one-dim basic kernel functions. Use **Algorithm 1** to construct the optimal kernel.

Algorithm 3: Forward+Backward Construction + Heredity Principle

1. Construct one-dim basic kernel functions. Use **Algorithm 1** to construct the optimal kernel.
2. Backward checking: remove the kernels whose weights are less than a user-specified threshold, say 0.05; update the weights.

Algorithm 3: Forward+Backward Construction + Heredity Principle

1. Construct one-dim basic kernel functions. Use **Algorithm 1** to construct the optimal kernel.
2. Backward checking: remove the kernels whose weights are less than a user-specified threshold, say 0.05; update the weights.
3. Identify the active dimensions whose corresponding kernels are selected. Based on weak or strong heredity principle, construct the two-dim basic kernel functions. Use **Algorithm 1** to construct the optimal kernel.

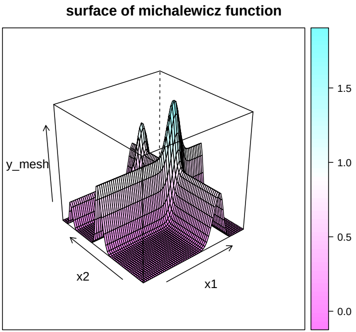
Algorithm 3: Forward+Backward Construction + Heredity Principle

1. Construct one-dim basic kernel functions. Use **Algorithm 1** to construct the optimal kernel.
2. Backward checking: remove the kernels whose weights are less than a user-specified threshold, say 0.05; update the weights.
3. Identify the active dimensions whose corresponding kernels are selected. Based on weak or strong heredity principle, construct the two-dim basic kernel functions. Use **Algorithm 1** to construct the optimal kernel.
4. Repeat the above steps for higher dimensions kernels until convergence condition is reached.

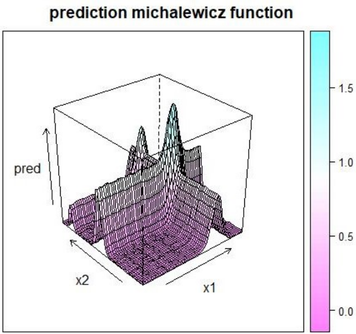
Parallel computing is incorporated.

Example

Michalewicz function



(a) the surface of Michalewicz function for $p = 2$



(b) the predicted surface of Michalewicz function for $p = 2$

Michalewicz function

Table: Performance of high dimensional Michalewicz function, $n = 300$, $p = 6$, $d = 10, 20, 60$

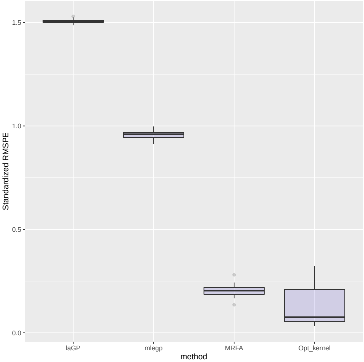
dimension	n	method	rmse_sd	fp	fn
10	300	lagp	0.9413	/	/
		mlegp	0.9110	2.6	0.78
		MRFA	0.1568	1.2	0
		optK	**0.0382**	**0**	**0**
20		lagp	0.9556	/	/
		mlegp	0.9452	11.86	0.24
		MRFA	0.1740	4.38	0
		optK	**0.0593**	**0**	**0**
60		lagp	1.5056	/	/
		mlegp	0.9565	53.68	0.02
		MRFA	0.2034	12.22	0
		optK	**0.1292**	**0**	**0**

Michalewicz function

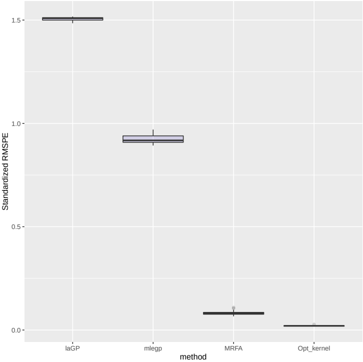
Table: Performance of high dimensional Michalewicz function, $n = 500$, $p = 6$, $d = 10, 20, 60$

dimension	n	method	rmse_sd	fp	fn
10	500	lagp	0.9128	/	/
		mlegp	0.8778	1.95	0.75
		MRFA	0.0574	1.4	0
		optK	**0.0200**	**0**	**0**
20		lagp	0.9151	/	/
		mlegp	0.9318	12.35	0.05
		MRFA	0.0652	5.5	0
		optK	**0.0197**	**0**	**0**
60		lagp	1.5053	/	/
		mlegp	0.9237	54	0
		MRFA	0.0828	13.55	0
		optK	**0.0202**	**0**	**0**

Michalewicz function



(a) n=300



(b) n=500

Figure: boxplot for 60-dimensional Michalewicz function

Conclusion

Conclusion

1. Existing literature: there are finite number of atom kernels from a compact and convex kernel space to form the optimal convex combination of kernel minimizing the regularized loss function.
2. Inspired by optimal design, we propose the construction algorithm to construct the optimal convex combination of kernels.
3. Combined with heredity principle, we construct low-dim kernel function as candidates and select them stage-wise.
4. Future directions: convex combination algorithms can be applied to nodes selection in deep neural networks.
5. Thanks & Questions?

References I

Atkinson, A. C. [2014], "Optimal design," *Wiley StatsRef: Statistics Reference Online*, pp. 1–17.

Borgonovo, E., Morris, M. D., and Plischke, E. [2018], "Functional ANOVA with multiple distributions: implications for the sensitivity analysis of computer experiments," *SIAM/ASA Journal on Uncertainty Quantification*, 6(1), 397–427.

Constantine, P. G., Dow, E., and Wang, Q. [2014], "Active subspace methods in theory and practice: applications to kriging surfaces," *SIAM Journal on Scientific Computing*, 36(4), A1500–A1524.

Fukumizu, K., and Leng, C. [2014], "Gradient-based kernel dimension reduction for regression," *Journal of the American Statistical Association*, 109(505), 359–370.

Gönen, M., and Alpaydın, E. [2011], "Multiple kernel learning algorithms," *Journal of machine learning research*, 12(Jul), 2211–2268.

Gramacy, R. B., and Lian, H. [2012], "Gaussian process single-index models as emulators for computer experiments," *Technometrics*, 54(1), 30–41.

Kiefer, J. [1974], "General Equivalence Theory for Optimum Designs (Approximate Theory)," *The Annals of Statistics*, 2(5), 849 – 879.
URL: *https://doi.org/10.1214/aos/1176342810*

Liu, X., and Guillas, S. [2017], "Dimension reduction for Gaussian process emulation: an application to the influence of bathymetry on tsunami heights," *SIAM/ASA Journal on Uncertainty Quantification*, 5(1), 787–812.

Micchelli, C. A., and Pontil, M. [2005], "Learning the kernel function via regularization," *Journal of machine learning research*, 6(Jul), 1099–1125.

Sung, C.-L., Wang, W., Plumlee, M., and Haaland, B. [2017], "Multi-Resolution Functional ANOVA for Large-Scale, Many-Input Computer Experiments," *arXiv preprint arXiv:1709.07064*, .

Sung, C.-L., Wang, W., Plumlee, M., and Haaland, B. [2019], "Multiresolution functional anova for large-scale, many-input computer experiments," *Journal of the American Statistical Association*, .

Tripathy, R., Bilionis, I., and Gonzalez, M. [2016], "Gaussian processes with built-in dimensionality reduction: Applications to high-dimensional uncertainty propagation," *Journal of Computational Physics*, 321, 191–223.