Détermination de la prédictivité d'un modèle via la sélection incrémentale de l'ensemble de test

Elias Fekhari [1] Bertrand Iooss [1] Joseph Muré [1] Luc Pronzato [2]
Maria João Rendas [2]

[1]EDF R&D - 6 quai Watier, Chatou, France

[2]CNRS, Université Côte d'Azur, Laboratoire I3S - 2000 route des lucioles, Sophia Antipolis, France

29 août 2022

Tester un modèle d'apprentissage automatique (ML)

Modèle d'apprentissage automatique (ou métamodèle)

$\eta_m : \mathbb{R}^d \to \mathbb{R}$ construit à partir d'un ensemble d'apprentissage $(\mathbf{X}_m, \mathbf{y}_m)$, métamodèle du véritable modèle $y : \mathbb{R}^d \to \mathbb{R}$

Ensemble d'apprentissage

$\mathbf{y}_m = [y(\mathbf{x}^{(1)}), \ldots, y(\mathbf{x}^{(m)})]$ sont les sorties observées aux points $\mathbf{X}_m = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)}\} \subset \mathbb{R}^d$

Comment certifier sa performance ?

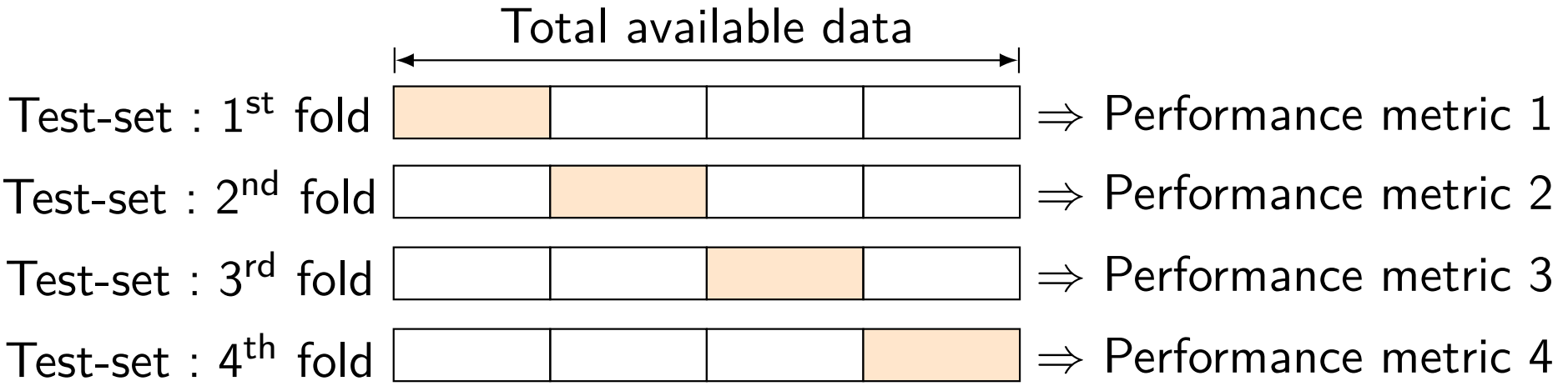
- Quel protocole de test utiliser ?
- Quelle métrique de performance utiliser ?

On ne peut jamais qu'estimer la véritable performance du métamodèle.

Méthodes classiques pour tester un modèle

Validation croisée

Méthodes usuelles : *k-fold* et *Leave-One-Out* (LOO) [1].



Limites de la validation croisée

- Coûteuse : il faut construire k modèles
- Moyenne la performance de modèles distincts : acceptable ?
- EDF doit pouvoir valider des modèles ML de prestataires.

Les ensembles d'apprentissage et de test doivent donc être indépendants.

Comment choisir un ensemble de test « optimal » ?

1. Tadayoshi Fushiki. "Estimation of prediction error by using K-fold cross-validation". In : *Statistics and Computing* 21.2 (2011), p. 137-146.

Qu'est-ce qu'un « bon » ensemble de test ?

Ensemble de test

$\mathbf{y}_n = [y(\mathbf{x}^{(1)}), \ldots, y(\mathbf{x}^{(n)})]$ sont les sorties observée aux points $\mathbf{X}_n = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n)}\} \subset \mathbb{R}^d$

- Itératif pour que l'estimation soit bonne quelle que soit la taille n.
- Représentatif de la loi μ du vecteur aléatoire $\mathbf{X}$.
- Complémentaire de $\mathbf{X}_m$: pour qu'un modèle construit sur l'union $\mathbf{X}_{n+m}$ soit meilleur

Ensemble des candidats

$\mathcal{S}$ est un sous-ensemble « assez dense » de $\mathbb{R}^d$ de cardinal $N \gg n$ discrétisant la loi μ.

Sélection itérative

À l'itération i, avec $\mathbf{X}_i = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(i)}\}$, optimiser la fonction $\mathcal{A}(\cdot|\mathbf{X}_i)$:

$$\mathbf{x}^{(i+1)} \in \underset{\mathbf{x} \in \mathcal{S} \setminus \mathbf{X}_i}{\arg\min}\, \mathcal{A}(\mathbf{x}|\mathbf{X}_i)\ . \tag{1}$$

Choix fondé sur la distance

Construction géométrique dans un ensemble borné : le nouveau point $\mathbf{x}$ est pris aussi éloigné que possible des points $\mathbf{x}^{(i)}$ précédemment sélectionnés.

Fully-Sequential Space-Filling [2] (FSSF)

À l'itération i, avec $\mathbf{X}_i = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(i)}\}$,

$$\mathbf{x}^{(i+1)} \in \underset{\mathbf{x} \in \mathcal{S} \setminus \mathbf{X}_i}{\arg\max} \left[\min_{j \in \{1,\ldots,i\}} \|\mathbf{x} - \mathbf{x}^{(j)}\| \right] . \tag{2}$$

- Si la loi μ n'est pas uniforme, il faut appliquer une transformation iso-probabiliste.
- Le FSSF est proche de l'algorithme CADEX (« Coffee house design »)

2. B. Shang et D. Apley. "Fully-sequential space-filling design algorithms for computer experiments". In : *Journal of Quality Technology* 53 (2020), p. 1-24.

Choix fondé sur la distance

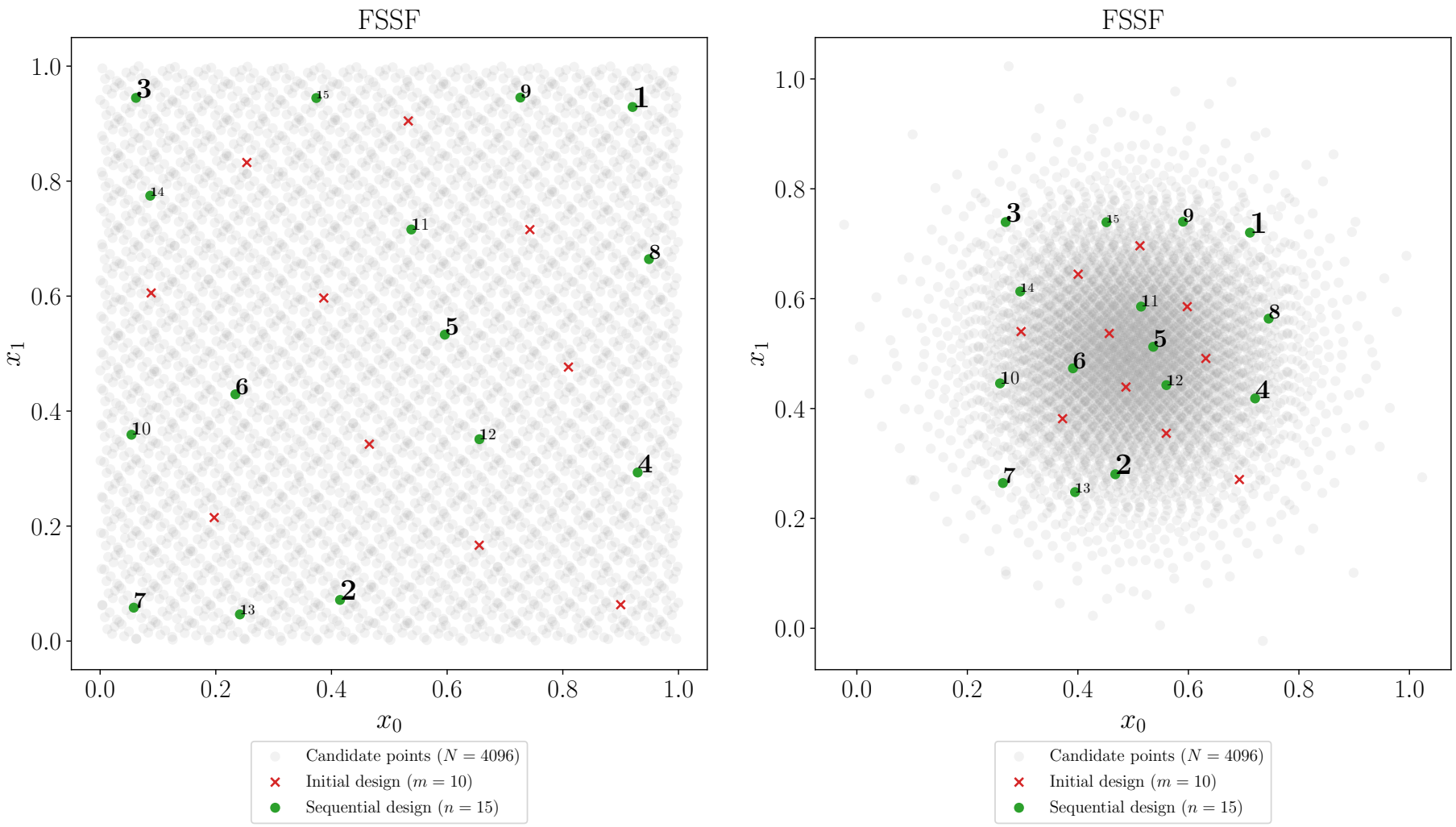


Figure – Ensembles de test séquentiels FSSF (uniforme et normal bivarié)

Maximum Mean Discrepancy [3]

Reproducing Kernel Hilbert Space (RKHS)

Soit une fonction symétrique définie positive $k : \mathcal{X}^2 \to \mathbb{R}$ (noyau).
Un RKHS $\mathcal{H}(k)$ est un espace de Hilbert de fonctions $f : \mathcal{X} \to \mathbb{R}$ tel que :

- $k(\cdot, \mathbf{x}) \in \mathcal{H}(k), \quad \forall \mathbf{x} \in \mathcal{X}$.
- (reproduction) $\langle f, k(\cdot, \mathbf{x}) \rangle_{\mathcal{H}(k)} = f(\mathbf{x}), \quad \forall \mathbf{x} \in \mathcal{X}, \forall f \in \mathcal{H}(k)$

Tout noyau défini positif définit un unique RKHS.

Maximum Mean Discrepancy (MMD)

Il s'agit d'une distance entre distributions P and Q définie par :

$$\mathrm{MMD}_k(P, Q) := \sup_{\|f\|_{\mathcal{H}(k)} \leq 1} \left| \int_{\mathcal{X}} f(\mathbf{x}) \mathrm{d}P(\mathbf{x}) - \int_{\mathcal{X}} f(\mathbf{x}) \mathrm{d}Q(\mathbf{x}) \right| \tag{3}$$

Un noyau k est dit caractéristique si $\mathrm{MMD}_k(P, Q) = 0 \Leftrightarrow P = Q$.

3. C.J. Oates. *Minimum Discrepancy Methods in Uncertainty Quantification*. Lecture Notes at ETICS Summer School. 2021.

Maximum Mean Discrepancy

Dans la suite, nous supposons k continu et borné. Alors[4]

$$\mathrm{MMD}_k(P, Q) = \|\mu_P - \mu_Q\|_{\mathcal{H}(k)} \quad \text{where} \quad \mu_P = \int k(\mathbf{x}, \cdot) dP(\mathbf{x}). \tag{4}$$

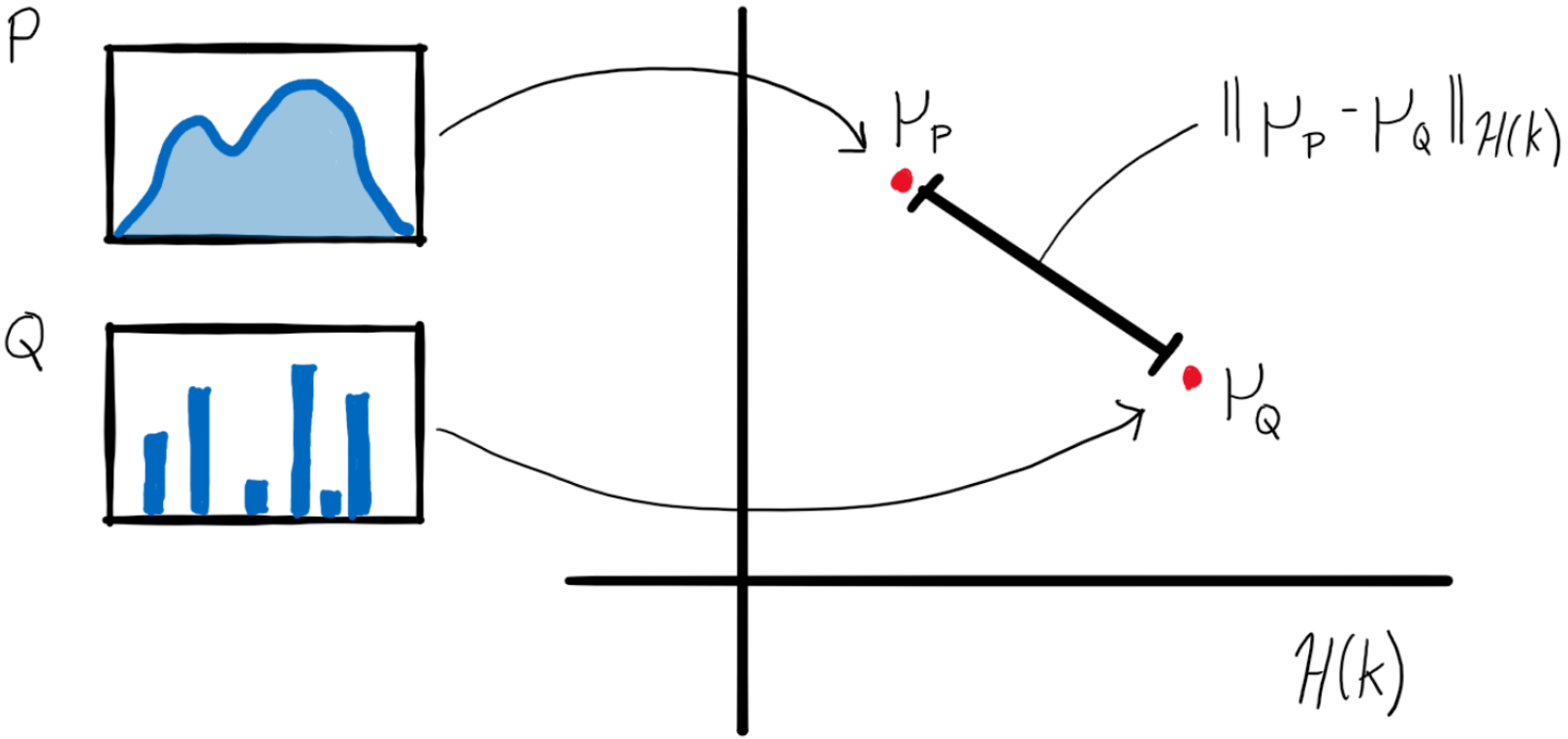


Figure – Plongement par moyenne des lois P et Q dans le RKHS $\mathcal{H}(k)$.

4. Oates, *Minimum Discrepancy Methods in Uncertainty Quantification*.

Choix visant l'uniformité

À l'itération i, avec $\mathbf{X}_i = \{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(i)}\}$, la loi discrète correspondante $\xi_i = \frac{1}{i}\sum_{j=1}^{i} \delta(\mathbf{x}^{(j)})$ et un noyau k :

$$\mathbf{x}^{(i+1)} \in \underset{\mathbf{x}\in\mathcal{S}\setminus\mathbf{X}_i}{\arg\min} \left(\int k(\mathbf{x},\mathbf{x}')d\xi_i(\mathbf{x}') - \int k(\mathbf{x},\mathbf{x}')d\mu(\mathbf{x}') \right) \tag{5}$$

Kernel herding [5]

$$\mathbf{x}^{(i+1)} \in \underset{\mathbf{x}\in\mathcal{S}\setminus\mathbf{X}_i}{\arg\min} \left(\frac{1}{i}\sum_{i=1}^{i} k(\mathbf{x},\mathbf{x}^{(i)}) - \frac{1}{N}\sum_{\mathbf{x}'\in\mathcal{S}} k(\mathbf{x},\mathbf{x}') \right) \tag{6}$$

Points supports gloutons [6] (noyau distance d'énergie)

$$\mathbf{x}^{(i+1)} \in \underset{\mathbf{x}\in\mathcal{S}\setminus\mathbf{X}_i}{\arg\min} \left(\frac{1}{N}\sum_{\mathbf{x}'\in\mathcal{S}} \|\mathbf{x}-\mathbf{x}'\| - \frac{1}{i+1}\sum_{i=1}^{n} \|\mathbf{x}-\mathbf{x}^{(i)}\| \right) \tag{7}$$

5. Y. Chen, M. Welling et A. Smola. "Super-samples from kernel herding". In : *Proc. of the 26th UAI Conference*. AUAI Press. 2010.
6. S. Mak et V.R. Joseph. "Support points". In : *Annals of Statistics* (2018).

Choix visant l'uniformité

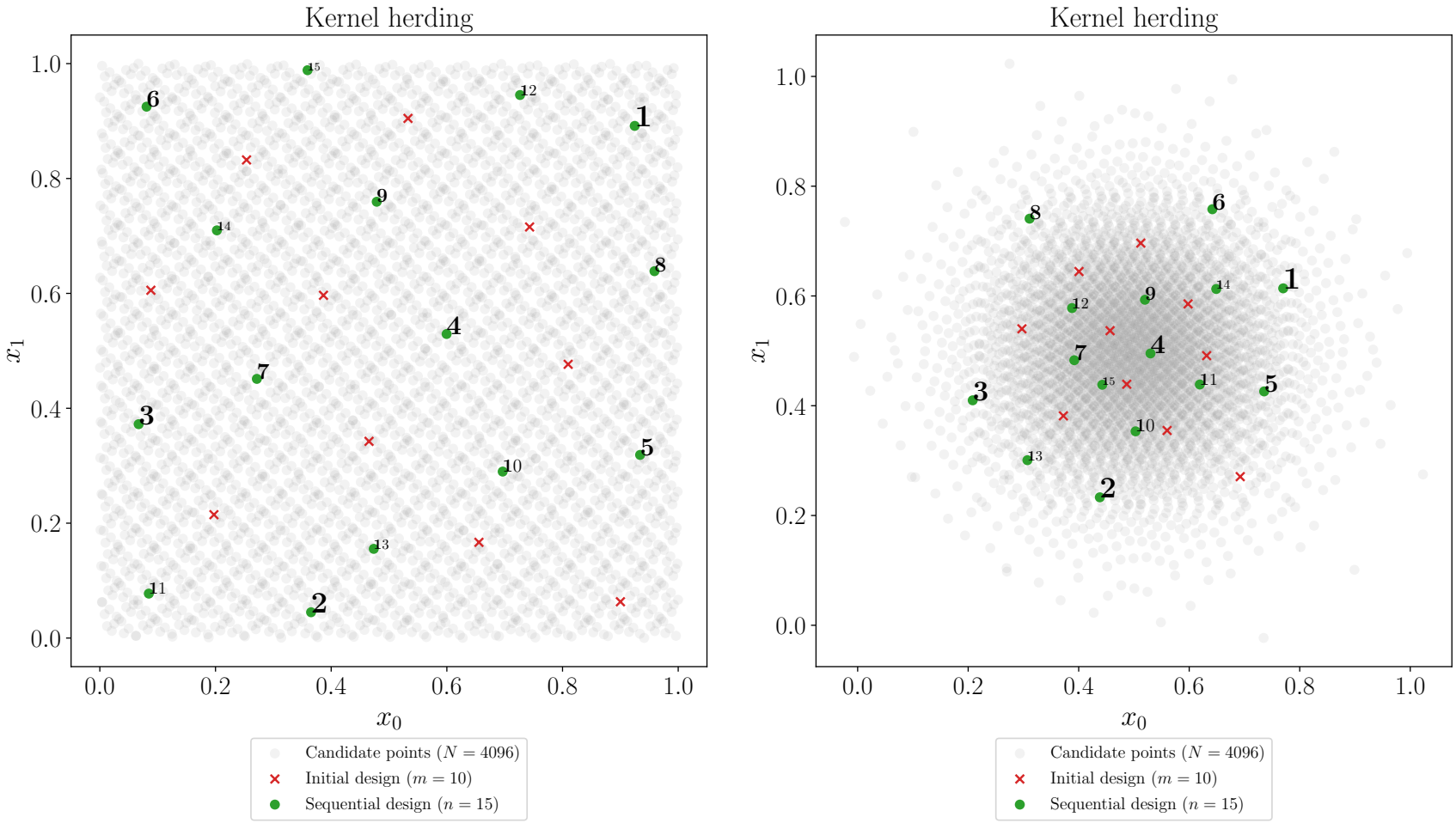


Figure – Ensembles test construits séquentiellement par kernel herding (loi uniforme et normale bi-variée)

Méthode disponible dans la librairie python `otkerneldesign`

Choix visant l'uniformité

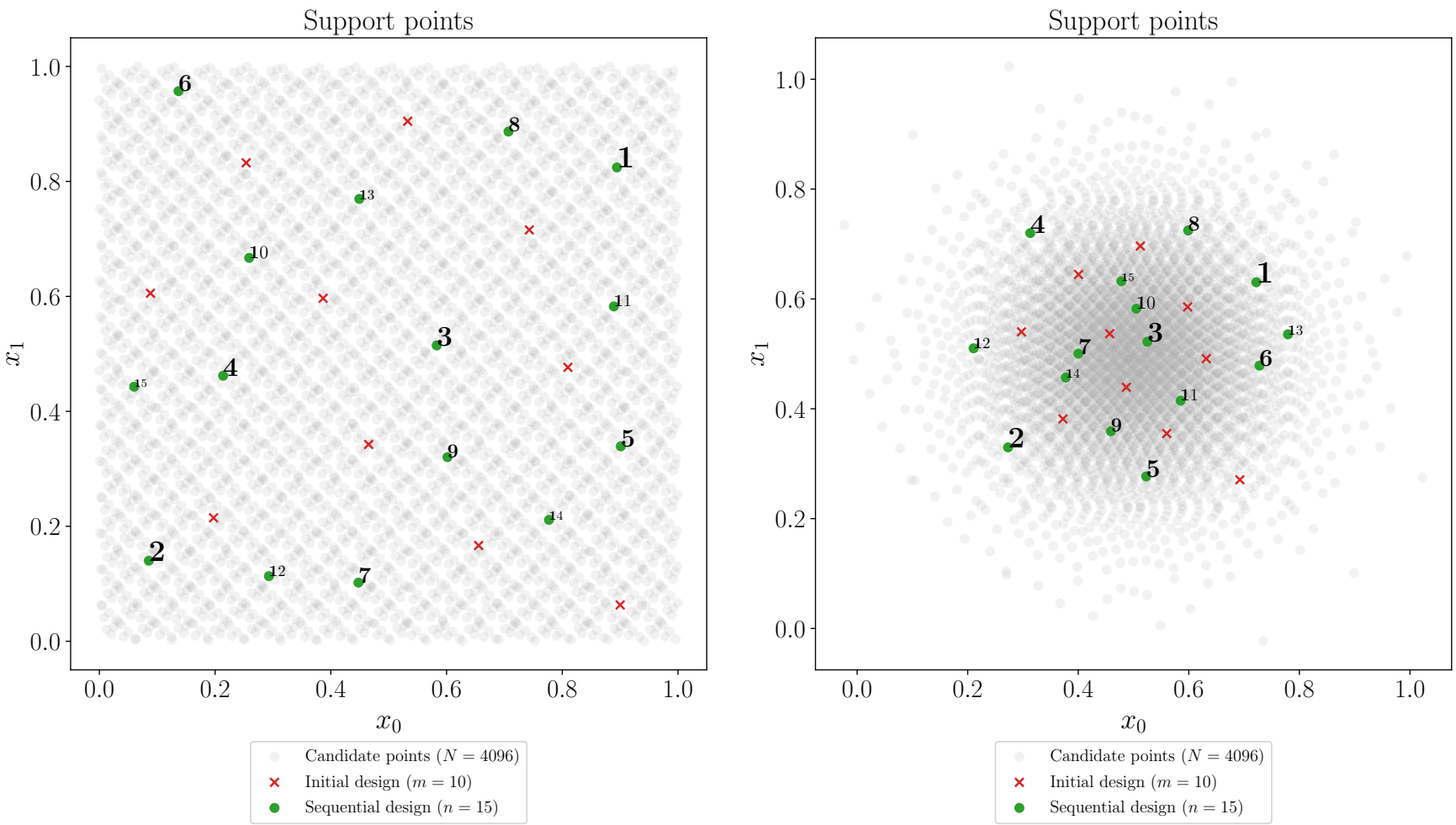


Figure – Ensembles test construits séquentiellement par points supports gloutons (loi uniforme et normale bi-variée)

Méthode disponible dans la librairie python `otkerneldesign`

KernelHerding

class **otkerneldesign.KernelHerding(*kernel=None, distribution=None, candidate_set_size=None, candidate_set=None, initial_design=None*)**

Incrementally select new design points with kernel herding.

Parameters: **kernel** : `openturns.CovarianceModel`

Covariance kernel used to define potentials. By default a product of Matern kernels with smoothness 5/2.

distribution : `openturns.Distribution`

Distribution the design points must represent. If not specified, then *candidate_set* must be specified instead. Even if *candidate_set* is specified, can be useful if it allows the use of analytical formulas.

candidate_set_size : ***positive int***

Size of the set of all candidate points. Unnecessary if *candidate_set* is specified. Otherwise, 2^{12} by default.

candidate_set : ***2-d list of float***

Large sample that empirically represents a distribution. If not specified, then *distribution* and *candidate_set_size* must be in order to generate it automatically.

initial_design : ***2-d list of float***

Sample of points that must be included in the design. Empty by default.

Examples

```
>>> import openturns as ot
>>> import otkerneldesign as otkd
>>> distribution = ot.ComposedDistribution([ot.Normal(0.5, 0.1)] * 2)
>>> dimension = distribution.getDimension()
>>> # Kernel definition
>>> ker_list = [ot.MaternModel([0.1], [1.0], 2.5)] * dimension
>>> kernel = ot.ProductCovarianceModel(ker_list)
>>> # Kernel herding design
>>> kh = otkd.KernelHerding(kernel=kernel, distribution=distribution)
>>> kh_design, _ = kh.select_design(size=20)
```

Previous topic
Index of classes
Next topic
KernelHerdingTensorized

This Page
Show Source

Quick search
Go

Au-delà des métriques de performance usuelles

Coefficient de prédictivité idéal du prédicteur η_m

$$Q^2_{\text{ideal}}(\mu) = 1 - \frac{\text{ISE}_\mu(\mathbf{X}_m, \mathbf{y}_m)}{\text{Var}_\mu(y(\mathbf{X}))} = 1 - \frac{\int_{\mathcal{X}} [y(\mathbf{x}) - \eta_m(\mathbf{x})]^2 \, \mathrm{d}\mu(\mathbf{x})}{\int_{\mathcal{X}} [y(\mathbf{x}) - \int_{\mathcal{X}} y(\mathbf{x}') d\mu(\mathbf{x}')]^2 \, \mathrm{d}\mu(\mathbf{x})}. \tag{8}$$

Coefficient de prédictivité : estimateur arithmétique

$$\widehat{Q}^2_n = 1 - \frac{\text{ISE}_{\xi_n}(\mathbf{X}_m, \mathbf{y}_m)}{\text{Var}_{\xi_n}(y(\mathbf{X}))} = 1 - \frac{\sum_{i=1}^n \left[y(\mathbf{x}^{(i)}) - \eta_m(\mathbf{x}^{(i)}) \right]^2}{\sum_{i=1}^n \left[y(\mathbf{x}^{(i)}) - \overline{y}_n \right]^2}, \tag{9}$$

où $\xi_n = \frac{1}{n} \sum_{i=1}^n \delta(\mathbf{x}^{(i)})$, $\quad \overline{y}_n = \frac{1}{n} \sum_{i=1}^n y(\mathbf{x}^{(i)})$.

- Cet n'estimateur n'utilise pas l'ensemble d'apprentissage pour estimer la variance.
- Une pondération intelligente de l'ISE (erreur quadratique intégrée) pourrait améliorer l'estimation.

Au-delà des métriques de performance usuelles

Soit le processus d'erreur $\delta_m(\mathbf{x}) = y(\mathbf{x}) - \eta_m(\mathbf{x}) \sim \mathrm{GP}(0, \sigma^2 K_{|m})$.
Exprimons l'erreur quadratique de l'ISE à l'aide de ξ_n :

$$\begin{aligned}\overline{\Delta}^2(\xi_n, \mu; \mathbf{X}_m, \mathbf{y}_m) &= \mathbb{E}\left[\left(\mathrm{ISE}_{\xi_n}(\mathbf{X}_m, \mathbf{y}_m) - \mathrm{ISE}_{\mu}(\mathbf{X}_m, \mathbf{y}_m)\right)^2\right], \\ &= \mathbb{E}\left[\left(\int_{\mathcal{X}} \delta_m^2(\mathbf{x})\, \mathrm{d}(\xi_n - \mu)(\mathbf{x})\right)^2\right], \\ &= \sigma^2\, \mathrm{MMD}^2_{\overline{K}_{|m}}(\xi_n, \mu)\,. \end{aligned} \tag{10}$$

Ici $\overline{K}_{|m}$ est défini comme $\overline{K}_{|m}(\mathbf{x}, \mathbf{x}') = 2\, K^2_{|m}(\mathbf{x}, \mathbf{x}') + K_{|m}(\mathbf{x}, \mathbf{x}) K_{|m}(\mathbf{x}', \mathbf{x}')$.
Avec $\xi_n = \sum_{i=1}^n w_i \delta(\mathbf{x}^{(i)})$ [en cas d'uniformité $N^{-1} \sum_{i=1}^n \delta(\mathbf{x}^{(i)})$], l'idée est de trouver les poids $\mathbf{w}_n^* = (w_i^*)_{i=1}^n$ minimisant (10). Par calcul direct :

$$\mathbf{w}_n^* = \overline{\mathbf{K}}_{|m}^{-1}(\mathbf{X}_n)\mathbf{p}_{\overline{K}_{|m},\mu}(\mathbf{X}_n)\,,$$

avec $\mathbf{p}_{\overline{K}_{|m},\mu}(\mathbf{X}_n) = \left[\int \overline{K}_{|m}(\mathbf{x}^{(1)}, \mathbf{x})\, \mathrm{d}\mu(\mathbf{x}), \ldots, \int \overline{K}_{|m}(\mathbf{x}^{(n)}, \mathbf{x})\, \mathrm{d}\mu(\mathbf{x})\right]^\top$.

Au-delà des métriques de performance usuelles

Coefficient de prédictivité : estimateur avec pondération optimale [7]

$$Q^2_{n*} = 1 - \frac{\sum_{i=1}^{n} w_i^* \left[y(\mathbf{x}^{(i)}) - \eta_m(\mathbf{x}^{(i)}) \right]^2}{\frac{1}{n} \sum_{i=1}^{n} \left[y(\mathbf{x}^{(i)}) - \overline{y}_n \right]^2} . \tag{11}$$

- Les poids w_i^* ne dépendent pas du paramètre de variance du processus gaussien σ^2.
- On pourrait imaginer une pondération pour le dénominateur également.

7. E. Fekhari et al. "Model predictivity assessment : incremental test-set selection and accuracy evaluation". In : *Preprint* (2021).

Benchmark analytique

Problèmes intégrés au benchmark :

- Une fonction analytique.
- La variable d'entrée est aléatoire.
- Un ensemble d'apprentissage de taille m construit par LHS optimisé (3 tailles correspondant à un métamodèle de krigeage mauvais/bon/très bon)
- Une valeur de référence calculée pour chaque métamodèle sur un grand ensemble de test Monte-Carlo.

On compare différentes tailles d'ensemble test, méthodes de construction de l'ensemble et estimateurs du Q^2.

Benchmark analytique

Problèmes intégrés au benchmark :

- Une fonction analytique.
- La variable d'entrée est aléatoire.
- Un ensemble d'apprentissage de taille m construit par LHS optimisé (3 tailles correspondant à un métamodèle de krigeage mauvais/bon/très bon)
- Une valeur de référence calculée pour chaque métamodèle sur un grand ensemble de test Monte-Carlo.

On compare différentes tailles d'ensemble test, méthodes de construction de l'ensemble et estimateurs du Q^2.

Cas test analytique numéro 3 (« g-sobol » en dimension 8) :
La loi μ est uniforme sur $\mathcal{X} = [0,1]^8$ et $m \in \{15, 30, 100\}$.

$$f_3(\mathbf{x}) = \prod_{i=1}^{8} \frac{|4x_i - 2| + a_i}{1 + a_i}, \quad a_i = i^2.$$

Benchmark analytique

Cas tests analytiques numéros 1 et 2 (dimension 2) où $\mathbf{x} \in \mathcal{X} = [0,1]^2$

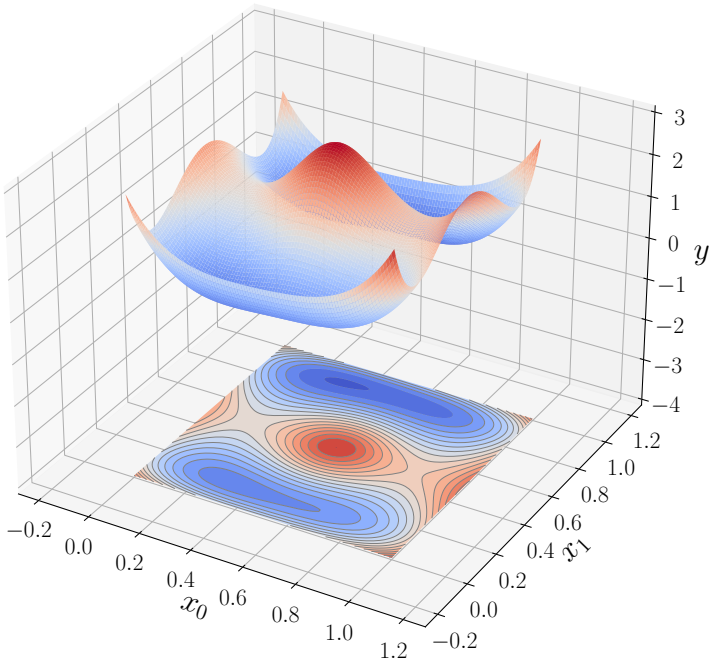


Figure – $f_1(\mathbf{x})$ dans le cas test numéro 1 ; μ est uniforme ; $m \in \{8, 15, 30\}$

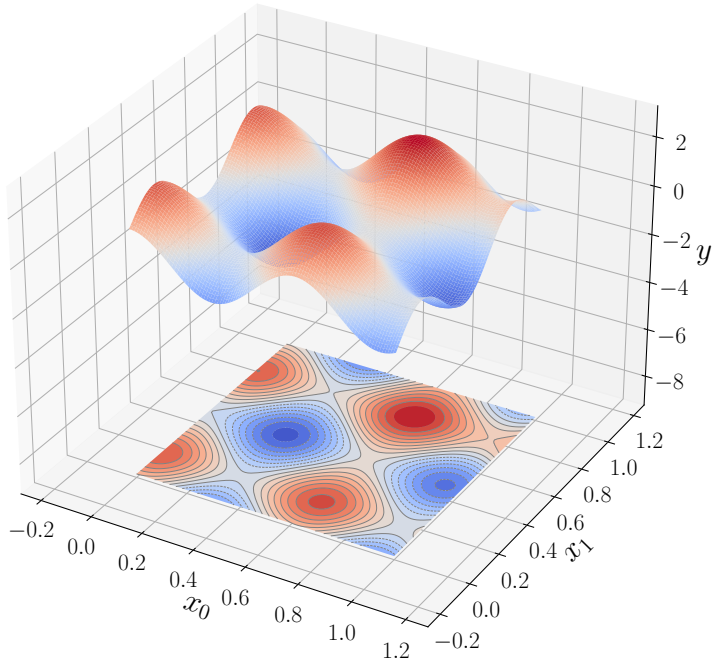
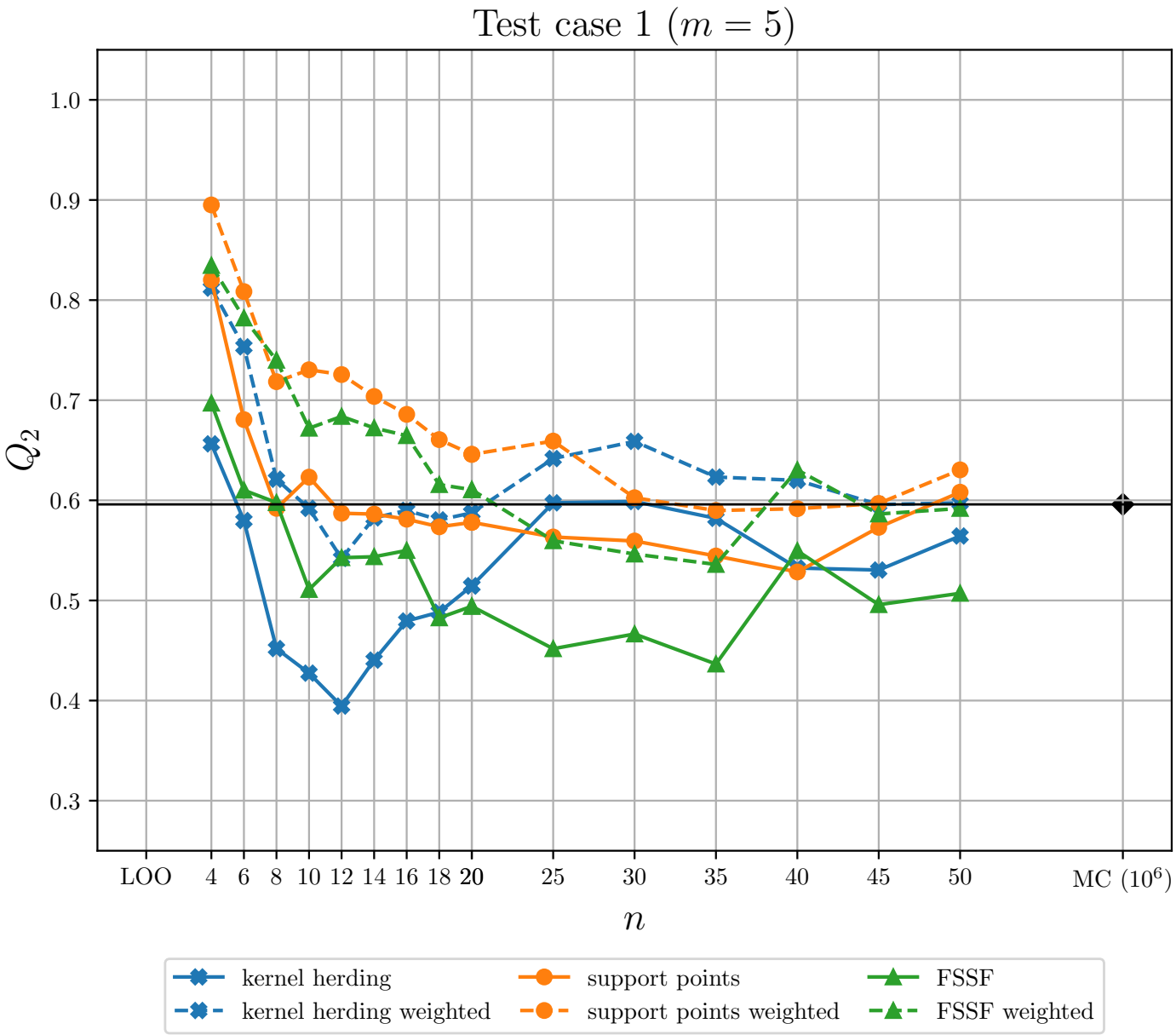
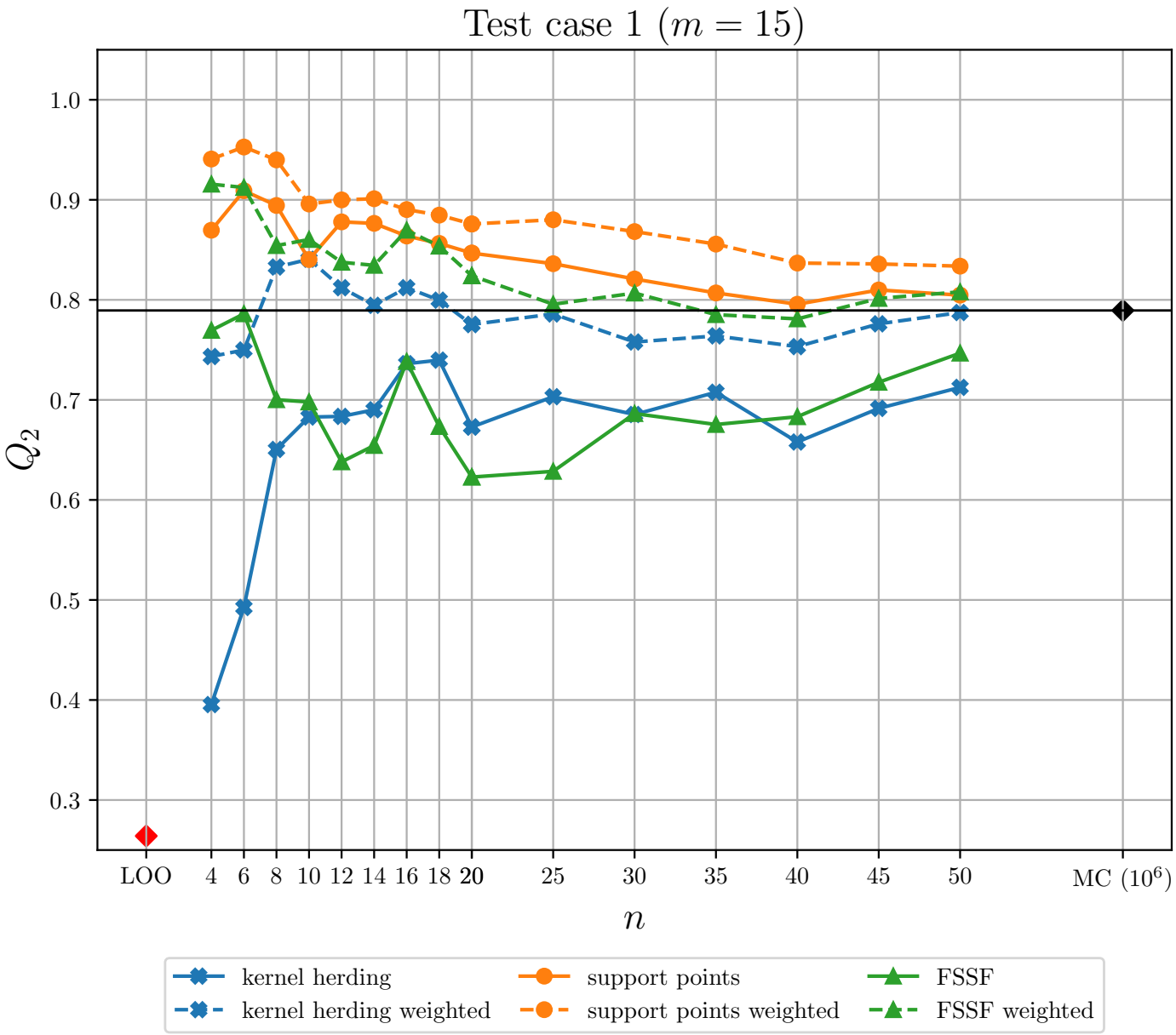


Figure – $f_2(\mathbf{x})$ dans le cas test numéro 2 ; μ est normale standard ; $m \in \{5, 15, 30\}$

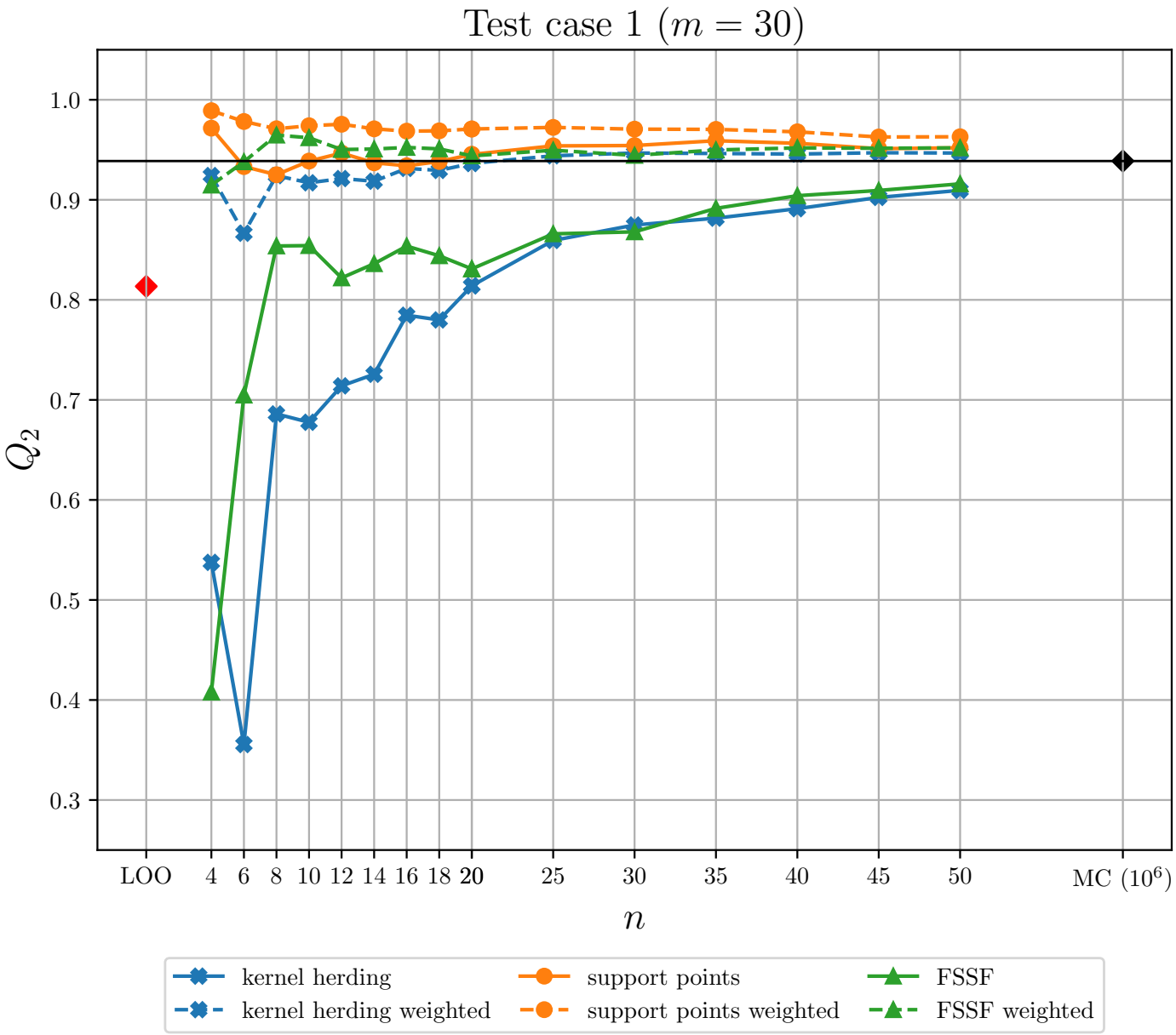
Coefficient de prédictivité d'un mauvais modèle



Coefficient de prédictivité d'un bon modèle



Coefficient de prédictivité d'un très bon modèle



Résultats du benchmark analytique

Analyse et interprétation :

- L'ensemble test doit simultanément compléter l'ensemble d'apprentissage et imiter la loi visée.
- Les points support et le kernel herding donnent en général les meilleurs résultats.
- Le résultat du kernel herding est sensible au choix de noyau.
- Toutes les méthodes d'échantillonnage sont exposées au fléau de la dimension.
- Pondérer l'ensemble de test est de bon aloi en raison de son écart à l'ensemble d'entraînement.
- La validation croisée leave-one-out sous-estime en général le Q_2, surtout quand m est petit.
- Après le test, le modèle peut éventuellement être amélioré en complétant l'apprentissage par l'ensemble test.

Bibliographie

[1] Y. Chen, M. Welling et A. Smola. "Super-samples from kernel herding". In : *Proc. of the 26th UAI Conference*. AUAI Press. 2010.

[2] E. Fekhari et al. "Model predictivity assessment : incremental test-set selection and accuracy evaluation". In : *Preprint* (2021).

[3] Tadayoshi Fushiki. "Estimation of prediction error by using K-fold cross-validation". In : *Statistics and Computing* 21.2 (2011), p. 137-146.

[4] S. Mak et V.R. Joseph. "Support points". In : *Annals of Statistics* (2018).

[5] C.J. Oates. *Minimum Discrepancy Methods in Uncertainty Quantification*. Lecture Notes at ETICS Summer School. 2021.

[6] B. Shang et D. Apley. "Fully-sequential space-filling design algorithms for computer experiments". In : *Journal of Quality Technology* 53 (2020), p. 1-24.