

Modèle à processus latent et algorithme EM pour la régression non linéaire

Faïcel Chamroukhi ^{*,**}, Allou Samé^{*}, Gérard Govaert^{**}, Patrice Aknin^{*}

^{*}Institut National de Recherche sur les Transports et leur Sécurité
Laboratoire des Technologies Nouvelles
2 rue de la Butte Verte, 93166 Noisy-le-Grand Cedex
{chamroukhi, same, aknin}@inrets.fr

^{**}Université de Technologie de Compiègne
Laboratoire HEUDIASYC, UMR CNRS 6599
BP 20529, 60205 Compiègne Cedex
govaert@utc.fr

Résumé. Cet article propose une méthode de régression non linéaire qui s'appuie sur un modèle intégrant un processus latent qui permet d'activer préférentiellement, et de manière souple, des sous-modèles de régression polynomiaux. Les paramètres du modèle sont estimés par la méthode du maximum de vraisemblance mise en œuvre par un algorithme EM dédié. Une étude expérimentale menée sur des données simulées et des données réelles met en évidence de bonnes performances de l'approche proposée.

1 Introduction

La régression non linéaire est un problème central dans de nombreux domaines qui concernent la prédiction, le débruitage de signaux et leur paramétrisation. Son but est de caractériser au mieux la relation (non linéaire) existant entre une variable dépendante (grandeur physique), que nous supposons scalaire dans cet article, et une variable indépendante qui, comme c'est souvent le cas, est liée au temps. Cette relation non linéaire peut être due au fait que les données sont issues d'un modèle physique intrinsèquement non linéaire par rapport au temps, ou que le processus de génération des données comporte différents régimes linéaires (voire polynomiaux) qui se succèdent au cours du temps.

Plusieurs modèles ont déjà été proposés dans le cadre de l'apprentissage statistique pour résoudre ce type de problème. Parmi ces approches, on peut citer les modèles polynomiaux par morceaux (McGee et Carleton, 1970), (Bellman, 1961; Stone, 1961), (Samé et al., 2007), les méthodes à base de B-splines (Deboor, 1978; Bishop, 2006), le perceptron multi-couche dans sa version régressive (Bishop, 2006) et les méthodes à fonctions de base radiale (Bishop, 2006). La plupart de ces méthodes ramènent généralement le problème de régression non linéaire à des problèmes de régression linéaire simples à résoudre.

Dans ce travail, nous proposons une méthode alternative qui consiste à remplacer le modèle de régression non linéaire habituel par un modèle de régression intégrant un processus

caché, qui permet d'activer préférentiellement et de manière souple un modèle de régression polynomial parmi K modèles. L'utilisation d'une fonction logistique comme loi conditionnelle des variables latentes assure une souplesse de transition (lente ou rapide) entre les différents polynômes, ce qui permet d'obtenir une modélisation correcte des non linéarités. La loi conditionnelle de la variable dépendante, sous ce modèle, constitue un mélange d'experts (Jacobs et al., 1991), (Jordan et Jacobs, 1994) contraint, avec une variance commune pour toutes les composantes du mélange. Une estimation des paramètres du modèle par l'algorithme EM est ainsi proposée.

Cette méthode est exploitée dans le cadre d'une application de suivi d'état de fonctionnement du mécanisme d'aiguillage des rails, où nous avons été amenés à paramétrer des signaux non linéaires acquis durant des manœuvres d'aiguillage.

Dans la section 2 nous décrivons comment le modèle de régression à processus latent peut être utilisé dans le cadre de la régression non linéaire. Ensuite nous présentons la méthode d'estimation des paramètres via l'algorithme EM. La section 4 montre les performances de la méthode proposée sur des données simulées et la section 5 est consacrée à son application sur des données réelles issues du domaine ferroviaire.

2 Régression non linéaire et modèle à processus latent

2.1 Cadre général

On suppose disposer d'un échantillon $((x_1, t_1), \dots, (x_n, t_n))$, où x_i désigne la variable aléatoire scalaire dépendante et t_i la variable temporelle indépendante. Le problème de régression non linéaire, sous sa formulation générique (Antoniadis et al., 1992), consiste à estimer une fonction f de paramètre θ , en considérant le modèle

$$M1 : x_i = f(t_i; \theta) + \varepsilon_i, \quad (1)$$

où les ε_i sont des bruits gaussiens centrés *i.i.d* de variance σ^2 . L'estimation s'effectue généralement par la maximisation de la vraisemblance ou de manière équivalente par la minimisation du critère des moindres-carrés donné par

$$C_1(\theta) = \sum_{i=1}^n (x_i - f(t_i; \theta))^2. \quad (2)$$

Dans le modèle $M1$, la fonction $f(t; \theta)$ représente l'espérance de x conditionnellement à t . La minimisation du critère C_1 , sous certaines conditions de régularité sur f (Antoniadis et al. (1992)), fournit un estimateur asymptotiquement gaussien, efficace et sans biais du paramètre θ . En pratique, on a souvent recours à des algorithmes itératifs d'optimisation qui convergent localement.

Pour couvrir un panel assez large de fonctions non linéaires de régression qui soient facilement paramétrables, nous optons pour des fonctions qui peuvent s'écrire sous la forme d'une somme finie de polynômes pondérés par des fonctions logistiques :

$$f(t; \theta) = \sum_{k=1}^K \pi_k(t; w) \beta_k^T \mathbf{t}, \quad (3)$$

avec

$$\pi_k(t; \mathbf{w}) = \frac{\exp(w_{k0} + w_{k1}t)}{\sum_{\ell=1}^K \exp(w_{\ell 0} + w_{\ell 1}t)}, \quad (4)$$

où le vecteur $\beta_k = (\beta_{k0}, \dots, \beta_{kp})^T$ de \mathbb{R}^{p+1} désigne l'ensemble des coefficients d'un polynôme de degré p et $\mathbf{t} = (1, t, t^2, \dots, t^p)^T$ son vecteur de monômes associés. Le vecteur $\mathbf{w} = (w_{10}, w_{11}, \dots, w_{K0}, w_{K1})$ de \mathbb{R}^{2K} désigne l'ensemble des paramètres de la fonction logistique et $\theta = (\mathbf{w}, \beta_1, \dots, \beta_K)$ représente l'ensemble des paramètres de la fonction f .

Ce modèle particulier de régression peut s'interpréter comme étant un modèle formé de différents sous-modèles de régression activés de manière souple ou brusque par des fonctions logistiques. Cette méthode permet aussi de détecter les points de rupture d'un signal par le suivi au cours du temps du processus latent, et de mesurer la vitesse d'évolution des proportions du mélange (souple ou brusque) grâce à la flexibilité de la transformation logistique utilisée. L'illustration de cette flexibilité se trouve dans (Chamroukhi et al., 2009).

Pour le modèle $M1$, défini par les équations (1), (3) et (4), le choix particulier de la fonction f ne permet pas d'obtenir une solution analytique du problème de minimisation du critère C_1 . Il faut s'appuyer sur une procédure numérique d'optimisation du type Gauss-Newton, Newton-Raphson ou quasi-Newton qui converge localement.

2.2 Modèle de régression à processus latent proposé

Pour atteindre l'objectif d'estimation de la fonction de régression f du modèle $M1$, nous proposons d'utiliser une méthode alternative s'appuyant sur un modèle génératif intégrant un processus latent discret (z_1, \dots, z_n) avec $z_i \in \{1, \dots, K\}$. Ce modèle est défini par :

$$M2 : x_i = \sum_{k=1}^K z_{ik} \beta_k^T \mathbf{t}_i + \varepsilon_i \quad ; \quad \varepsilon_i \sim \mathcal{N}(0, \sigma^2), \quad (5)$$

où z_{ik} vaut 1 si $z_i = k$ et vaut 0 sinon, et où les ε_i sont supposés être indépendants. Les variables z_i du processus latent, conditionnellement aux instants t_i , sont supposées être générées indépendamment suivant la loi multinomiale $\mathcal{M}(1, \pi_1(t_i; \mathbf{w}), \dots, \pi_K(t_i; \mathbf{w}))$, où $\pi_k(t_i; \mathbf{w})$ est défini par l'équation (4). À partir du modèle $M2$, on peut vérifier que conditionnellement à un sous-modèle de régression k et au temps t_i , x_i est distribué suivant une loi normale de moyenne $\beta_k^T \mathbf{t}_i$ et de variance σ^2

$$p(x_i | z_i = k, t_i; \Phi) = \mathcal{N}(x_i; \beta_k^T \mathbf{t}_i, \sigma^2), \quad (6)$$

où $\Phi = (\theta, \sigma^2)$ et $\mathcal{N}(\cdot; \mu, \sigma^2)$ désigne la fonction de densité d'une loi normale d'espérance μ et de variance σ^2 . On peut montrer alors que conditionnellement à t_i , la variable aléatoire x_i est distribuée suivant le mélange de densités normales

$$p(x_i | t_i; \Phi) = \sum_{k=1}^K \pi_k(t_i; \mathbf{w}) \mathcal{N}(x_i; \beta_k^T \mathbf{t}_i, \sigma^2). \quad (7)$$

Il faut noter que cette dernière loi conditionnelle est un mélange d'experts (Jacobs et al., 1991), (Jordan et Jacobs, 1994) contraint, avec une variance commune pour toutes les composantes du

Modèle à processus latent et algorithme EM pour la régression non linéaire

mélange. Le modèle $M2$ constitue un modèle parcimonieux du modèle plus général supposant des variances différentes. En pratique, pour les données traitées, ces deux modèles ont conduit à des ajustements quasi identiques.

On peut alors vérifier que le modèle de régression proposé conduit à la même espérance conditionnelle que celle du modèle $M1$:

$$\begin{aligned}
 E[x|t; \Phi] &= \int_{\mathbb{R}} xp(x|t; \Phi)dx \\
 &= \sum_{k=1}^K \pi_k(t; \mathbf{w}) \int_{\mathbb{R}} x \mathcal{N}(x; \beta_k^T \mathbf{t}, \sigma^2) dx \\
 &= \sum_{k=1}^K \pi_k(t; \mathbf{w}) \beta_k^T \mathbf{t} \\
 &= f(t; \theta).
 \end{aligned} \tag{8}$$

Ainsi, grâce aux propriétés asymptotiques classiques de normalité, d'absence de biais et d'efficacité de l'estimateur du maximum de vraisemblance de θ , la fonction de régression f estimée à partir du modèle $M2$ est asymptotiquement identique à celle obtenue à partir du modèle $M1$. Ce qui nous conforte sur le fait que le modèle proposé peut être une bonne alternative pour résoudre le problème de régression non linéaire, si on dispose d'un algorithme adapté pour l'estimation de ses paramètres.

La section suivante montre comment les paramètres du modèle peuvent être estimés par la méthode du maximum de vraisemblance.

3 Estimation des paramètres via l'algorithme EM

Dans cette section, compte tenu du fait que la densité de la loi conditionnelle de x s'écrit sous la forme d'un mélange de densités, nous exploitons le cadre élégant de l'algorithme EM (Dempster et al., 1977) pour estimer ses paramètres.

Les hypothèses d'indépendance des ε_i et d'indépendance des z_i conditionnellement aux t_i entraînent l'indépendance des x_i conditionnellement aux t_i . La log-vraisemblance à maximiser s'écrit donc

$$\begin{aligned}
 L(\Phi) &= \sum_{i=1}^n \log p(x_i|t_i; \Phi) \\
 &= \sum_{i=1}^n \log \sum_{k=1}^K \pi_k(t_i; \mathbf{w}) \mathcal{N}(x_i; \beta_k^T \mathbf{t}_i, \sigma^2).
 \end{aligned} \tag{9}$$

Cette maximisation ne pouvant pas être effectuée analytiquement, nous nous appuyons sur l'algorithme EM (Dempster et al., 1977) pour l'effectuer. L'algorithme EM, dans cette situation, itère à partir d'un paramètre initial $\Phi^{(0)}$ les deux étapes suivantes jusqu'à la convergence.

3.1 Étape E (Espérance)

Cette étape consiste à calculer l'espérance de la log-vraisemblance complétée

$$\log p(\mathbf{x}, \mathbf{z}; \Phi) = \sum_{i=1}^n \sum_{k=1}^K z_{ik} \log[\pi_k(t_i; \mathbf{w}) \mathcal{N}(x_i; \beta_k^T \mathbf{t}_i, \sigma^2)], \quad (10)$$

conditionnellement aux données observées et au paramètre courant $\Phi^{(q)}$ (q étant l'itération courante). Dans notre situation, cette espérance conditionnelle s'écrit :

$$\begin{aligned} Q(\Phi, \Phi^{(q)}) &= E \left[\log p(\mathbf{x}, \mathbf{z}; \Phi) | \mathbf{x}; \Phi^{(q)} \right] \\ &= \sum_{i=1}^n \sum_{k=1}^K E(z_{ik} | x_i; \Phi^{(q)}) \log[\pi_k(t_i; \mathbf{w}) \mathcal{N}(x_i; \beta_k^T \mathbf{t}_i, \sigma^2)] \\ &= \sum_{i=1}^n \sum_{k=1}^K p(z_{ik} = 1 | x_i; \Phi^{(q)}) \log[\pi_k(t_i; \mathbf{w}) \mathcal{N}(x_i; \beta_k^T \mathbf{t}_i, \sigma^2)] \\ &= \sum_{i=1}^n \sum_{k=1}^K \tau_{ik}^{(q)} \log[\pi_k(t_i; \mathbf{w}) \mathcal{N}(x_i; \beta_k^T \mathbf{t}_i, \sigma^2)], \end{aligned} \quad (11)$$

où

$$\begin{aligned} \tau_{ik}^{(q)} &= p(z_{ik} = 1 | x_i; \Phi^{(q)}) \\ &= \frac{\pi_k(t_i; \mathbf{w}^{(q)}) \mathcal{N}(x_i; \beta_k^{T(q)} \mathbf{t}_i, \sigma^{2(q)})}{\sum_{\ell=1}^K \pi_\ell(t_i; \mathbf{w}^{(q)}) \mathcal{N}(x_i; \beta_\ell^{T(q)} \mathbf{t}_i, \sigma^{2(q)})}, \end{aligned} \quad (12)$$

est la probabilité a posteriori que x_i soit issu de la k ième composante du mélange. Cette étape nécessite simplement le calcul des $\tau_{ik}^{(q)}$.

3.2 Étape M (Maximisation)

Cette étape (de mise à jour) consiste à calculer le paramètre $\Phi^{(q+1)}$ qui maximise $Q(\Phi, \Phi^{(q)})$ par rapport à Φ . La quantité $Q(\Phi, \Phi^{(q)})$ s'écrit sous la forme d'une somme des deux quantités suivantes :

$$Q_1(\mathbf{w}) = \sum_{i=1}^n \sum_{k=1}^K \tau_{ik}^{(q)} \log \pi_k(t_i; \mathbf{w}) \quad (13)$$

et

$$Q_2(\beta_1, \dots, \beta_K, \sigma^2) = \sum_{i=1}^n \sum_{k=1}^K \tau_{ik}^{(q)} \log \mathcal{N}(x_i; \beta_k^T \mathbf{t}_i, \sigma^2). \quad (14)$$

Pour maximiser $Q(\Phi, \Phi^{(q)})$, il suffit donc de maximiser séparément les quantités $Q_1(\mathbf{w})$ et $Q_2(\beta_1, \dots, \beta_K, \sigma^2)$.

Modèle à processus latent et algorithme EM pour la régression non linéaire

La maximisation de $Q_2(\beta_1, \dots, \beta_K, \sigma^2)$ est celle classique qu'on rencontre dans un mélange de lois gaussiennes d'espérance décrite par un modèle linéaire. On obtient les $\beta_k^{(q+1)}$ en résolvant analytiquement K problèmes de moindres-carrés ordinaires pondérés par les $\tau_{ik}^{(q)}$:

$$\begin{aligned}\beta_k^{T(q+1)} &= \arg \min_{\beta} \sum_{i=1}^n \tau_{ik}^{(q)} (x_i - \beta^T \mathbf{t}_i)^2 \\ &= (\mathbf{T}^T \mathbf{W}_k^{(q)} \mathbf{T})^{-1} \mathbf{T}^T \mathbf{W}_k^{(q)} \mathbf{x},\end{aligned}\tag{15}$$

où \mathbf{T} est la matrice de régression de dimension $[n \times (p+1)]$ définie par

$$\mathbf{T} = \begin{bmatrix} 1 & t_1 & t_1^2 & \dots & t_1^p \\ 1 & t_2 & t_2^2 & \dots & t_2^p \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & t_n & t_n^2 & \dots & t_n^p \end{bmatrix},$$

$\mathbf{W}_k^{(q)}$ est une matrice diagonale de dimension $[n \times n]$ ayant pour éléments diagonaux $(\tau_{1k}^{(q)}, \dots, \tau_{nk}^{(q)})$ et $\mathbf{x} = (x_1, \dots, x_n)^T$ est le vecteur de dimension $[(n+1) \times 1]$ des observations. La variance $\sigma^{2(q+1)}$, qui est identique pour toutes les composantes du mélange, est donnée par :

$$\begin{aligned}\sigma^{2(q+1)} &= \arg \min_{\sigma^2} \left[n \log \sigma^2 + \frac{1}{\sigma^2} \sum_{i=1}^n \sum_{k=1}^K \tau_{ik}^{(q)} (x_i - \beta_k^T \mathbf{t}_i)^2 \right] \\ &= \frac{1}{n} \sum_{i=1}^n \sum_{k=1}^K \tau_{ik}^{(q)} (x_i - \beta_k^{T(q+1)} \mathbf{t}_i)^2.\end{aligned}\tag{16}$$

La maximisation de Q_1 par rapport à \mathbf{w} est un problème convexe de régression logistique multinomial pondéré par les $\tau_{ik}^{(q)}$. Contrairement au cas précédent, cette maximisation ne peut pas s'effectuer de manière analytique. On a recours à un algorithme, adapté à ce type de problème, et qui est lui-même itératif : l'algorithme IRLS (Iterative Reweighted Least Squares) (Green, 1984). Le paragraphe suivant décrit cet algorithme.

Algorithme IRLS (Iteratively Reweighted Least Squares) : l'algorithme IRLS est utilisé pour maximiser $Q_1(\mathbf{w})$ par rapport à \mathbf{w} à l'étape M de chaque itération q de l'algorithme EM. Puisque $\sum_{k=1}^K \pi_k(t_i; \mathbf{w}) = 1$, les valeurs de w_{K0} et w_{K1} sont fixées à zéro pour éviter les problèmes d'identification. L'algorithme IRLS est équivalent à l'algorithme de Newton-Raphson qui consiste à partir d'un vecteur paramètre initial $\mathbf{w}^{(0)}$ et à s'appuyer, à chaque nouvelle itération $c+1$, sur la formule suivante :

$$\mathbf{w}^{(c+1)} = \mathbf{w}^{(c)} - \left[H(\mathbf{w}^{(c)}) \right]^{-1} g(\mathbf{w}^{(c)}),\tag{17}$$

où $H(\mathbf{w}^{(c)})$ et $g(\mathbf{w}^{(c)})$ sont respectivement la hessienne et le gradient de $Q_1(\mathbf{w})$ calculés avec le paramètre $\mathbf{w}^{(c)}$. La matrice hessienne $H(\mathbf{w}^{(c)})$ est formée de $(K-1) \times (K-1)$ matrices

blocs $H_{k\ell}(\mathbf{w}^{(c)})$ ($k, \ell = 1, \dots, K-1$) (Chen et al., 1999), où

$$H_{k\ell}(\mathbf{w}^{(c)}) = - \sum_{i=1}^n \pi_k(t_i; \mathbf{w}^{(c)}) [\delta_{k\ell} - \pi_\ell(t_i; \mathbf{w}^{(c)})] \mathbf{v}_i \mathbf{v}_i^T, \quad (18)$$

$\delta_{k\ell}$ étant le symbole de Kronecker ($\delta_{k\ell} = 1$ si $k = \ell$, 0 sinon) et $\mathbf{v}_i = (1, t_i)^T$. Le gradient de $Q_1(\mathbf{w})$ s'écrit :

$$g(\mathbf{w}^{(c)}) = [g_1(\mathbf{w}^{(c)}), \dots, g_{K-1}(\mathbf{w}^{(c)})]^T, \quad (19)$$

avec

$$g_k(\mathbf{w}^{(c)}) = \sum_{i=1}^n [\tau_{ik}^{(q)} - \pi_k(t_i; \mathbf{w}^{(c)})] \mathbf{v}_i^T \quad \forall \quad k = 1, \dots, K-1. \quad (20)$$

L'équation (17), initialisée avec le paramètre $\mathbf{w}^{(q)}$, fournit à la convergence le paramètre $\mathbf{w}^{(q+1)}$.

Chaque itération de l'algorithme EM proposé fait croître le critère de log-vraisemblance (McLachlan et Krishnan, 1997). Cela résulte de la maximisation de $Q(\Phi, \Phi^{(q)})$ et de l'inégalité de Jensen (Jensen, 1906). Il a été prouvé par (Jordan et Xu, 1995) et (Xu et Jordan, 1996), notamment pour le cas du mélange gaussien et celui du mélange d'experts, que l'algorithme EM permet de maximiser localement la log-vraisemblance. On peut limiter le nombre d'itérations de la procédure IRLS interne à l'algorithme EM. Cette version consiste à faire croître $Q(\Phi, \Phi^{(q)})$ à chaque itération au lieu de la maximiser. On peut par exemple limiter le nombre d'itérations de l'IRLS jusqu'à une seule itération (Foulley, 2002). On obtient ainsi un algorithme EM généralisé (GEM) (Dempster et al., 1977; McLachlan et Krishnan, 1997), qui possède les mêmes propriétés de convergence que l'algorithme EM. En pratique, nous avons pu observer que cette limitation entraînait une augmentation du nombre d'itérations de l'algorithme EM. Par conséquent, nous avons opté pour une stratégie qui consiste à initialiser aléatoirement la procédure IRLS seulement pour la première itération de l'algorithme EM. Pour cette première initialisation, la convergence de l'algorithme IRLS requiert une quinzaine d'itérations. À partir de la deuxième itération de l'algorithme EM, l'algorithme IRLS défini par l'équation (17) est initialisé avec le paramètre $\mathbf{w}^{(q)}$ estimé à l'itération précédente de l'algorithme EM. Au delà de la quatrième itération de l'algorithme EM, on observe que l'algorithme IRLS converge en moins de 5 itérations. Les critères d'arrêt utilisés pour l'algorithme EM et l'algorithme IRLS consistent en des valeurs seuils sur les variations relatives des log-vraisemblances à maximiser ($|\frac{L(\Phi^{(q+1)}) - L(\Phi^{(q)})}{L(\Phi^{(q)})}| < 10^{-6}$ pour EM et $|\frac{Q_1(\mathbf{w}^{(c+1)}) - Q_1(\mathbf{w}^{(c)})}{Q_1(\mathbf{w}^{(c)})}| < 10^{-6}$ pour IRLS) ainsi qu'à des nombres d'itérations maximum (1000 pour EM et 50 pour IRLS).

3.3 Choix du nombre de composantes et de l'ordre des polynômes de régression

Les valeurs optimales du nombre de composantes K du modèle et de l'ordre p des polynômes de régression peuvent être obtenues en maximisant le critère d'information bayésien

(BIC) (Schwarz, 1978) défini comme étant le critère de vraisemblance obtenu à la convergence de l'algorithme, pénalisé par un terme qui dépend du nombre de paramètres libres du modèle. La pénalisation utilisée ici est $pen = -\frac{\nu(K,p)\log(n)}{2}$ où $\nu(K,p) = K(p+3) - 1$ est le nombre total de paramètres libres à estimer.

Il est possible d'utiliser aussi le critère d'entropie normalisée NEC (Entropy Normalized Criterion) (Celeux et Soromenho, 1996; Biernacki et al., 1999) adapté au contexte des modèles de mélange.

3.4 Région de confiance de la courbe de régression $f(t; \theta)$

À partir de la modélisation proposée, on peut montrer, comme dans le cadre des mélanges hiérarchiques d'experts pour les modèles linéaires généralisés, que sous certaines conditions de régularité, $f(t; \hat{\theta})$ suit asymptotiquement une loi normale de moyenne $f(t; \theta)$ et de variance $s^2(t; \Phi)$ (Jiang et Tanner, 1999). Cependant, comme les "vrais" paramètres Φ et θ sont inconnus, on propose de les remplacer par leurs estimations $\hat{\Phi}$ et $\hat{\theta}$ fournies par l'algorithme EM. Dans le cas de la régression linéaire, la région de confiance approchée au niveau $1 - \alpha$ est définie par :

$$f(t; \hat{\theta}) \pm \sqrt{\nu_{\theta} p(\nu_{\theta}, n - \nu_{\theta}; \alpha)} s(t; \hat{\Phi}), \quad (21)$$

où $\nu_{\theta} = \dim(\theta)$ et $p(\nu_{\theta}, n - \nu_{\theta}; \alpha) = p(F(\nu_{\theta}, n - \nu_{\theta}) \geq \alpha)$, F étant le 100 $(1 - \alpha)$ percentile d'une variable de Fisher à ν_{θ} et $n - \nu_{\theta}$ degrés de liberté (Tomassone et al., 1992).

La fonction

$$f(t; \hat{\theta}) = \sum_{k=1}^K \pi_k(t; \hat{w}) \hat{\beta}_k^T \mathbf{t} \quad (22)$$

représente la courbe ajustée et

$$s^2(t; \hat{\Phi}) = \frac{1}{n} D(\hat{\Phi})^T I(\hat{\Phi})^{-1} D(\hat{\Phi}) \quad (23)$$

sa variance empirique,

$$I(\hat{\Phi}) = -E \left[\frac{\partial^2 \log p(y|t; \Phi)}{\partial \Phi \partial \Phi^T} \right]_{\Phi = \hat{\Phi}} \quad (24)$$

étant la matrice d'information de Fisher et

$$D(\hat{\Phi}) = \frac{\partial f(t; \theta)}{\partial \Phi} \Big|_{\Phi = \hat{\Phi}} \quad (25)$$

le gradient de $f(t; \theta)$ (Jiang et Tanner, 1999), calculés avec le paramètre estimé $\hat{\Phi}$.

Dans le cas non-linéaire, quand $n - \nu_{\theta}$ est grand, (21) peut être approximée asymptotiquement (Tomassone et al., 1992; Gauchi et al., 2010) et on obtient ainsi la région de confiance approchée au niveau $1 - \alpha$ suivante :

$$f(t; \hat{\theta}) \pm \sqrt{\chi_{\nu_{\theta}, 1-\alpha}^2} s(t; \hat{\Phi}). \quad (26)$$

4 Expérimentation sur des données simulées

L’objet de cette partie est d’évaluer l’approche de régression proposée en utilisant des données simulées. Pour ce faire, nous la comparons à deux méthodes :

- une méthode de régression polynomiale par morceaux (McGee et Carleton, 1970), (Samé et al., 2007), (Chamroukhi et al., 2009) où l’estimation des paramètres est effectuée par une procédure de programmation dynamique (Bellman, 1961; Stone, 1961),
- une méthode de régression à processus markovien caché, c’est-à-dire où le processus caché (z_1, \dots, z_n) est supposé être une chaîne de Markov (Fridman, 1993), (Rabiner, 1989). Les paramètres de ce modèle sont estimés par l’algorithme de Baum-Welch (Baum et al., 1970).

La qualité des estimations fournies par chacune des deux méthodes est l’écart quadratique moyen (EQM) $\frac{1}{n} \sum_{i=1}^n (f_{sim}(t_i) - f_{est}(t_i))^2$ entre la courbe de régression estimée et la courbe de régression simulée où :

- $f_{est}(t_i) = \sum_{k=1}^K \pi_k(t_i; \hat{\mathbf{w}}) \hat{\beta}_k^T \mathbf{t}_i$ pour le modèle proposé,
- $f_{est}(t_i) = \sum_{k=1}^K \hat{z}_{ik} \hat{\beta}_k^T \mathbf{t}_i$ pour le modèle de régression par morceaux,
- $f_{est}(t_i) = \sum_{k=1}^K \omega_k(t_i; \hat{\Phi}) \hat{\beta}_k^T \mathbf{t}_i$ pour le modèle markovien où les $\omega_k(t_i; \hat{\Phi})$ sont les probabilités dites de filtrage $\omega_k(t_i; \hat{\Phi}) = p(z_i = k | x_1, \dots, x_i; \hat{\Phi})$ qui se calculent via une procédure du type “forward-backward” (Rabiner, 1989).

Chaque jeu de données est généré en ajoutant un bruit gaussien centré à des points d’une courbe non linéaire, régulièrement échantillonnés sur l’intervalle temporel $[0; 5]$.

4.1 Paramètres de simulation et réglage des algorithmes

Trois fonctions non linéaires de régression ont été considérées. Ces fonctions et leurs paramètres associés sont fournis dans le tableau 1. Des exemples de données simulées à partir de ces courbes sont représentés sur la figure 1.

Fonctions	Paramètres
$f_1(t, \theta) = \sum_{k=1}^4 \pi_k(t; \mathbf{w}) \beta_k^T \mathbf{t}$	$\beta_1 = [34, -60, 30]$ $\mathbf{w}_1 = [547, -154]$ $\beta_2 = [-17, 29, -7]$ $\mathbf{w}_2 = [526, -135]$ $\beta_3 = [185, -104, 15]$ $\mathbf{w}_3 = [464, -115]$ $\beta_4 = [-804, 343, -35]$ $\mathbf{w}_4 = [0, 0]$
$f_2(t, \theta) = \beta_1^T \mathbf{t} \mathbb{1}_{[0; 2.5]}(t) + \beta_2^T \mathbf{t} \mathbb{1}_{[2.5; 5]}(t)$	$\beta_1 = [33, -20, 4]$ $\beta_2 = [-78, 47, -5]$
$f_3(t) = 20 \sin(1.6\pi t) \exp(-0.7t)$	

TAB. 1 – Expressions analytiques des courbes de régression utilisées avec leurs paramètres

Les trois algorithmes testés ont été lancés avec $(K = 4, p = 2)$ pour la situation 1, $(K = 2, p = 2)$ pour la situation 2 et avec $(K = 5, p = 3)$ pour la situation 3. Pour chaque jeu de données, l’algorithme EM a été lancé à partir de 10 initialisations aléatoires différentes et seule la solution ayant la plus grande vraisemblance a été retenue.

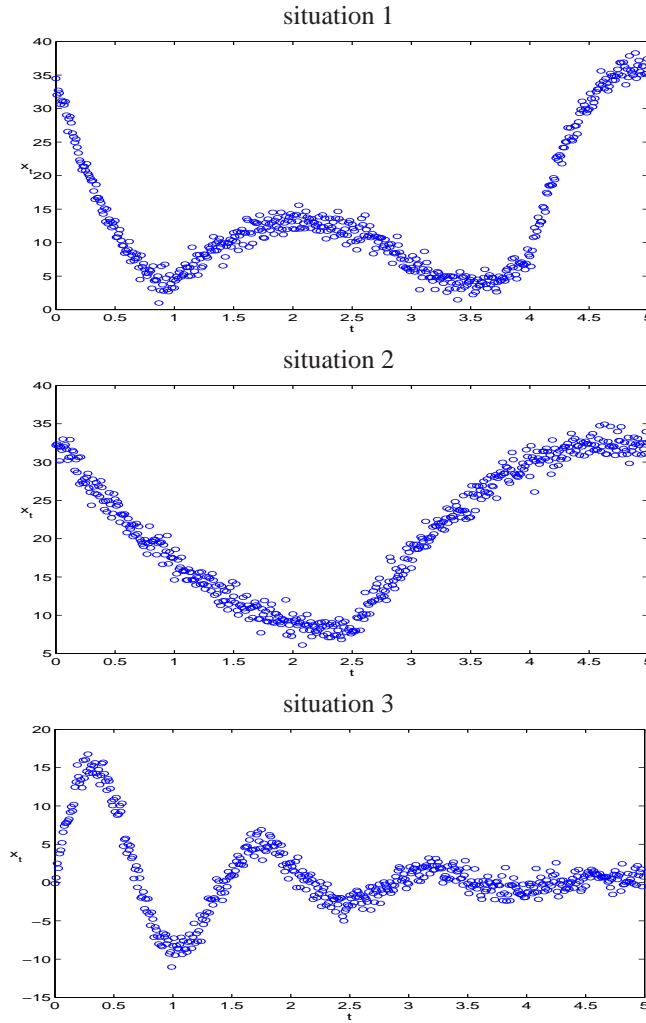


FIG. 1 – Exemple de données simulées à partir des trois courbes

Il faut souligner ici que la première situation a tendance à favoriser l'approche proposée mais permet de la valider. Les deux autres fonctions considérées sont plus pertinentes pour évaluer la méthode proposée. Notamment, la troisième situation constitue un exemple typique de fonction non linéaire.

4.2 Résultats

La figure 2 montre pour chacune des situations, comment varie l'écart entre les courbes de régression simulées et les courbes estimées, en fonction de la taille d'échantillon et de la

variance du bruit. Pour chaque taille d'échantillon et chaque valeur de la variance du bruit, l'erreur quadratique présentée correspond à une moyenne sur 20 jeux de données différents.

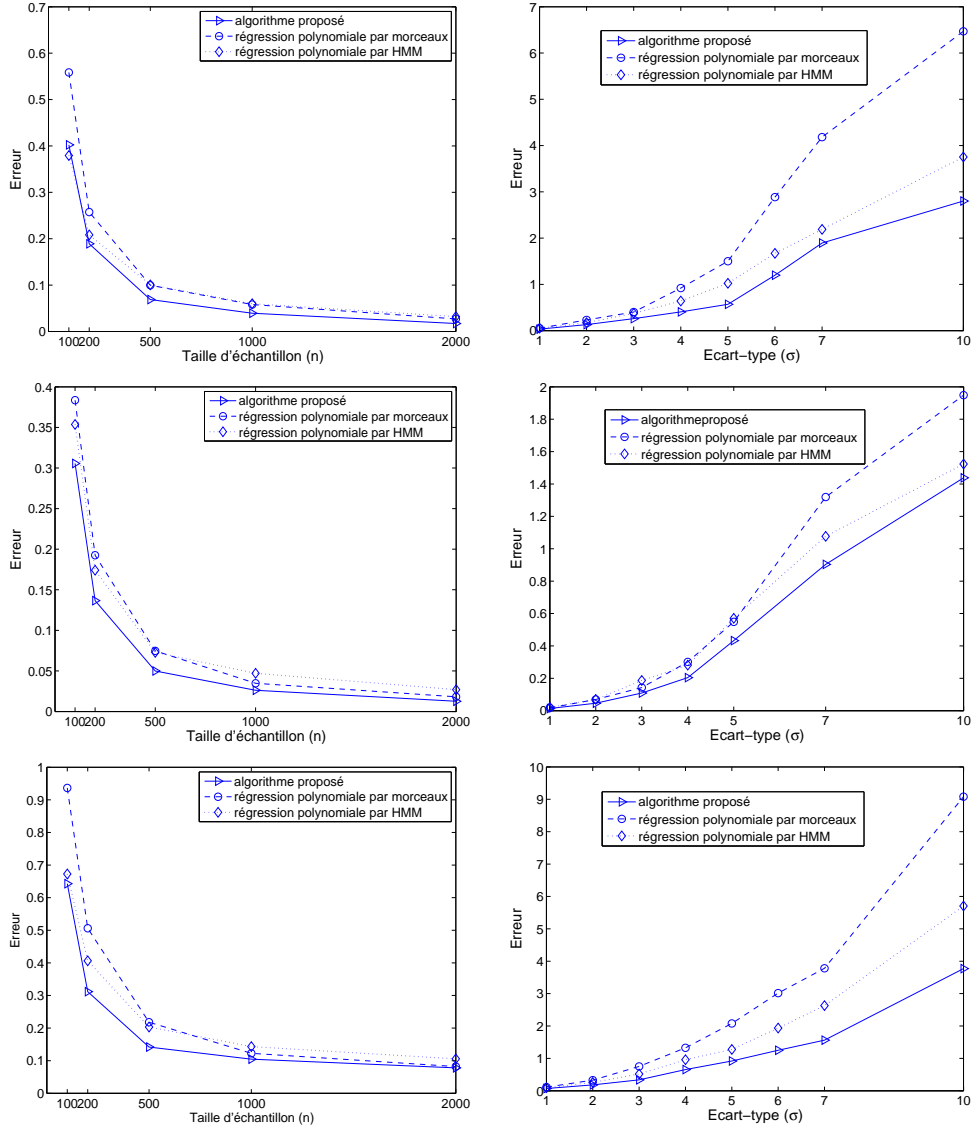


FIG. 2 – Erreur entre la courbe estimée et la courbe réelle simulée en fonction de la taille d'échantillon (gauche) ($\sigma = 1.5$) et en fonction de la variance du bruit (droite) ($n = 500$), pour la situation 1 (haut), la situation 2 (milieu) et la situation 3 (bas)

Ces graphiques montrent que la méthode proposée donne de meilleurs résultats que la méthode de régression polynomiale par morceaux et la méthode basée sur un processus mar-

kovien caché. Cette différence entre les résultats fournis par les trois méthodes s'explique par leur gestion différente des passages d'un sous-modèle à l'autre. En effet, contrairement aux deux alternatives qui sont adaptées à des signaux avec changement brusques, l'approche proposée permet de s'adapter à la fois aux transitions souples et brusques grâce à la flexibilité de la fonction logistique qui modélise le processus latent.

En outre, on peut observer que l'écart entre les courbes estimées et les courbes simulées décroît quand la taille d'échantillon augmente. L'augmentation de la variance du bruit entraîne quant à elle une augmentation de l'erreur qui est plus prononcée pour le modèle de régression polynomial par morceaux. La figure 3 montre, pour chacune des situations, un exemple de courbe de régression estimée avec l'algorithme proposé et la région de confiance à 95 % correspondante.

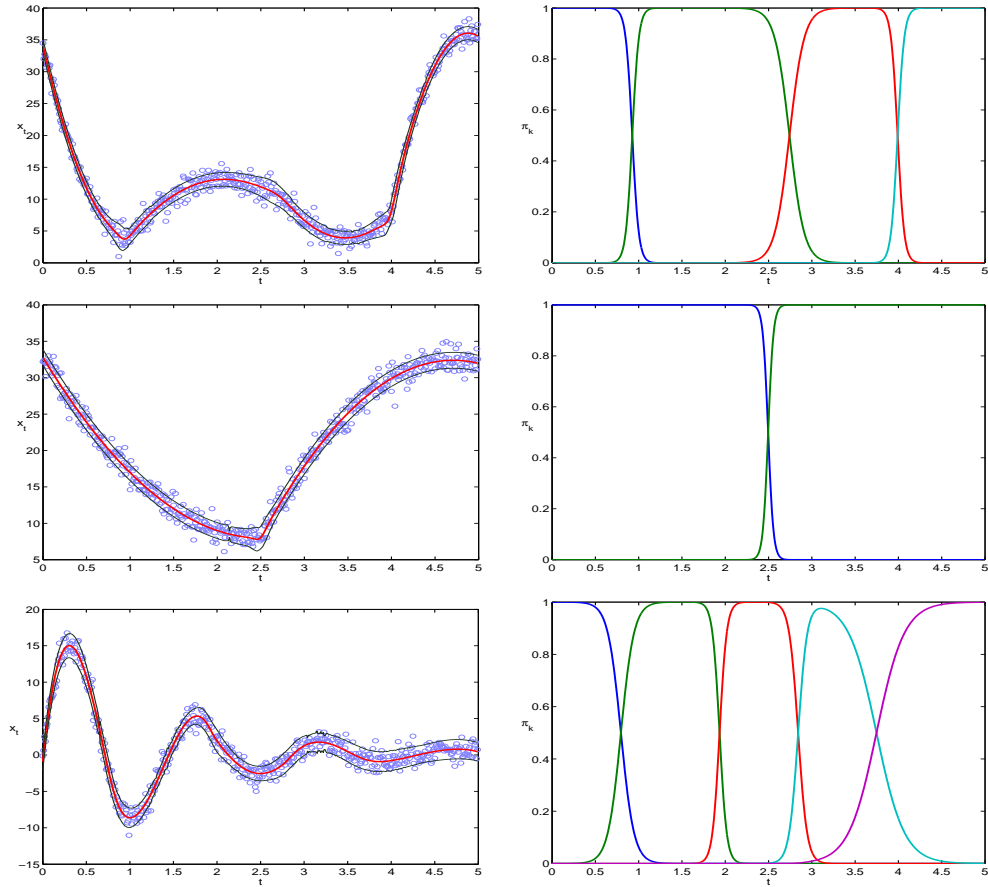


FIG. 3 – Fonctions de régression $f(t; \theta)$ estimées par l'algorithme proposé et régions de confiance à 95 % associées (à gauche) et proportions π_k correspondantes (à droite)

Le tableau 2 montre les pourcentages moyens de choix du nombre de sous-modèles de régression K et du degré p des polynômes, sélectionnés par le critère BIC, sur les signaux

simulés de la situation 1. On observe que les vraies valeurs ($K = 4$ et $p = 2$) ont été sélectionnées avec un pourcentage maximum de 63 %.

	p	1	2	3	4	5	6
K							
2		0	0	0	13	3	0
3		0	0	7	13	0	0
4		0	63	1	0	0	0
5		0	0	0	0	0	0
6		0	0	0	0	0	0
7		0	0	0	0	0	0

TAB. 2 – Pourcentage de choix de (K, p) par le critère BIC obtenu avec les signaux de la situation 1

Le tableau 3 montre les temps de calculs moyens obtenus pour les trois méthodes. On remarque que le modèle proposé et le modèle markovien, qui sont basés sur une méthode itérative de type EM pour l'estimation des paramètres, ont des temps de calculs quasi identiques qui sont peu sensibles à la taille d'échantillon. Cependant, la méthode de régression par morceaux est très coûteuse en temps de calcul à cause de la procédure de programmation dynamique dont le temps de calcul augmente considérablement avec la taille d'échantillon.

n	Algorithme :	Situation 1			Situation 2			Situation 3		
		algo 1	algo 2	algo 3	algo 1	algo 2	algo 3	algo 1	algo 2	algo 3
100		0.27	0.15	0.12	0.08	0.14	0.07	0.35	0.15	0.17
200		0.45	0.67	0.19	0.09	0.66	0.10	0.48	0.70	0.31
500		1.08	5.09	0.46	0.21	5.05	0.21	1.77	5.33	0.74
1000		1.93	25.86	0.64	0.27	25.65	0.34	3.06	26.59	1.01
2000		2.80	147.7	1.39	0.47	147.2	0.65	5.03	148.2	2.17

TAB. 3 – Temps de calculs moyens en secondes, en fonction de la taille d'échantillon n , obtenus avec les trois méthodes : approche proposée (algorithme 1), approche de régression par morceaux (algorithme 2) et approche de régression à processus markovien (algorithme 3), pour les trois situations des données simulées

5 Expérimentation sur des données réelles

Dans le cadre d'une application de suivi d'état de fonctionnement du mécanisme d'aiguillage des rails, nous avons été amenés à paramétrer des signaux non linéaires représentant la puissance consommée par le moteur d'aiguillage durant des manœuvres d'aiguillage. Cette paramétrisation vise à représenter dans un espace de dimension peu élevée ces signaux non linéaires qui sont formés de 562 points. Nous avons réalisé cette tâche par la méthode de régression à processus latent proposée, les paramètres de régression étant directement utilisés comme paramètres des signaux.

Une manœuvre d'aiguillage est constituée de mouvements mécaniques des différents organes liés à l'aiguille, qui sont mobilisés successivement. Ces mouvements se traduisent, sur le signal de puissance consommée en fonction du temps, par différentes phases de fonctionnement (5 phases : Appel moteur, décalage-déverrouillage, translation, verrouillage-calage et

friction). Le nombre des composantes régressives K du modèle, pour chaque signal considéré, a été donc fixé à $K = 5$ qui correspond au nombre de phases d'une manœuvre d'aiguillage. L'ordre du polynôme a été fixé à $p = 3$ qui est adapté à la forme des signaux traités.

La figure 4 nous montre trois exemples de signaux réels sur lesquels nous avons appliqué la méthode. Ces signaux correspondent à trois états de fonctionnement : état sans défaut (a), état avec défaut tolérable (b) et état avec défaut critique (c).

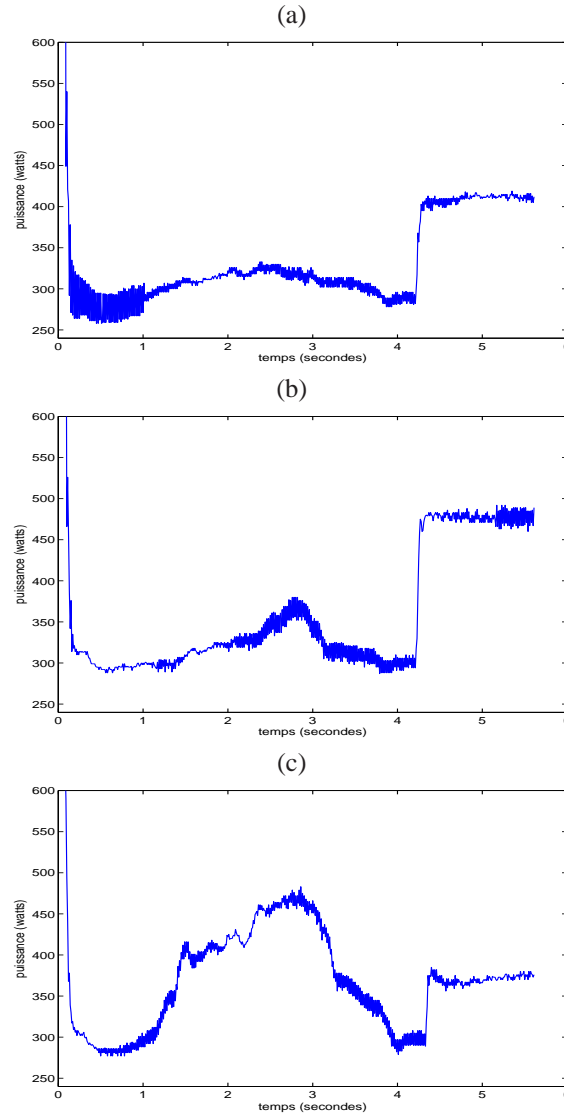


FIG. 4 – Exemples de signaux de manœuvres d'aiguillage

La qualité des estimations fournies par les différentes méthodes est mesurée par l'écart quadratique moyen (EQM) entre le vrai signal et son estimation. Ce critère est donné par $EQM = \frac{1}{n} \sum_{i=1}^n (x_i - \hat{x}_i)^2$.

Les résultats correspondants aux trois situations de signaux de manœuvres d'aiguillages obtenus par les trois méthodes sont donnés dans le tableau 4. Ils mettent en évidence les bonnes performances de la méthode proposée en termes d'écart quadratique moyen. On remarque que, pour la situation (a), les écarts quadratiques moyens obtenus par la régression par morceaux et la régression par HMM sont légèrement inférieurs à l'écart quadratique moyen obtenu avec la méthode proposée. Cela est dû au fait que le signal de la situation (a) présente clairement des changements brusques entre les différentes phases.

Situation	Approche proposée	Régression par morceaux	Régression par HMM
(a)	784.93	781.64	783.20
(b)	1800.81	1928.92	1816.31
(c)	309.80	310.25	314.83

TAB. 4 – Erreurs quadratiques moyennes obtenues par les trois méthodes pour les trois signaux de manœuvres d'aiguillage

La figure 5 montre les résultats graphiques correspondant aux trois signaux présentés. Les proportions du mélange estimées sont cohérentes avec la réalité des phases des manœuvres d'aiguillage considérées.

La figure 6 montre la convergence du critère de log-vraisemblance sur le signal réel présenté dans la figure 4 (c). On peut remarquer pour cet exemple que, la log-vraisemblance croît très rapidement jusqu'à 35 itérations puis augmente très peu jusqu'à la convergence. Pour tous les signaux traités, nous avons pu constater que l'algorithme EM convergeait en un nombre d'itérations autour de 90.

Le tableau 5 montre les pourcentages de choix du degré p des polynômes, sélectionnés par le critère BIC avec $K = 5$ sous-modèles de régression qui correspondent aux phases impliquées dans une manœuvre d'aiguillage. Les valeurs $p = 3$ et $p = 4$ sont majoritairement sélectionnées.

p	1	2	3	4	5	6
%	0	9.5238	31.4286	38.0952	14.2857	6.6667

TAB. 5 – Pourcentage de choix de p avec $K = 5$ obtenu sur 120 signaux réels

6 Conclusion

Une méthode de régression non linéaire a été proposée dans cet article. Cette méthode peut être vue comme une solution alternative au problème des moindres-carrés pour la régression non linéaire. Elle s'appuie sur un modèle de régression simple intégrant un processus latent qui permet d'activer successivement, et manière souple, des sous-modèles de régression polynomiaux. Pour estimer les paramètres du modèle proposé, un algorithme de type EM adapté au contexte de processus latent a été proposé. La méthode proposée se distingue par le fait d'être adaptée aux transitions à la fois souples et brusques grâce à la modélisation particulière utilisée pour le processus latent. Dans l'étude expérimentale menée, les modèles alternatifs considérés

Modèle à processus latent et algorithme EM pour la régression non linéaire

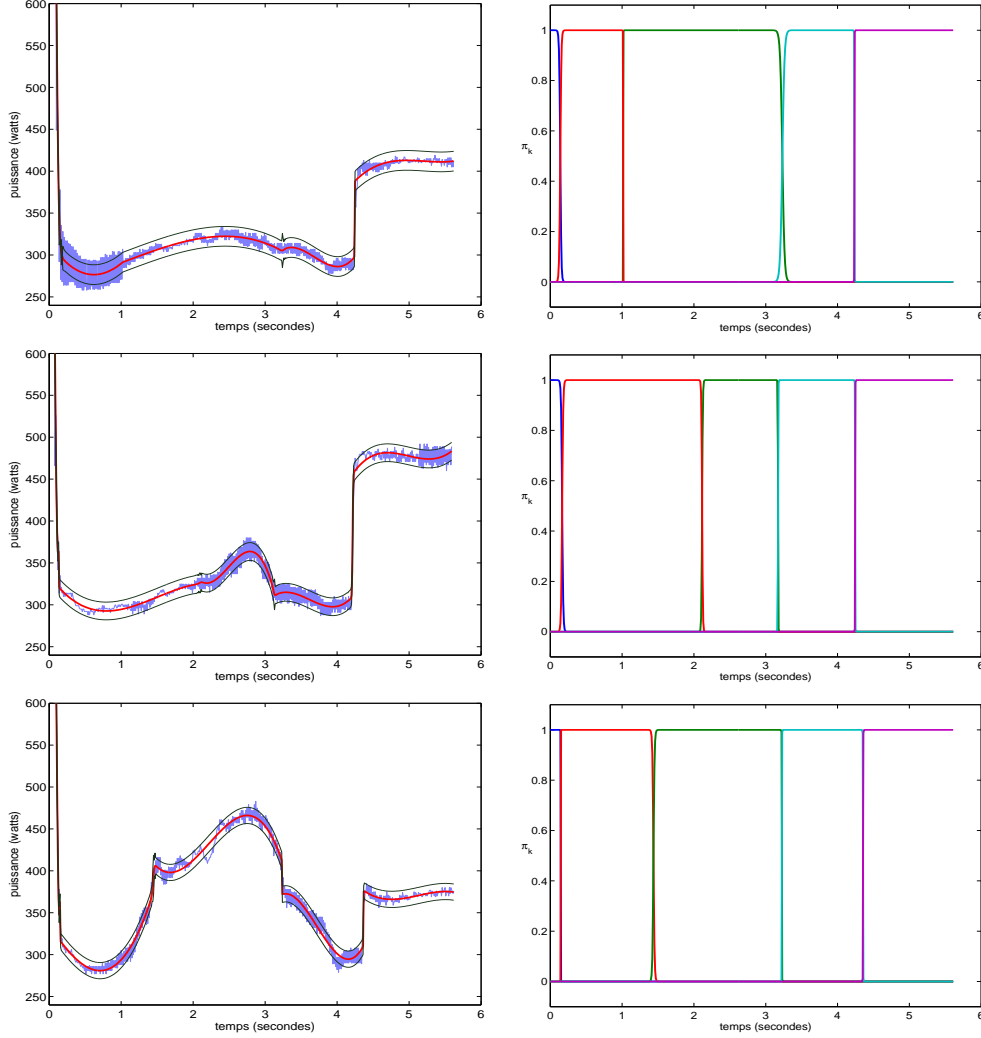


FIG. 5 – *Signal et modèle estimé par l'algorithme proposé et régions de confiance au niveau de confiance de 95 % associées (gauche) et proportions correspondantes (droite), pour la situation sans défaut (haut), avec défaut mineur (milieu) et avec défaut critique (bas)*

sont un modèle de régression par morceaux et un modèle markovien de régression. Les résultats obtenus sur des données simulées et sur des données réelles issues d'une application du domaine ferroviaire confortent l'intérêt de notre démarche.

Remerciements

Les auteurs remercient vivement les relecteurs pour leurs remarques pertinentes, en particulier sur les intervalles de confiance en régression non linéaire.

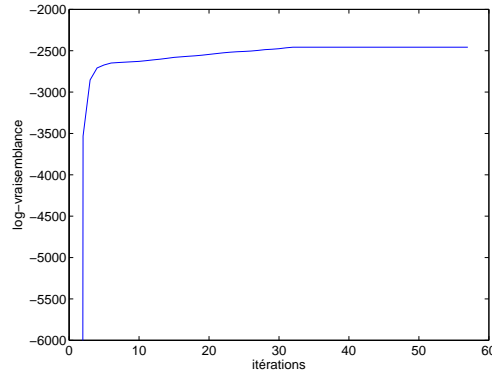


FIG. 6 – Convergence du critère de log-vraisemblance sur le signal réel présenté dans la figure 4 (c)

Références

- Antoniadis, A., J. Berruyer, et R. Carmona (1992). *Régression non linéaire et applications*. Economica.
- Baum, L., T. Petrie, G. Soules, et N. Weiss (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains. *Annals of Mathematical Statistics* 41, 164–171.
- Bellman, R. (1961). On the approximation of curves by line segments using dynamic programming. 4(6), 284.
- Biernacki, C., G. Celeux, et G. Govaert (1999). An improvement of the nec criterion for assessing the number of clusters in a mixture model. *Pattern Recognition Letters* 20, 267–272.
- Bishop, C. M. (2006). *Pattern recognition and machine learning*. U K: Springer Verlag.
- Celeux, G. et G. Soromenho (1996). An entropy criterion for assessing the number of clusters in a mixture model. *Journal of Classification* 13(2), 195–212.
- Chamroukhi, F., A. Samé, G. Govaert, et P. Akin (2009). A regression model with a hidden logistic process for signal parameterization. *Proceedings of XVIIth European Symposium on Artificial Neural Networks ESANN*, 503–508.
- Chen, K., L. Xu, et H. Chi (1999). Improved learning algorithms for mixture of experts in multiclass classification. *Neural Networks* 12(9), 1229–1252.
- Deboor, C. (1978). *A Practical Guide to Splines*. Springer-Verlag.
- Dempster, A. P., N. M. Laird, et D. B. Rubin (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of The Royal Statistical Society, B* 39(1), 1–38.
- Foulley, J. (2002). Algorithme em : Théorie et application au modèle mixte. *Journal de la Société Française de Statistique* 143((3-4)), 57–109.

- Fridman, M. (1993). Hidden markov model regression. Technical report, Institute of mathematics, University of Minnesota.
- Gauchi, J.-P., J.-P. Vila, et L. Coroller (2010). New prediction interval and band in the non-linear regression model : applications to predictive modeling in foods. *Communications in Statistics - Simulation and Computation* 39, 322–334.
- Green, P. (1984). Iteratively reweighted least squares for maximum likelihood estimation, and some robust and resistant alternatives. *Journal of The Royal Statistical Society, B* 46(2), 149–192.
- Jacobs, R. A., M. I. Jordan, S. J. Nowlan, et G. E. Hinton (1991). Adaptive mixtures of local experts. *Neural Computation* 1(3), 79–87.
- Jensen, J. L. W. V. (1906). Sur les fonctions convexes et les inégalités entre les valeurs moyennes. *Acta Mathematica* 30(1), 175–193.
- Jiang, W. et M. A. Tanner (1999). On the asymptotic normality of hierarchical mixtures-of-experts for generalized linear models. *IEEE Transactions on Information Theory* 46, 1005–1013.
- Jordan, M. I. et R. A. Jacobs (1994). Hierarchical mixtures of experts and the em algorithm. *Neural Computation* 6, 181–214.
- Jordan, M. I. et L. Xu (1995). Convergence results for the em approach to mixtures of experts architectures. *Neural Networks* 8(9), 1409–1431.
- McGee, V. E. et W. T. Carleton (1970). Piecewise regression. *Journal of the American Statistical Association* 65, 1109–1124.
- McLachlan, G. J. et T. Krishnan (1997). *The EM algorithm and extensions*. New York: Wiley.
- Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE* 77(2), 257–286.
- Samé, A., P. Akin, et G. Govaert (2007). Classification automatique pour la segmentation des signaux unidimensionnels. *Rencontres de la SFC, ENST, Paris*.
- Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics* 6, 461–464.
- Stone, H. (1961). Approximation of curves by line segments. *Mathematics of Computation* 15(73), 40–47.
- Tomassone, R., E. Lesquoy, et C. Millier (1992). *La Régression nouveaux regards sur une ancienne méthode statistique*. Paris : Masson.
- Xu, L. et M. I. Jordan (1996). On convergence properties of the em algorithm for gaussian mixtures. *Neural Computation* 8(1), 129–151.

Summary

A non linear regression approach which consists of a specific regression model incorporating a latent process, allowing various polynomial regression models to be activated preferentially and smoothly, is introduced in this paper. The model parameters are estimated by maximum likelihood performed via a dedicated EM algorithm. An experimental study using simulated and real data sets reveals good performances of the proposed approach.