

IDENTIFICAÇÃO RECURSIVA DE SISTEMAS DINÂMICOS USANDO MODELOS LOCAIS ROBUSTOS A OUTLIERS

JESSYCA A. BESSA, GUILHERME A. BARRETO*

**Depto. Engenharia de Teleinformática, Universidade Federal do Ceará
Campus do Pici, Centro de Tecnologia, Fortaleza, Ceará.*

Email: `bessa.jessyca@ifce.edu.br`, `gbarreto@ufc.br`

Abstract— In this paper we revisit the design of local linear models for dynamic system identification aiming at adapting their use to scenarios contaminated with outliers. To this purpose, we evaluate the performance of three local linear models originated within the field of artificial neural networks and introduce modifications in their learning rules with the help of a robust statistics framework known as *M-estimation*. The computer experiments carried out over 2 synthetic and 1 real-world datasets corroborate the considerable improvement in the performance of the evaluated models in outlier-contaminated scenarios.

Keywords— System identification, artificial neural networks, local linear models, outliers, *M-estimation*.

Resumo— Neste artigo revisitamos o projeto de modelos lineares locais para identificação de sistemas dinâmicos com o intuito de adaptar seu uso a cenários com presença de outliers nos dados. Para este fim, avaliamos o desempenho de três modelos lineares locais oriundos da área de redes neurais artificiais e promovemos ajustes em suas regras de aprendizado com o auxílio do arcabouço de estatística robusta conhecido como *estimação-M*. Os experimentos computacionais realizados em 2 conjuntos de dados sintéticos e um de dados reais validam nossa hipótese de considerável melhora no desempenho dos modelos avaliados em cenários contaminados com outliers.

Palavras-chave— Identificação de sistemas, redes neurais, modelos lineares locais, *outliers*, *estimação-M*.

1 Introdução

O problema de identificação de sistemas consiste em construir um modelo a partir dos dados de entrada-saída observados (Aguirre, 2015). Existem várias maneiras de descrever um sistema a partir dos dados, escolhendo o modelo mais adequado ao problema de interesse. O processo de identificação equivale a selecionar uma estrutura de modelo (global ou local), estimar seus parâmetros e avaliar a qualidade deste modelo (Ljung, 1998).

Um modelo é dito global quando baseia-se no pressuposto de que todos os seus parâmetros são estimados usando todo o conjunto de dados. Isso nem sempre é computacionalmente viável, principalmente se um grande número de observações está disponível. Uma alternativa consiste em usar modelos locais, cujos parâmetros são estimados usando apenas uma parcela (i.e. partição) dos dados (Foss and Johansen, 1993; Murray-Smith, 1994; Murray-Smith and Johansen, 1995).

Independente da estrutura do modelo (global ou local) é desejável que o modelo seja capaz de lidar com amostras discrepantes, comumente chamadas de *outliers*, conferindo-lhe robustez em cenários não-ideais de uso. Tendo isso em mente, neste artigo revisitamos o projeto de modelos lineares locais para identificação de sistemas dinâmicos. O objetivo maior consiste em adaptar tais modelos para uso em cenários mais realistas, i.e. com forte ocorrência de *outliers* nos dados.

Para isso, escolhemos três modelos lineares locais oriundos da área de redes neurais artificiais e modificamos suas regras de aprendizado com o

auxílio do arcabouço de estatística robusta conhecido como *estimação-M* (Huber, 1964). Experimentos computacionais são realizados em 2 conjuntos de dados sintéticos e um de dados reais com o objetivo de validar nossa hipótese de que o uso de estimadores-*M* aumenta consideravelmente a robustez dos modelos avaliados a *outliers*.

O restante do artigo está organizado da seguinte maneira. Na Seção 2 descrevemos os modelos locais avaliados neste artigo. O arcabouço teórico da estimação-*M* é descrito brevemente na Seção 3. Resultados são reportados e discutidos na Seção 4. O artigo é concluído na Seção 5.

2 Modelos Locais Avaliados

Para todas os modelos a serem apresentados nesta seção, devemos considerar que, para a tarefa de identificação de sistemas, cada vetor de entrada $\mathbf{x}(t) \in \mathbb{R}^{p+q}$ é definido como

$$\mathbf{x}(t) = [u(t-1), \dots, u(t-p), y(t-1), \dots, y(t-q)]^T, \quad (1)$$

onde $\mathbf{x}(t)$ é chamado também de vetor de regressores e $p + q = n$. Note que $y(t) \in \mathbb{R}^m$ é o vetor que representa a saída desejada dada por $y(t) = f(\mathbf{x}(t))$ e $f: \mathbb{R}^n \rightarrow \mathbb{R}^m$.

Local Linear Mapping (LLM) - A idéia básica do modelo LLM (Walter et al., 1990) é aproximar uma função por meio de um conjunto de mapeamentos lineares, cada um dos quais limitado a uma região local do espaço de entrada $\mathbb{R}^n$. Cada região local está associada a um neurônio e cada neurônio está associado a um filtro FIR/LMS. Esta matriz é usada para quantizar o espaço de entrada

em um número reduzido de vetores de protótipos, enquanto o filtro associado ao neurônio vencedor fornece uma estimativa local da saída a ser aproximada.

Mais formalmente, a quantização vetorial do espaço de entrada χ é realizada pelo modelo LLM via algoritmo SOM (*Self Organizing Maps*), com cada neurônio j possuindo um vetor protótipo $\mathbf{w}_j$, $j = 1, \dots, S$, com S denotando o número de neurônios. Associado a cada vetor $\mathbf{w}_j$, existe um vetor de coeficientes $\mathbf{a}_j(t) \in \mathbb{R}^{p+q}$ que desempenha o papel dos coeficientes de um modelo ARX local:

$$\mathbf{a}(t) = [a_{j,1}, \dots, a_{j,p}, \dots, a_{j,p+q}]^T, \quad (2)$$

onde $m = 1$. Assim, os parâmetros ajustáveis do modelo LLM são o conjunto de vetores de peso $\mathbf{w}_j$ e seus respectivos vetores de coeficientes $\mathbf{a}_j$, com $j = 1, \dots, S$.

Seguindo a lógica do algoritmo SOM, a escolha do neurônio vencedor baseia-se na menor distância euclidiana do vetor de pesos ao vetor de entrada:

$$j^*(t) = \arg \min \|\mathbf{x}(t) - \mathbf{w}_{j^*}(t)\|, \quad (3)$$

de modo que o valor predito da saída do modelo baseado em LLM é então calculado como

$$\hat{y}(t) = \mathbf{a}_{j^*}^T(t) \mathbf{x}(t), \quad (4)$$

onde $\mathbf{a}_{j^*}(t)$ é o vetor de coeficientes do filtro linear associado ao neurônio vencedor $j^*(t)$ e é usado para construir uma aproximação local para a saída.

A regra de aprendizado dos protótipos $\mathbf{w}_j$ segue a forma usual daquela do algoritmo SOM:

$$\mathbf{w}_j(t+1) = \mathbf{w}_j(t) + \alpha(t)h(j^*, j; t)[\mathbf{x}(t) - \mathbf{w}_j(t)], \quad (5)$$

enquanto a regra de ajuste dos vetores de coeficientes $\mathbf{a}_j(t)$ é uma extensão do algoritmo LMS normalizado :

$$\mathbf{a}_j(t+1) = \mathbf{a}_j(t) + \alpha'(t)h(j^*, j; t)\Delta\mathbf{a}_j(t), \quad (6)$$

onde $0 < \alpha, \alpha' \ll 1$ são as taxas de aprendizado do vetor de peso e vetor de coeficientes, respectivamente. O termo de correção $\Delta\mathbf{a}_j(t)$ é computado via regra LMS (Widrow, 2005):

$$\Delta\mathbf{a}_j(t) = [y(t) - \mathbf{a}_{j^*}^T(t)\mathbf{x}(t)] \frac{\mathbf{x}(t)}{\|\mathbf{x}(t)\|^2} \quad (7)$$

onde $y(t)$ é a saída observada. Métodos baseados em LLM foram utilizados para o aprendizado da cinemática inversa dos braços de um robô (Ritter et al., 1992), para a previsão de séries temporais (Martinetz et al., 1993) e para reconhecimento de padrões de plantas (Moshou et al., 1998).

Radial Basis Functions (RBF) Networks - A rede RBF é uma arquitetura clássica de rede neural de uma camada oculta (Musavi et al., 1992).

Os neurônios ocultos, designados como funções de base radial, possuem uma função de ativação não linear, enquanto, os neurônios de saída tem, em geral, uma função de ativação linear.

No modelo RBF, considere o centro da função de base representado como $\mathbf{c}$. A j -ésima função de base é composta de dois elementos: uma métrica de distância $d_j(\mathbf{x}) = d_j(\mathbf{x}; \mathbf{c}_j)$ e a função de base em si, $z_j = \phi(d_j(\mathbf{x}))$. Uma métrica de distância euclidiana, $d_j(\mathbf{x}) = \|\mathbf{x} - \mathbf{c}_j\|^2$, e uma função de base gaussiana para $z_j = \exp\{-d_j^2/2\}$, são opções comuns, com σ denotando o raio (ou largura) da função de base.

Existem várias maneiras de projetar a rede RBF, mas nos restringiremos ao procedimento descrito por Moody and Darken (1989). Estes autores treinam uma rede RBF em três etapas executadas sequencialmente. (i) determinação dos centros por através de um algoritmo de quantização vetorial, tal como a rede SOM. (ii) Determinação do raio σ das funções de base. Para este fim, o raio de um neurônio tem seu valor definido como metade da distância entre o centro do neurônio j e o centro mais próximo. (iii) Determinação do vetor de pesos de saída $\mathbf{w} \in \mathbb{R}^S$.

Para aplicações que envolvem aproximação de funções, como identificação de sistemas, normalmente a função de ativação do neurônio de saída é linear¹:

$$\hat{y}(t) = \mathbf{w}^T(t)\mathbf{z}(t) = \sum_{j=1}^S w_j(t)z_j(t), \quad (8)$$

em que $\mathbf{w} = [w_1 \ w_2 \ \dots \ w_S]^T$ é o vetor de pesos do neurônio de saída e $\mathbf{z} = [z_1 \ z_2 \ \dots \ z_S]^T$ é o vetor com as ativações das funções de base. A regra LMS é usada para adaptar os pesos de saída:

$$\mathbf{w}(t+1) = \mathbf{w}(t) + \alpha(t)[y(t) - \hat{y}(t)]\mathbf{z}(t). \quad (9)$$

Local Model Network (LMN) - O modelo LMN foi introduzido por Johansen and Foss (1992) como uma generalização da rede RBF (Murray-Smith and Johansen, 1995). Esta abordagem pode ser vista como um sistema não-linear em um conjunto de submodelos localmente acurados que são, então, integrados suavemente por funções de base associadas. Isso significa que um número menor de modelos locais pode abranger áreas maiores do espaço de entrada, quando comparado com a rede RBF simples.

Assim, a Eq. (8) do modelo RBF é generalizada para envolver não apenas um peso constante associado a cada função de base, mas também uma função do vetor de entrada $\mathbf{x}$ e de um vetor pesos $\mathbf{w}_j$ associado a cada função de base, passando a ser escrita como

$$\hat{y}(t) = \sum_{j=0}^S z_j(t)f_j(\mathbf{x}; \mathbf{w}_j), \quad (10)$$

¹Aqui assumimos apenas um neurônio de saída.

Tabela 1: Descrição dos conjuntos de dados sintéticos usados nos experimentos computacionais. $U(A, B)$ é um número aleatório uniforme entre A e B .

#	Saída	Entrada/Amostras		Ruído
		Estimação	Teste	
1	$y_i = \frac{y_{i-1}}{1+y_{i-1}^2} + u_{i-1}^3$	$u_i = U(-2, 2)$ 300 amostras	$u_i = \sin(2\pi i/25) + \sin(2\pi i/10)$ 100 amostras	$\mathcal{N}(0, 0.65)$
2	$y_i = y_{i-1} - 0.5 \tanh(y_{i-1} + u_{i-1}^3)$	$u_i = \mathcal{N}(u_i 0, 1)$ $-1 \leq u_i \leq 1$ 150 amostras	$u_i = \mathcal{N}(u_i 0, 1)$ $-1 \leq u_i \leq 1$ 150 amostras	$\mathcal{N}(0, 0.0025)$

onde $f_j(\mathbf{x}; \mathbf{w}_j)$ é a função local associada à j -ésima função de base, como por exemplo uma função linear: $f_j(\mathbf{x}; \mathbf{w}_j) = \mathbf{w}_j^T \mathbf{z}$.

A idéia de usar modelos locais dentro de uma rede RBF foi sugerida em vários trabalhos que remontam ao início da década de 1990 (Jones et al., 1990; Stokbro et al., 1990; Barnes et al., 1991; Jacobs et al., 1991), mas aplicações mais recentes deste modelo local continuam a aparecer (König et al., 2014; Costa et al., 2015; Mürker et al., 2017; Belz et al., 2017).

3 Identificação Robusta via Estimação- M

Em aplicações do mundo real de estimação de parâmetros em tarefas de regressão, os dados disponíveis são frequentemente contaminados por *outliers* (Barnett et al., 1994), que são essas observações que parecem ser inconsistentes com o restante de conjunto de dados. Algumas das suas principais fontes são erros de medição, erros humanos (e.g. digitação), erros de arredondamento, falhas no sistema e mudanças no *setpoint* do sistema.

Técnicas de regressão robusta buscam desenvolver estimadores que são resilientes à presença de *outliers*, tal como os desenvolvidos no escopo da *estimação- M* (Huber, 1964), em que M advém de *máxima verossimilhança*. Neste contexto, Zou et al. (2000) e Chan and Zhou (2010) propuseram uma versão robusta a *outliers* da regra LMS, que recebeu o nome de *least mean M -estimate* (LMM).

A regra LMM introduz modificações mínimas na regra LMS original, sendo escrita para o caso da Eq. (9) como

$$\mathbf{w}(t+1) = \mathbf{w}(t) + \alpha(t)q(e(t))e(t)\mathbf{x}(t) \quad (11)$$

onde o termo $q(e(t))$ denota a função de ponderação do erro. Neste artigo, usaremos a função de Huber como função de ponderação:

$$q(e(t)) = \begin{cases} \frac{\kappa}{|e(t)|}, & \text{se } |e(t)| > \kappa, \\ 1, & \text{caso contrário.} \end{cases} \quad (12)$$

em que a constante $\kappa > 0$ é um limiar a partir do qual a magnitude do erro é considerada alta demais para ser produzida por uma amostra comum.

Pode se ver que quando $|e(t)|$ é menor do que κ , a função de peso $q(e(t))$ é igual a 1 e a Eq. (11) torna-se idêntica ao algoritmo LMS. Quando $|e(t)|$ é maior do que κ , $q(e(t))$, começa a reduzir e é igual a zero quando $|e(t)| \rightarrow \infty$. Assim, o algoritmo LMM efetivamente reduz o efeito de grandes erros, geralmente causados por *outliers*, durante o processo de estimação dos parâmetros do modelo. Recomenda-se-se usar $\kappa = 1.345\sigma$, onde σ corresponde ao desvio padrão dos resíduos.

A principal hipótese de trabalho deste artigo é que a substituição da regra LMS usada pelos modelos locais apresentados na Seção 2 pela regra LMM confere àqueles modelos uma maior resiliência a *outliers*. Almeja-se avaliar se há melhoria clara no desempenho das versões robustas e quantificar esta melhoria. Os experimentos computacionais que testaram essa hipótese e os resultados obtidos são apresentados e discutidos em detalhes na próxima seção.

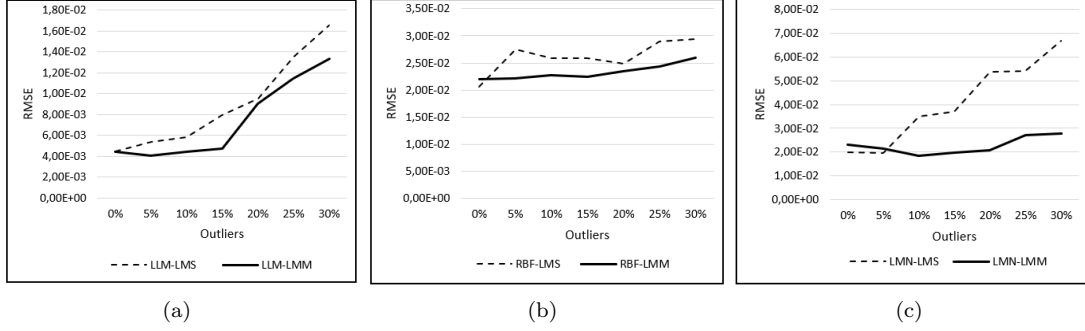
4 Resultados e Discussão

Nesta seção, reportamos os resultados de um amplo estudo comparativo de desempenho dos modelos locais descritos anteriormente. Ao todo, seis modelos são implementados em Octave/Matlab, 3 em suas versões originais (LLM/LMS, RBF/LMS e LMN/LMS) e suas 3 versões robustas (LLM/LMM, RBF/LMM e LMN/LMM). Três conjuntos de dados comumente usados em estudos de *benchmarking* em tarefas de identificação de sistemas dinâmicos foram escolhidos para o estudo comparativo, sendo dois conjuntos de dados sintéticos e um de dados reais (atuador hidráulico). Todos os modelos são numericamente avaliados por meio do erro médio quadrático: $RMSE = \sqrt{\frac{1}{N} \sum_{t=1}^N e^2(t)}$.

Resultados para Dados Sintéticos: Detalhes sobre estes conjuntos estão na Tabela 1. Além do ruído gaussiano adicionado à cada amostra de saída (y_i) dos dados de treinamento, uma proporção de 0%, 5%, 10%, 15%, 20%, 25% e 30% do total das saídas de treinamento também são contaminadas com *outliers* artificialmente gerados. Para isso, seleciona-se aleatoriamente uma

Tabela 2: RMSE na fase de teste usando conjunto de dados sintéticos 1.

Modelos Locais Avaliados						
Outliers	LLM-LMS	LLM-LMM	RBF-LMS	RBF-LMM	LMN-LMS	LMN-LMM
0%	4.50E-03	4.44E-03	2.07E-02	2.20E-02	1.99E-02	2.30E-02
5%	5.37E-03	4.10E-03	2.74E-02	2.22E-02	1.96E-02	2.13E-02
10%	5.86E-03	4.48E-03	2.58E-02	2.27E-02	3.51E-02	1.85E-02
15%	7.98E-03	4.73E-03	2.59E-02	2.24E-02	3.70E-02	1.98E-02
20%	9.53E-03	9.04E-03	2.48E-02	2.34E-02	5.39E-02	2.07E-02
25%	1.35E-02	1.15E-02	2.90E-02	2.43E-02	5.42E-02	2.70E-02
30%	1.66E-02	1.33E-02	2.93E-02	2.59E-02	6.68E-02	2.78E-02

Figura 1: RMSE no conjunto de teste para diferentes proporções de *outliers* durante o treinamento (dados sintéticos 1): (a) LLM, (b) RBF e (c) LMN.

certa proporção de valores de saída (y_i) do total de dados de treinamento e adiciona-se a estes valores ruído não-gaussiano gerado a partir de uma densidade *t*-Student de média nula e 2 graus de liberdade escalonado pelo desvio-padrão da saída. Esta metodologia de contaminação é a mesma utilizada em Mattos et al. (2017).

Definimos 200 épocas de treinamento para todos os modelos. O número de neurônios foi definido em $S = 5$ após alguma experimentação inicial com os dados. Valores de S maiores que 5 não melhoraram significativamente os resultados de um modo geral, enquanto valores menores conduziram a desempenho inferiores de todos os modelos avaliados, robustos ou não. Os resultados a serem reportados correspondem à tarefa de predição livre, que equivale à tarefa de predição com realimentação da saída predita para a entrada. As taxas de aprendizado inicial e final foram definidas em $\alpha_0 = 0,5$ e $\alpha_T = 0,001$.

Os valores de RMSE para o conjunto de dados sintéticos 1 estão mostrados na Tabela 2. Ao analisar esta tabela pode-se observar o melhor desempenho (i.e. menor RMSE) para as versões robustas dos modelos locais avaliados. A tendência de as versões robustas sofrerem menor influência dos *outliers* durante a fase de treinamento e, consequentemente, apresentarem melhor desempenho no conjunto de teste, pode ser melhor apreciada na Figura 1. Nesta, os desempenhos dos modelos original e robusto são avaliados em função da proporção de *outliers* no conjunto de treinamento aumenta. Pode-se observar que as versões robustas sentem menos a presença dos *outliers* pois tendem a apresentar menor RMSE que suas versões origi-

nais, com especial destaque para o modelo LMN robusto - vide Figura 1(c).

A título de ilustração do ganho de desempenho da versão robusta sobre a versão original, na Figura 2 são mostradas as séries preditas pelos modelos LLM-LMS (original) e LLM-LMM (robusto), na fase de teste para o conjunto sintético 1, quando ambos os modelos são treinados com 15% de *outliers*. Percebe-se bem o melhor desempenho da versão robusta do modelo LLM. Vale ressaltar que a escolha por mostrar apenas os resultados de predição do modelo RBF é arbitrária, pois os modelos LMM e RBF são similares.

Os desempenhos dos modelos avaliados para o conjunto de dados sintéticos 2 seguem o mesmo padrão de comportamento que aqueles obtidos para o conjunto de dados sintéticos 1, com as versões robustas apresentando melhor desempenho que as versões originais, especialmente quando o número de *outliers* é expressivo. Para não tornar a exposição repetitiva, na Figura 3 são mostradas as séries preditas pelos modelos LMN-LMS (original) e LMN-LMM (robusto), na fase de teste para o conjunto sintético 2, quando ambos os modelos são treinados com 15% de *outliers*. Como esperado o desempenho da versão robusta do modelo LMN é bem superior ao de sua versão original (treinada com a regra LMS). A escolha por reportar na Figura 3 apenas os resultados da predição do modelo RBF é arbitrária, pois os modelos LMM e RBF são similares.

Resultados para atuador Hidráulico: Este conjunto de dados é formado por duas séries temporais, uma de entrada e outra de saída, de medidas em um atuador hidráulico de uma

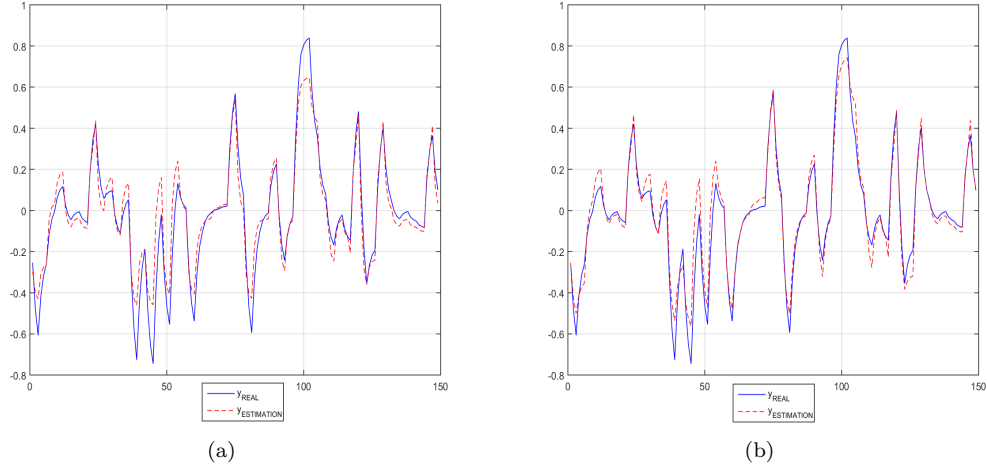


Figura 2: Predição da saída usando para o modelo LLM treinado com 15% de *outliers* (dados sintéticos 1): (a) Versão original e (b) versão robusta.

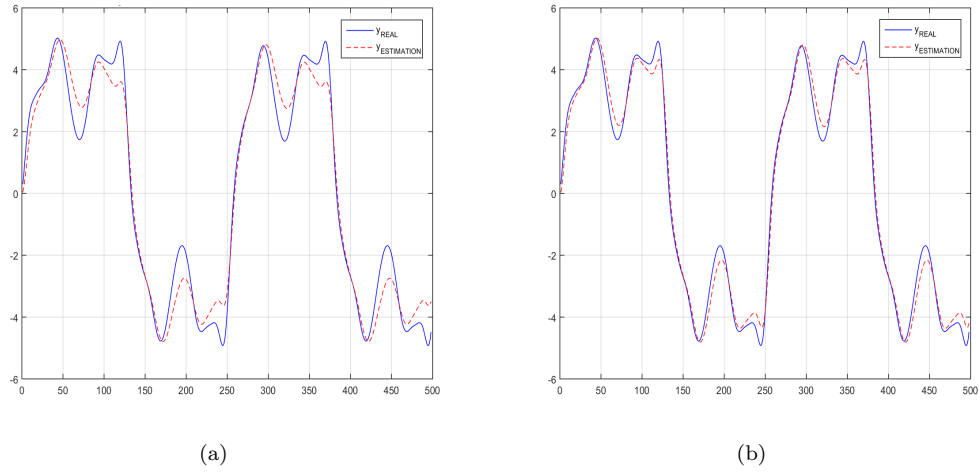


Figura 3: Predição da saída usando para o modelo LMN treinado com 15% de *outliers* (dados sintéticos 2): (a) Versão original e (b) versão robusta.

grua (Sjöberg et al., 1995). Do total de 1024 amostras, utilizamos 512 amostras para construção de modelos (conjunto de treinamento) e a outra metade para avaliação de modelo (conjunto de testes). Optou-se por definir $p = q = 4$, conforme sugerido no trabalho de Santos and Barreto (2017).

O número de neurônios escolhido foi o mesmo dos experimentos com os conjuntos de dados artificiais (i.e. $S = 5$). As taxas de aprendizado inicial e final também são as mesmas. O mesmo procedimento de contaminação dos dados de treinamento com *outliers* nas proporções usadas nos experimentos anteriores com dados sintéticos foi mantido. Contudo, mostramos aqui apenas os resultados para a proporção de 15% de *outliers* por questões de espaço.

Os resultados correspondentes às séries preditas pelo modelo RBF para o conjunto do atuador hidráulico estão mostrados na Figura 4. É possível perceber sem muito esforço que o desempe-

nho da versão robusta desse modelo (RBF-LMM) é bastante superior que o da versão original (RBF-LMS).

5 Conclusões

Neste artigo, abordamos a tarefa de identificação de sistemas dinâmicos em cenários contaminados com *outliers*. Para este fim, revisitamos uma classe de modelos de identificação conhecida como modelos locais e a questão da robustez na estimação de parâmetros na presença de *outliers*. Os modelos locais discutidos utilizavam a conhecida regra de Widrow-Hoff, também chamada de regra LMS, na estimação recursiva de seus parâmetros. A proposta principal do artigo consistiu na substituição da regra LMS por uma versão robusta a *outliers*, chamada de regra LMM, proposta por Zou et al. (2000).

Os resultados de um amplo estudo compara-

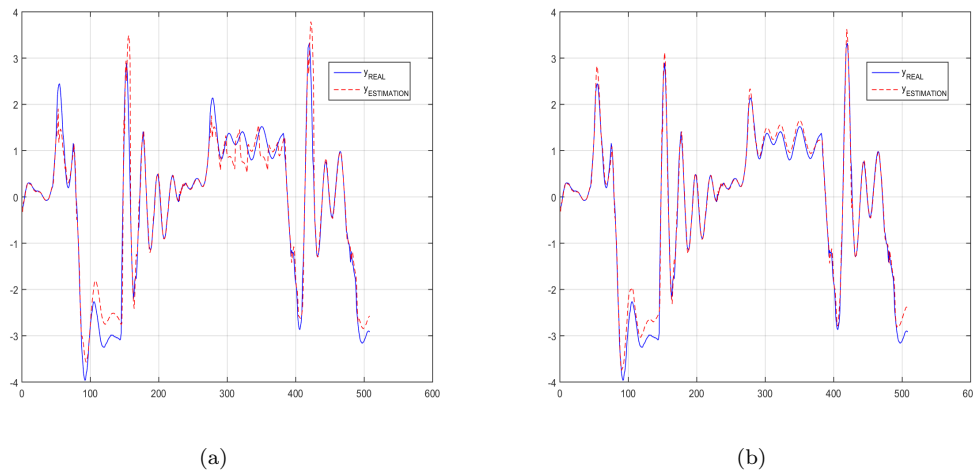


Figura 4: Predição da saída usando para o modelo RBF treinado com 15% de *outliers* (dados do atuador hidráulico): (a) Versão original e (b) versão robusta.

tivo com dados de sistemas dinâmicos artificiais e reais mostraram que a simples alteração da regra LMS por sua versão robusta proporcionou uma melhora considerável no desempenho dos modelos locais avaliados em cenários ricos em *outliers*. Este estudo também mostrou uma melhora considerável para todas as versões robustas dos modelos locais avaliados e que a melhora ficava ainda mais patente quando a proporção de *outliers* no conjunto de treinamento aumentava.

No momento estamos avaliando o desempenho de modelos locais robustos crescentes, ou seja, modelos em que a especificação *a priori* do número de neurônios não é necessária, pois neurônios vão sendo alocados com o passar do tempo, na medida que for necessário.

Agradecimentos

Os autores agradecem ao IFCE e ao CNPq (processo no. 309451/2015-9) pelo suporte financeiro a esta pesquisa.

Referências

- Aguirre, L. A. (2015). *Introdução à identificação de sistemas*, 4th edn, Editora UFMG.
- Barnes, C., Brown, S., Flake, G., Jones, R., O'Rourke, M. and Lee, Y. (1991). Applications of neural networks to process control and modeling, *Technical report*, Los Alamos National Lab., NM (USA).
- Barnett, V., Lewis, T. et al. (1994). *Outliers in statistical data*, Vol. 3, Wiley New York.
- Belz, J., Munker, T., Heinz, T. O., Kampmann, G. and Nelles, O. (2017). Automatic modeling with local model networks for benchmark processes, *IFAC-PapersOnLine* **50**(1): 470–475.
- Chan, S.-C. and Zhou, Y. (2010). On the performance analysis of the least mean m-estimate and normalized least mean M-estimate algorithms with gaussian inputs and additive gaussian and contaminated Gaussian noises, *Journal of Signal Processing Systems* **60**(1): 81–103.
- Costa, T. V., Fileti, A. M. F., Oliveira-Lopes, L. C. and Silva, F. V. (2015). Experimental assessment and design of multiple model predictive control based on local model networks for industrial processes, *Evolving Systems* **6**(4): 243–253.
- Foss, B. A. and Johansen, T. A. (1993). On local and fuzzy modelling, *Third International Conference on Industrial Fuzzy Control and Intelligent Systems (IFIS'93)*, pp. 80–87.
- Huber, P. (1964). Robust estimation of a location parameter, *The Annals of Mathematical Statistics* **35**(1): 73–101.
- Jacobs, R. A., Jordan, M. I., Nowlan, S. J. and Hinton, G. E. (1991). Adaptive mixtures of local experts, *Neural computation* **3**(1): 79–87.
- Johansen, T. A. and Foss, B. A. (1992). A NAR-MAX model representation for adaptive control based on local models, *Modeling, Identification and Control* **13**(1): 25.
- Jones, R. D., Lee, Y., Barnes, C., Flake, G., Lee, K., Lewis, P. and Qian, S. (1990). Function approximation and time series prediction with neural networks, *Proceedings of the 1990 IJCNN International Joint Conference*

- on *Neural Networks (IJCNN'1990)*, pp. 649–665.
- König, O., Hametner, C., Prochart, G. and Jakubek, S. (2014). Battery emulation for power-hil using local model networks and robust impedance control, *IEEE Transactions on Industrial Electronics* **61**(2): 943–955.
- Ljung, L. (1998). System identification, *Signal analysis and prediction*, Springer, pp. 163–173.
- Martinetz, T. M., Berkovich, S. G. and Schulten, K. J. (1993). 'neural-gas' network for vector quantization and its application to time-series prediction, *IEEE transactions on neural networks* **4**(4): 558–569.
- Mattos, C. L. C., Dai, Z., Damianou, A., Barreto, G. A. and Lawrence, N. D. (2017). Deep recurrent gaussian processes for outlier-robust system identification, *Journal of Process Control* **60**: 82–94.
- Moody, J. and Darken, C. J. (1989). Fast learning in networks of locally-tuned processing units, *Neural computation* **1**(2): 281–294.
- Moshou, D., De Keteiaere, B., Vrindts, E., Kennes, P., De Baerdemaeker, J. and Ramon, H. (1998). Local linear mapping neural networks for pattern recognition of plants, *IFAC Proceedings Volumes* **31**(12): 61–66.
- Münker, T., Heinz, T. O. and Nelles, O. (2017). Hierarchical model predictive control for local model networks, *Proceedings of the American Control Conference (ACC'2017)*, pp. 5026–5031.
- Murray-Smith, R. (1994). Local model networks and local learning, *Fuzzy Duisburg* **94**: 404–409.
- Murray-Smith, R. and Johansen, T. A. (1995). Local learning in local model networks, *Proceedings of the 4th International Conference on Artificial Neural Networks (ICANN'1995)*.
- Musavi, M. T., Ahmed, W., Chan, K. H., Faris, K. B. and Hummels, D. M. (1992). On the training of radial basis function classifiers, *Neural networks* **5**(4): 595–603.
- Ritter, H., Martinetz, T., Schulten, K., Barsky, D., Tesch, M. and Kates, R. (1992). *Neural computation and self-organizing maps: an introduction*, Addison-Wesley Reading, MA.
- Santos, J. D. A. and Barreto, G. A. (2017). Novel sparse LSSVR models in primal weight space for robust system identification with outliers, *Journal of Process Control*. In press.
- Sjöberg, J., Zhang, Q., Ljung, L., Benveniste, A., Delyon, B., Glorennec, P.-Y., Hjalmarsson, H. and Juditsky, A. (1995). Nonlinear black-box modeling in system identification: a unified overview, *Automatica* **31**(12): 1691–1724.
- Stokbro, K., Umberger, D. and Hertz, J. A. (1990). Exploiting neurons with localized receptive fields to learn chaos, *Complex Systems* **4**(3): 603–622.
- Walter, J., Ritter, H. and Schulten, K. (1990). Nonlinear prediction with self-organizing maps, *Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN'90)*, pp. 589–594.
- Widrow, B. (2005). Thinking about thinking: The discovery of the LMS algorithm, *IEEE Signal Processing Magazine* **22**(1): 100–106.
- Zou, Y., Chan, S.-C. and Ng, T.-S. (2000). Least mean M-estimate algorithms for robust adaptive filtering in impulse noise, *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing* **47**(12): 1564–1569.