Chaos polynomial creux basé sur la méthode LAR

Géraud Blatman

EDF R&D, Département Matériaux et Mécanique des Composants

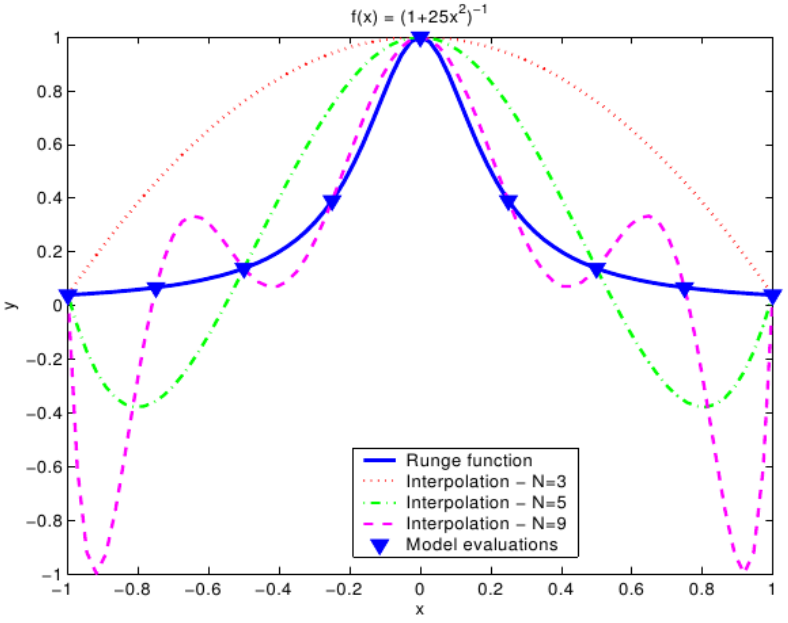
Journée Utilisateurs Open TURNS – 7 juin 2011

CHANGER L'ÉNERGIE ENSEMBLE

Retour sur les stratégies adaptatives dans Open TURNS

Sequential Strategy. Elle repose essentiellement sur l'erreur d'apprentissage, i.e. l'erreur aux points du plan d'expérience.

Or, pour un plan donné de taille *N*, celle-ci ne peut que diminuer lorsque l'on enrichit la base du chaos. Elle se révèle ainsi insensible à un éventuel surajustement aux données (surapprentissage).



Surapprentissage : erreur d'approximation nulle aux points du plan d'expérience mais importante entre ces points

Retour sur les stratégies adaptatives dans Open TURNS

Cleaning Strategy. Elle fait intervenir plusieurs paramètres d'algorithme, pas toujours évidents à régler par l'utilisateur :

- Le paramètre de seuillage des coefficients
- Le nombre de coefficients non nuls souhaité
- Le nombre maximum de termes à explorer dans la base

Le choix des 2 derniers paramètres est d'autant plus délicat qu'il dépend de la taille du plan d'expériences.

Caractéristiques de la nouvelle approche

- Construction d'un chaos polynomial creux, i.e. ne comportant que peu de coefficients non nuls

- Utilisation d'une algorithmie ne faisant intervenir que peu de paramètres

- Réglage du compromis précision – complexité du métamodèle au moyen d'un estimateur d'erreur sensible au surapprentissage

Plan

1. Régression régularisée et algorithme du LAR
2. Estimation de l'erreur d'approximation
3. LAR avec enrichissement automatique de la base du chaos

Position du problème

On considère ici un modèle à réponse scalaire :

$$\mathsf{Y} = \mathcal{M}(\mathbf{X}) \qquad \dim(\mathbf{X}) = M \quad , \quad \dim(\mathsf{Y}) = 1$$

Disposant d'un *N*-échantillon d'entrées et de sorties du modèle $\left\{ \left(\boldsymbol{x}^{(i)}, y^{(i)} = \mathcal{M}(\boldsymbol{x}^{(i)}) \right) \ , \ i = 1, \ldots, N \right\}$, on souhaite approcher la réponse par le chaos polynomial suivant :

$$\mathsf{Y}_{\Lambda_c} = \mathcal{M}_{\Lambda_c}(\boldsymbol{a}; \mathbf{X}) = \sum_{\boldsymbol{\alpha} \in \Lambda_c} a_{\boldsymbol{\alpha}} \, \phi_{\boldsymbol{\alpha}}(\mathbf{X})$$

où : $\boldsymbol{a} = \left(a_{\boldsymbol{\alpha}_0}, \ldots, a_{\boldsymbol{\alpha}_{P-1}} \right)^{\mathsf{T}} \qquad P = \mathrm{card}(\Lambda_c)$

L'ensemble Λ_c est un sous-ensemble fini et non vide de $\mathbb{N}^M$, qui correspond à la base candidate du chaos.

Moindres carrés régularisés

Moindres carrés ordinaires

$$\widehat{\boldsymbol{a}} = \arg\min_{\boldsymbol{a}\in\mathbb{R}^P} \left\{ \sum_{i=1}^{N} \left(y^{(i)} - \mathcal{M}_{\Lambda_c}(\boldsymbol{a}; \boldsymbol{x}^{(i)}) \right)^2 \right\}$$

- Non résoluble si $N < P$
- Danger de surapprentissage si N est trop proche de P

Moindres carrés régularisés

$$\widehat{\boldsymbol{a}} = \arg\min_{\boldsymbol{a}\in\mathbb{R}^P} \left\{ \sum_{i=1}^{N} \left(y^{(i)} - \mathcal{M}_{\Lambda_c}(\boldsymbol{a}; \boldsymbol{x}^{(i)}) \right)^2 + C\,\Omega(\mathcal{M}_{\Lambda_c}) \right\}$$

$\Omega(\mathcal{M}_{\Lambda_c})$: mesure la régularité de la solution

C : règle le compromis attache aux données - complexité

Formulation LASSO

On choisit de pénaliser la somme des valeurs absolues des coefficients :

$$\Omega(\mathcal{M}_{\Lambda_c}) \;=\; \|\boldsymbol{a}\|_1 \;=\; \sum_{\boldsymbol{\alpha}\in\Lambda_c} |a_{\boldsymbol{\alpha}}|$$

Cette formulation est appelée LASSO (*Least Absolute Shrinkage and Selection Operator*) [Tishirani, 1996].

Le choix de la régularisation L1 garantit le seuillage de plusieurs coefficients, i.e. une solution creuse. Cette parcimonie des coefficients est d'autant plus marquée que le paramètre *C* est grand.

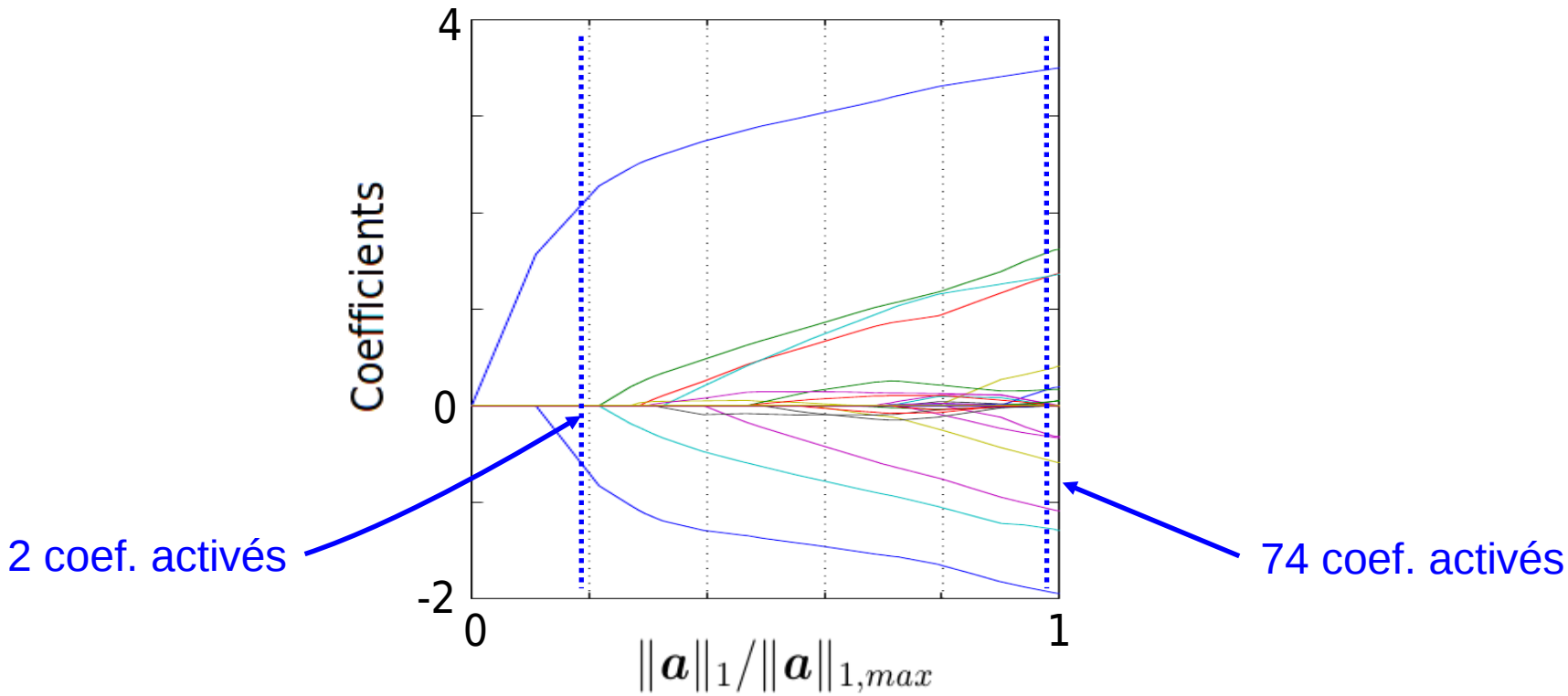
Le graphe de l'évolution des valeurs des estimations des coefficients en faisant varier *C* de 0 à une valeur maximale est appelé chemin de régularisation.

Chemin de régularisation

Exemple. Fonction d'Ishigami. Soit $X_1, X_2, X_3 \sim \mathcal{U}([-\pi, \pi])$

et $Y = \sin X_1 + 7 \sin^2 X_2 + 0,1 X_3^4 \sin X_1$

On veut approximer la réponse par un chaos de Legendre de degré 10 (P=286 termes) à partir d'un plan quasi-aléatoire de taille N=75.



Implémentation du LASSO

Pour une valeur de *C* donnée, on résoud le problème du LASSO via une procédure de programmation quadratique.

Le choix d'une valeur optimale pour *C* amène à résoudre les 2 points suivants :

- Effectuer un grand nombre de calculs LASSO pour plusieurs valeurs de *C* : peut se révéler coûteux
- Evaluer la précision de chacune des solutions au moyen d'un estimateur d'erreur pertinent

Least Angle Regression (LAR)

LAR est une méthode itérative de sélection de variables [Efron *et al.*, 2004].

Elle fournit un ensemble de solutions au problème du LASSO, en partant de la solution la plus creuse (i.e. égale à 0), puis en ajoutant successivement des termes dans la base du chaos.

Précisément, LAR équivaut à résoudre une succession de problèmes LASSO :

$$\widehat{\boldsymbol{a}}^{(k)} = \arg\min_{\boldsymbol{a}\in\mathbb{R}^P} \left\{ \sum_{i=1}^{N} \left(y^{(i)} - \mathcal{M}_{\Lambda_c}(\boldsymbol{a}; \boldsymbol{x}^{(i)}) \right)^2 + C_k \, \|\boldsymbol{a}\|_1 \right\}$$

pour des valeurs décroissantes de C : $C_0 > C_1 > \cdots > C_m$

Le nombre de composantes non nulles de $\widehat{\boldsymbol{a}}^{(k)}$ est égal à k.

Illustration (1) (Guigue *et al.*, 2006)

On considère une réponse vivant dans un espace de dimension M=3.

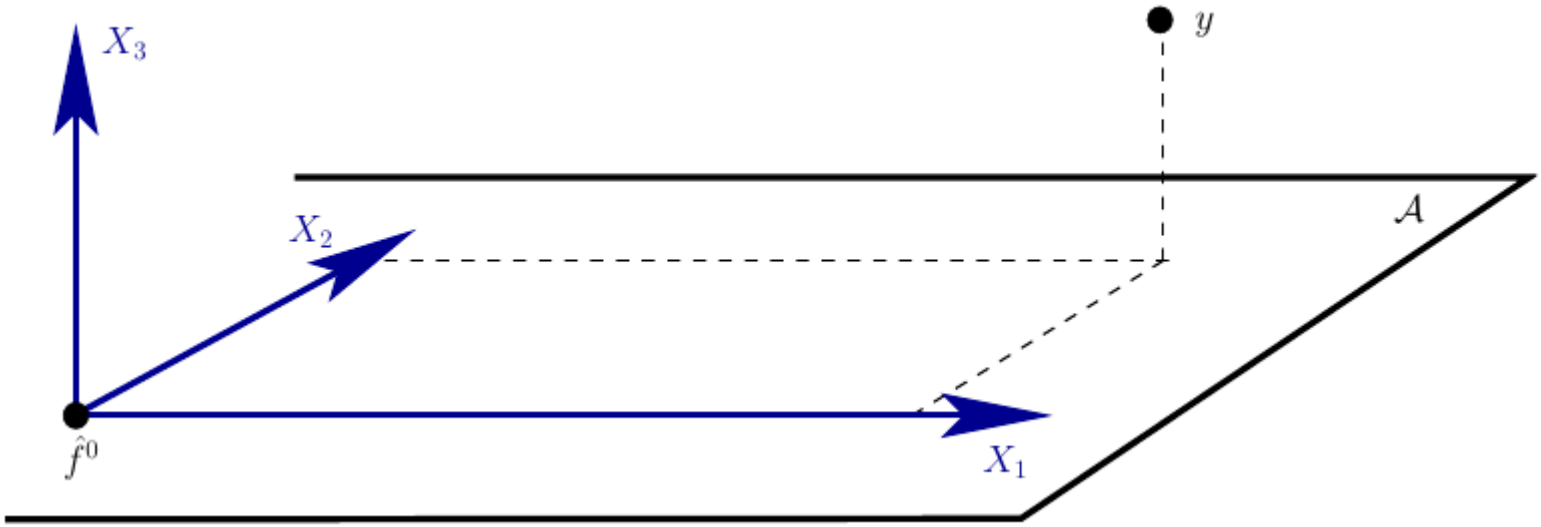


Illustration (2) (Guigue *et al.*, 2006)

Par le biais de projections successives du résidu $R = Y$, on identifie la variable la plus corrélée à celui-ci, à savoir X_1 : elle est ajoutée à l'ensemble Λ . On détermine la direction de descente ***u*** dans l'espace engendré par les variables de Λ .

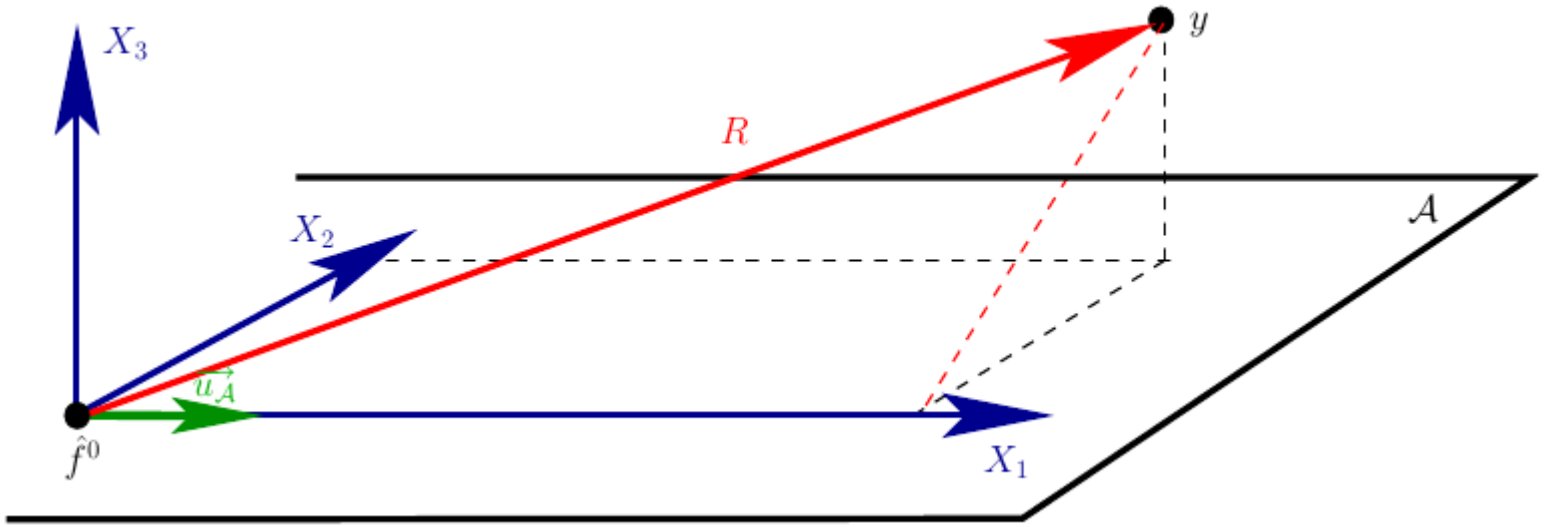


Illustration (3) (Guigue *et al.*, 2006)

Définition d'une solution à 1 variable (ici X_1) en progressant d'un pas γ dans la direction ***u***. Le pas est défini de sorte à ce qu'il y ait une équi-corrélation du résidu avec les variables actives dans Λ et une nouvelle variable (ici X_2).

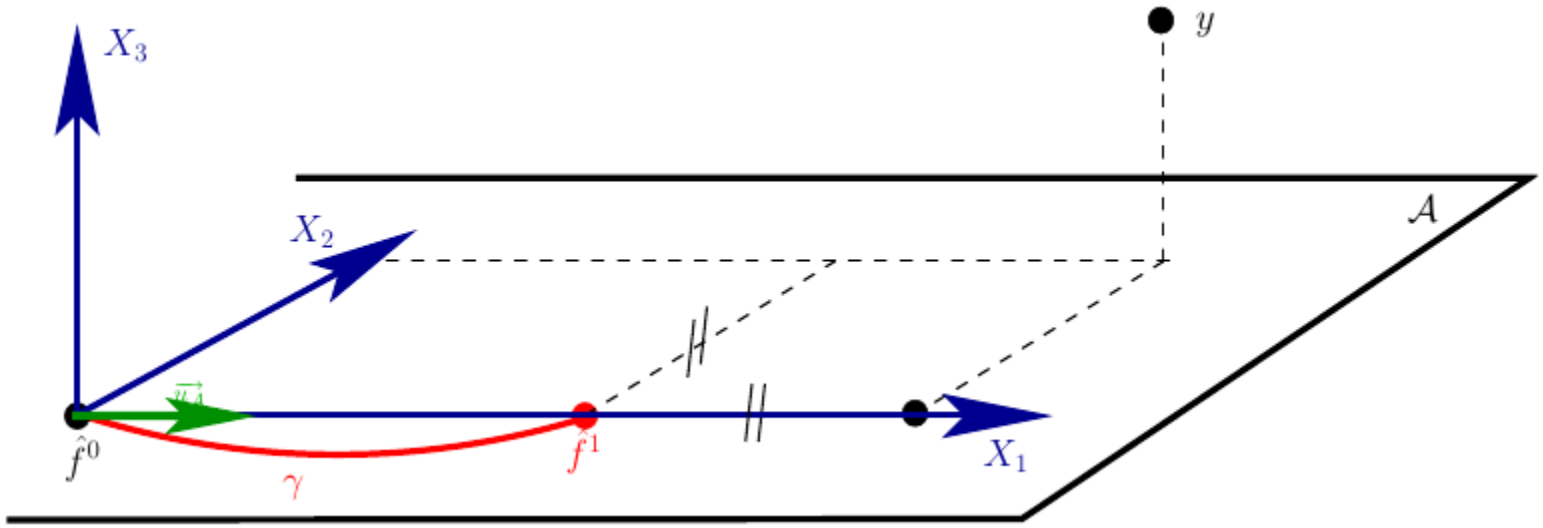


Illustration (4) (Guigue *et al.*, 2006)

On détermine la nouvelle direction ***u*** dans l'espace engendré par (X_1, X_2).

On remarque que le résidu est bien équi-corrélé avec toutes les variables actives. Λ

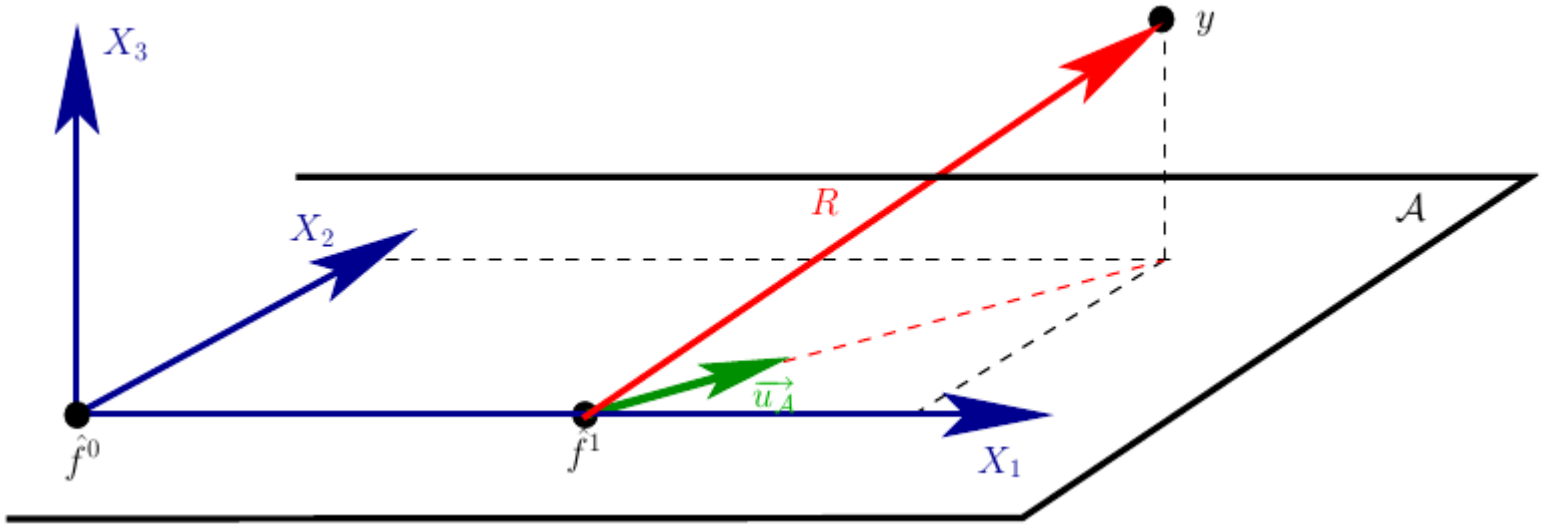
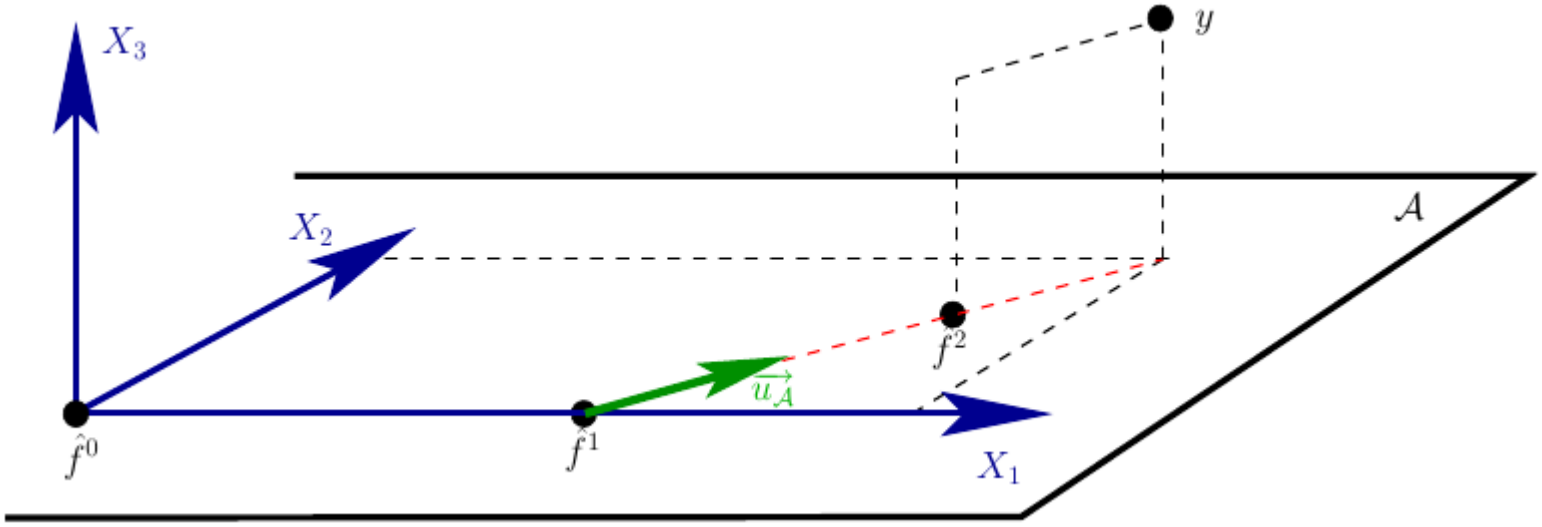


Illustration (5) (Guigue *et al.*, 2006)

Définition d'une solution à 2 variables (ici X_1 et X_2) en progressant d'un pas qui assure l'équi-corrélation avec X_3.



LAR hybride

Afin d'améliorer la précision des solutions obtenues par LAR, on recalcule les coefficients par moindres carrés ordinaires.

Autrement dit LAR n'est utilisé que comme une méthode de sélection de termes dans la base du chaos. Ensuite une projection L2 est effectuée sur cette famille de fonctions.

2. Estimation de l'erreur d'approximation

Sélection d'une solution optimale

On a vu que LAR retourne toute une collection de chaos polynomiaux de moins en moins creux.

Or, en pratique, on ne souhaite conserver qu'une seule de ces solutions, celle qui se révèle optimale vis-à-vis d'un certain critère.

On pourrait par exemple retenir la solution associée à l'erreur empirique la plus faible, mais on a vu que cette quantité était insensible au surapprentissage.

Validation croisée de type *leave-one-out*

On considère une approximation par chaos $\mathcal{M}_\Lambda(\mathbf{X}) = \sum_{\boldsymbol{\alpha}\in\Lambda} a_{\boldsymbol{\alpha}}\,\phi_{\boldsymbol{\alpha}}(\mathbf{X})$

construite à partir d'un plan d'expériences $\mathcal{X} = \{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\}$.

On retire le *i*-ème point du plan d'expérience : $\mathcal{X}_{-i} = \mathcal{X} - \{\boldsymbol{x}^{(i)}\}$,

et on construit le chaos polynomial $\mathcal{M}_\Lambda^{(-i)}(\mathbf{X})$ à partir de $\mathcal{X}_{-i}$.

On calcule le résidu prédit au point $\boldsymbol{x}^{(i)}$:

$$\Delta^{(i)} = \mathcal{M}(\boldsymbol{x}^{(i)}) - \mathcal{M}_\Lambda^{(-i)}(\boldsymbol{x}^{(i)})$$

On réitère pour tous les points du plan, puis on évalue l'erreur moyenne :

$$\varepsilon_{LOO} = \left(\frac{1}{N}\sum_{i=1}^{N} {\Delta^{(i)}}^2\right) \Big/ \widehat{\mathbb{V}\mathrm{ar}}[\mathcal{Y}]$$

Validation croisée de type *leave-one-out*

Calculer l'erreur *leave-one-out* conduit donc à effectuer *N* calculs de régression.

Toutefois, dans notre contexte de régression linéaire généralisée, celle-ci peut s'obtenir également sans effectuer de régression supplémentaire :

$$\varepsilon_{LOO} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{\mathcal{M}(\boldsymbol{x}^{(i)}) - \mathcal{M}_{\Lambda}(\boldsymbol{x}^{(i)})}{1 - \mathrm{diag}_i\left(\boldsymbol{\Phi}(\boldsymbol{\Phi}^{\mathsf{T}}\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^{\mathsf{T}}\right)}\right)^2 / \widehat{\mathbb{V}\mathrm{ar}}[\mathcal{Y}]$$

où : $$\boldsymbol{\Phi} = \begin{pmatrix} \phi_{\boldsymbol{\alpha}_0}(\boldsymbol{x}^{(1)}) & \cdots & \phi_{\boldsymbol{\alpha}_{P-1}}(\boldsymbol{x}^{(1)}) \\ \vdots & \ddots & \vdots \\ \phi_{\boldsymbol{\alpha}_0}(\boldsymbol{x}^{(N)}) & \cdots & \phi_{\boldsymbol{\alpha}_{P-1}}(\boldsymbol{x}^{(N)}) \end{pmatrix}$$

Erreur *leave-one-out* corrigée

Afin de rendre cet estimateur d'erreur plus conservatif, on le multiplie par un facteur de correction $c(P,N)$. L'erreur corrigée est notée :

$$\varepsilon^* = c(P,N)\,\varepsilon$$

Exemple (ajustement du R^2) : $$c(P,N) = \frac{N-1}{N-P-1}$$

Correction de Chapelle & Vapnik (2002) :

$$c(P,N) = \frac{N}{N-P}\left(1 + \frac{\mathrm{tr}(\boldsymbol{S}^{-1})}{N}\right) \qquad \boldsymbol{S} = \frac{1}{N}\boldsymbol{\Phi}^{\mathsf{T}}\boldsymbol{\Phi}$$

Construction d'un chaos creux par LAR

1. On se donne un *N*-échantillon d'entrées et de sorties du modèle :

$$\left\{\left(\boldsymbol{x}^{(i)}, y^{(i)} = \mathcal{M}(\boldsymbol{x}^{(i)})\right) \quad , \quad i = 1, \ldots, N\right\}$$

2. On choisit une base candidate de polynômes Λ_c .

3. On applique la procédure LAR hybride ; on obtient ainsi une collection de chaos de moins en moins creux $\{\mathcal{M}_{\Lambda_1}, \ldots, \mathcal{M}_{\Lambda_m}\}$.

4. On évalue la précision de chacune des solutions en calculant leurs erreurs *leave-one-out* corrigées : $\{\varepsilon^*_{LOO,1}, \ldots, \varepsilon^*_{LOO,m}\}$.

5. On sélectionne la solution associée à la plus faible erreur.

Conclusion

La procédure LAR permet de construire un chaos polynomial creux à partir d'un ensemble d'apprentissage donné.

Elle rend notamment possible le traitement de problèmes de moindres carrés sous-déterminés, i.e. pour lesquels $N<P$.

Limitations :

1. La procédure requiert le choix *a priori* d'une base candidate Λ_c .
   - Si cette base est trop « petite », alors certains termes importants ne seront pas intégrés.
   - Si elle est trop «grande», alors certains termes non importants pourront être ajoutés par erreur, parasitant la solution.
2. Choix *a priori* de la taille de l'échantillon.

3. LAR avec enrichissement automatique de la base du chaos

Vers une approche adaptative

Idée : appliquer LAR sur un ensemble de bases candidates de plus en plus grandes $\Lambda_{c1} \subset \Lambda_{c2} \subset \cdots \subset \Lambda_{ck} \subset \cdots$ de sorte à pouvoir explorer exhaustivement l'ensemble $\mathbb{N}^M$, c'est-à-dire :

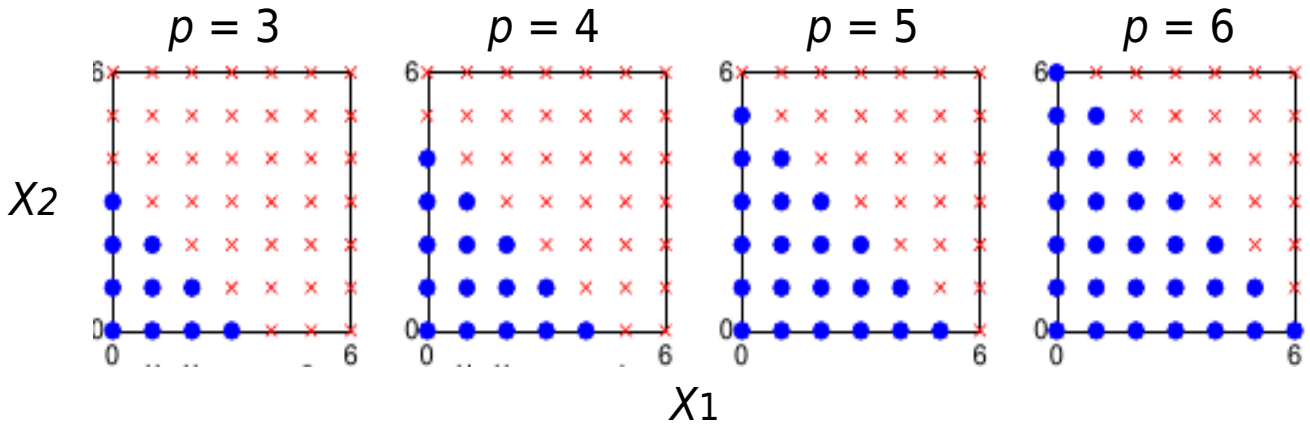
$$\lim_{k \to \infty} \Lambda_{ck} = \mathbb{N}^M$$

Objectif : trouver une suite pertinente de bases candidates. Une telle suite sera associée à une stratégie d'énumération appropriée de la base.

Pilotage par le degré total des polynômes

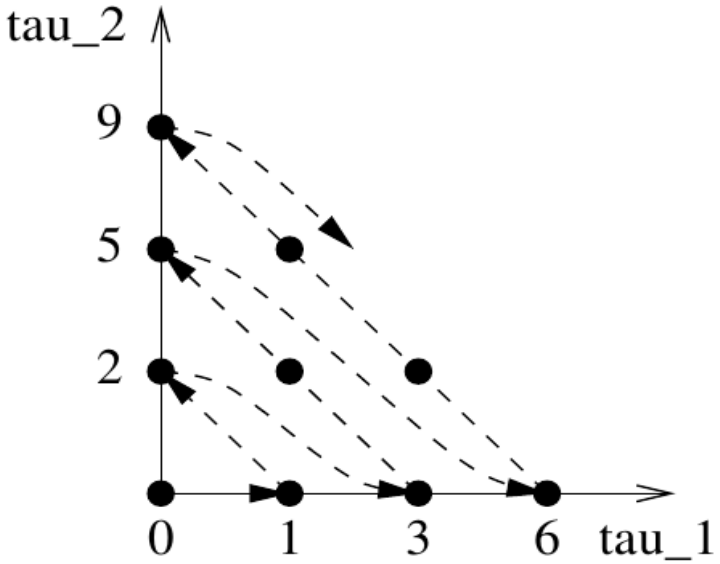
Suite de bases candidates : Polynômes dont le degré total est inférieur ou égal à $k \in \mathbb{N}^*$:

$$\Lambda_k = \left\{ \boldsymbol{\alpha} \in \mathbb{N}^M : \sum_{i=1}^{M} \alpha_i \leq k \right\}$$



Pilotage par le degré total des polynômes

Stratégie d'énumération de la base : stratégie linéaire



Pilotage par le degré total des polynômes

Limitation : la taille des bases augmente rapidement, d'autant plus que le nombre *M* de paramètres d'entrée est élevé.

En effet, on rappelle que le nombre d'éléments de chacune des bases est donné par :

$$\mathrm{card}(\Lambda_k) \;=\; P \;=\; \frac{(M+k)!}{M!\;k!}$$

Ainsi, LAR peut avoir tendance à intégrer à tort certains éléments de la base en réalité non significatifs.

Remède possible : définir des bases candidates de cardinalité plus faible, en privilégiant les effets principaux au détriment des interactions complexes (principe de hiérarchie des effets).

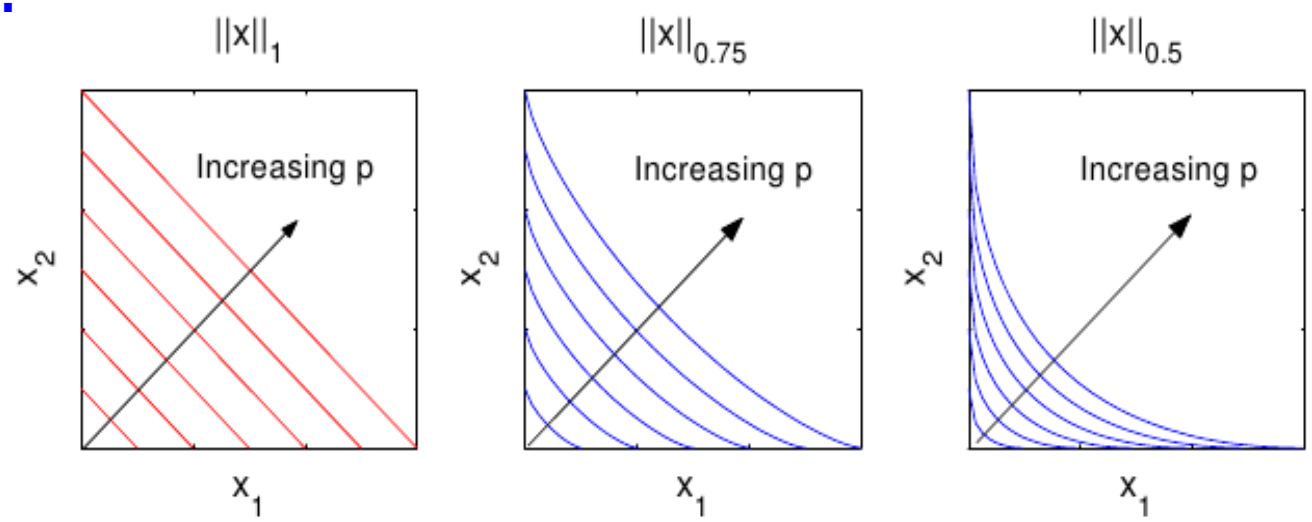
Pilotage par norme hyperbolique croissante

(Blatman & Sudret, 2009)

Quasi-norme hyperbolique :

$$\|\boldsymbol{\alpha}\|_q = \left(\sum_{i=1}^{M} \alpha_i^q \right)^{1/q} \quad , \quad 0 < q \leq 1$$

Exemples de suites de bases candidates pour plusieurs valeurs de *q* :



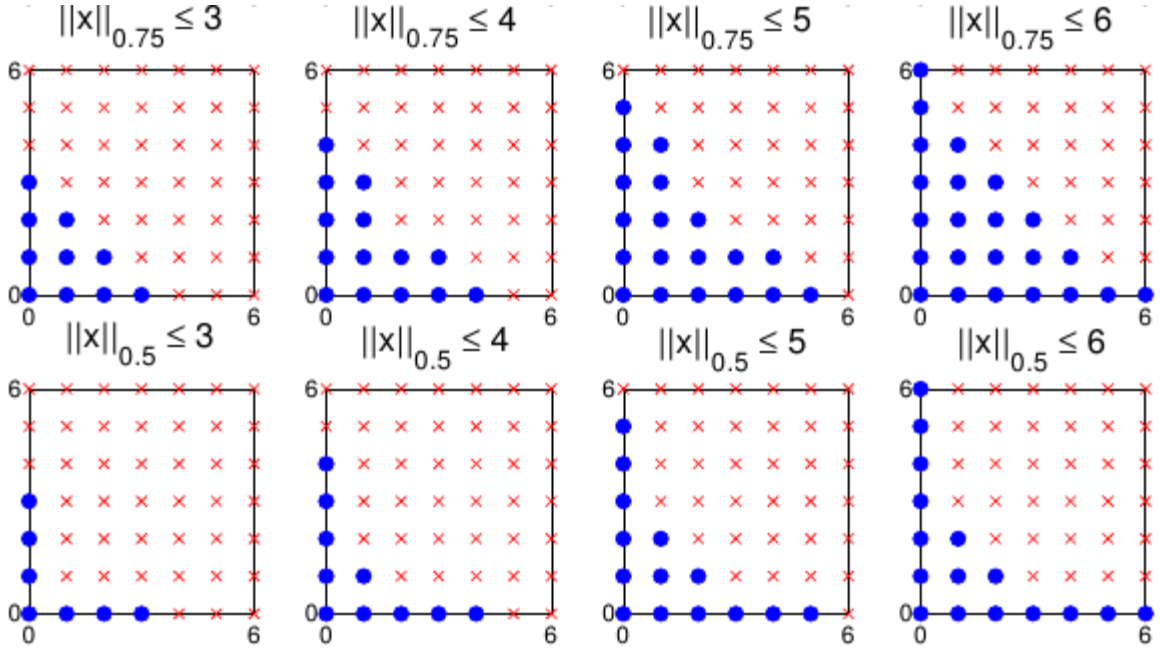
Les interactions sont d'autant plus négligées que *q* est faible.

Pilotage par norme hyperbolique croissante

Pour q fixé, on définit une stratégie d'énumération hyperbolique, qui vise à explorer les polynômes par q-norme hyperbolique croissante.

Suite possible de bases candidates :

$$\Lambda_k = \left\{ \boldsymbol{\alpha} \in \mathbb{N}^M \ : \ \|\boldsymbol{\alpha}\|_q \leq k \right\} \quad , \quad k \in \mathbb{N}^*$$



Algorithme « AdapLAR »

(Blatman & Sudret, 2009)

1. On se donne un *N*-échantillon d'entrées et de sorties du modèle :

$$\left\{\left(\boldsymbol{x}^{(i)}, y^{(i)} = \mathcal{M}(\boldsymbol{x}^{(i)})\right) \quad , \quad i = 1, \ldots, N\right\}$$

2. On choisit une suite de bases candidates de polynômes $\Lambda_{c1} \subset \Lambda_{c2} \subset \cdots$, basée sur une stratégie d'énumération linéaire ou bien hyperbolique (choix d'un paramètre *q*).

3. Pour k=1,...,k_{max} :
   - On exécute LAR en prenant Λ_{ck} pour base candidate : on obtient le chaos polynomial creux $\mathcal{M}_k^*$ et son erreur LOO corrigé $\varepsilon_{LOO,k}^*$.
   - Si $\varepsilon_{LOO,k}^*$ est inférieur à une erreur cible ou bien si $\varepsilon_{LOO,k}^* \geq \varepsilon_{LOO,k-1}^* \geq \varepsilon_{LOO,k-2}^*$ (surapprentissage), alors l'algorithme s'arrête.

Illustration

Fonction de Sobol'. Soit $\mathsf{X}_1, \ldots, \mathsf{X}_8 \sim \mathcal{U}([0,1])$,

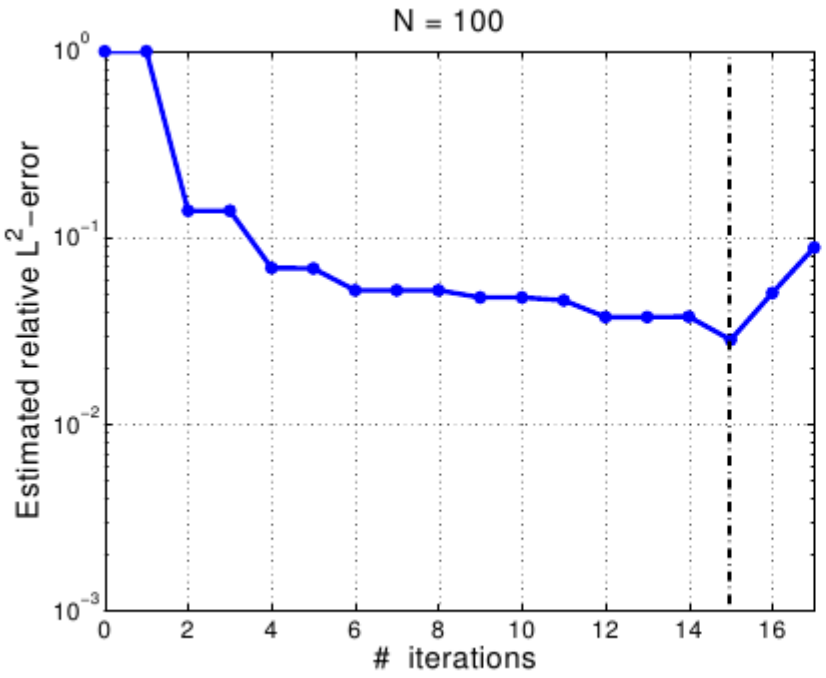
$$\mathsf{Y} = \prod_{i=1}^{M} \frac{|4\mathsf{X}_i - 2| + a_i}{1 + a_i}$$

$$\boldsymbol{a} = (1, 2, 5, 10, 20, 50, 100, 500)$$

Réglages de l'algorithme AdapLAR. Plans d'expériences quasi-aléatoires de tailles *N* égales à 100 et 300 ; stratégie hyperbolique (*q*=0,4) ; erreur-cible égale à 0 (on va jusqu'au surapprentissage).

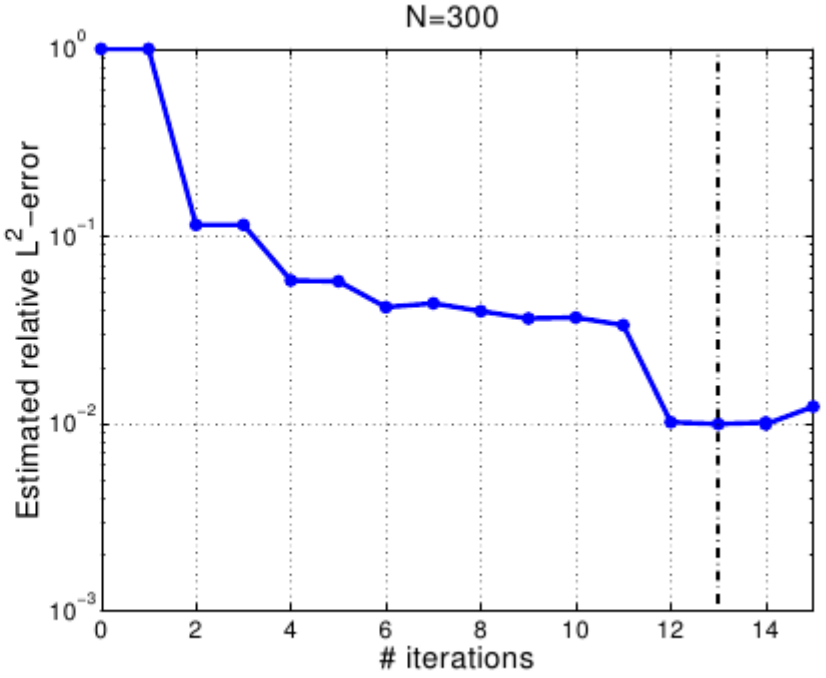
Illustration

Résultats



$$\varepsilon^*_{LOO} \simeq 0,0284$$

$$\mathrm{card}(\Lambda) = 25$$



$$\varepsilon^*_{LOO} \simeq 0,0099$$

$$\mathrm{card}(\Lambda) = 50$$

Conclusion et perspectives

Conclusion

- La stratégie usuelle d'approximation par chaos polynomial (moindres carrés ordinaires + troncature fixe) peut conduire à des coûts de calculs prohibitifs en grande dimension.
- Même si elle apporte des améliorations, la stratégie de troncature « Cleaning » n'est pas toujours évidente à paramétrer.
- La procédure non paramétrique LAR permet la construction automatique d'un chaos polynomial creux à partir d'une base candidate donnée, à un coût de calcul relativement faible.
- La méthode s'appuie sur un estimateur d'erreur sensible au phénomène de surapprentissage (*leave-one-out* corrigé).
- L'algorithme « AdapLAR », qui itère sur LAR et repose sur des schémas d'énumération judicieusement choisis (linéaire ou hyperbolique), permet de s'affranchir du choix d'une base candidate *a priori*.

Perspectives

- Intégration de nouvelles fonctionnalités liées au chaos creux dans la prochaine version d'Open TURNS (OT) : LAR, stratégie de troncature hyperbolique, estimateur d'erreur *leave-one-out* corrigé, etc.

- Comparaison des différentes approches de chaos polynomial disponibles dans OT sur un ou plusieurs exemples d'application (*Example Guide*).

- A plus long terme, extension de la procédure AdapLAR pour enrichir automatiquement le plan d'expériences.

- Extension de AdapLAR pour les réponses vectorielles resultant de la discrétisation d'un champ.

Algorithme itératif du LAR (1)

1. Initialisation : termes actifs $\Lambda := \emptyset$ termes inactifs $\bar{\Lambda} := \Lambda_c$
   coefficients $\forall \boldsymbol{\alpha}\ ,\ a_{\boldsymbol{\alpha}}^{(0)} = 0$; résidu $R^{(0)} = \boldsymbol{Y}$
   $k = 0$

2. Activation du terme le plus corrélé avec le résidu :

$$i \ = \ \arg\max_{j \in \bar{\Lambda}} \left| \boldsymbol{\phi}_{\boldsymbol{\alpha}_j} \left(\boldsymbol{Y} \ - \ \boldsymbol{\Phi}\ \boldsymbol{a}^{(k)} \right) \right| \qquad , \qquad \Lambda \ = \ \Lambda \cup \{ \boldsymbol{\alpha}_i \}$$

3. Calcul de la direction $\boldsymbol{u}$ et du pas γ de descente de la solution.
   Actualisation du résidu et des coefficients associés aux termes actifs :

$$R^{(k+1)} \ = \ R^{(k)} \ - \ \gamma\ \boldsymbol{u}$$
$$\boldsymbol{a}_{\Lambda}^{(k+1)} \ = \ \boldsymbol{a}_{\Lambda}^{(k)} \ + \ \gamma\ \boldsymbol{\omega}$$

4. Tant que $k < \min(\mathrm{card}(\Lambda_c), N-1)$ incrément $k = k+1$
   et retour à l'étape 2.

Algorithme itératif du LAR (2)

Direction de descente. Elle correspond à une direction d'équi-corrélation à tous les prédicteurs actifs :

$$\boldsymbol{\Phi}_\Lambda^\mathsf{T}\,\boldsymbol{u} \;=\; \rho\,\mathbf{1} \quad , \quad \rho \;>\; 0$$

Solution :

$$\boldsymbol{u} \;=\; \boldsymbol{\Phi}_\Lambda \left(\boldsymbol{\Phi}_\Lambda^\mathsf{T}\,\boldsymbol{\Phi}_\Lambda\right)^{-1} \rho\,\mathbf{1}$$

où ρ est déterminé de sorte que $\boldsymbol{u}$ soit de norme unitaire.

Algorithme itératif du LAR (2)

Pas de descente. Il est défini de sorte qu'un et un seul prédicteur dans $\bar{\Lambda}$ soit autant corrélé au résidu que les prédicteurs dans Λ.

On remarque qu'une fois ce pas γ défini, la valeur absolue du coef. de corrélation entre les termes de Λ et le résidu actualisé vaut : $\rho - \gamma$.

Comme le nouveau prédicteur activé est le plus corrélé avec le résidu, le pas doit vérifier :

$$\begin{aligned} \rho - \gamma &= \arg\max_{\boldsymbol{\alpha}_i \in \bar{\Lambda}} \left| \boldsymbol{\phi}_{\boldsymbol{\alpha}_i}^{\mathsf{T}} \, R^{(k+1)} \right| \\ &= \arg\max_{\boldsymbol{\alpha}_i \in \bar{\Lambda}} \left| \underbrace{\boldsymbol{\phi}_{\boldsymbol{\alpha}_i}^{\mathsf{T}} \, R^{(k)}}_{= c_i} - \underbrace{\gamma \, \boldsymbol{\phi}_{\boldsymbol{\alpha}_i}^{\mathsf{T}} \, \boldsymbol{u}}_{= d_i} \right| \end{aligned}$$

D'où le pas : $$\gamma = \arg \min_{\boldsymbol{\alpha}_i \in \bar{\Lambda} \, , \, \gamma > 0} \left(\frac{\rho - c_i}{1 - d_i} \, , \frac{\rho + c_i}{1 + d_i} \right)$$