Identification and Inverse Modelling of Industrial Systems Using Regional Models

Antonio G. P. Neto [*] **Luis G. M. Souza** [**]
Francisco de Assis da S. M. [***]

[*] *Department of Electrical Engineering, Federal University of Piauí, PI, (e-mail: galenoneto@ufpi.edu.br).*
[**] *Department of Electrical Engineering, Federal University of Piauí, PI, (e-mail: lmotasouza@ufpi.edu.br)*
[***] *Department of Production Engineering, Federal University of Piauí, PI, (e-mail: assismota@ufpi.edu.br)*

Abstract: With the advancement of technology the speed of industrial processes has greatly increased resulting in the need of obtaining models and controllers in a faster and more interactive way. Fortunately, the speed and ability to obtain data have also shown great advances, allowing the use of techniques capable of modeling processes reliably and quickly using the System Identification process. For generate a model from the input and output data of the systems, the System Identification has been the subject of many studies, with several techniques being proposed capable of generating reliable models in a short period of time. Two of these techniques, presented in this article, are the techniques known as Regional Models and Robust Regional Models, which use Clustering techniques such as Self Organizing Map (SOM) and K-means to dividing the system's data space into similar regions in order to produce more reliable models using supervised neural networks; the robust approach also performs the treatment of Outliers in the data using the M-Estimation technique. The techniques presented are applied in nonlinear industrial systems and evaluated based on their Normalized Mean Square Error (NMSE).

Keywords: System Identification, Neural Networks, ELM, OLS, Regional Models.

1. INTRODUCTION

Currently, modeling process has demanded even more advanced techniques in order to produce faithful models, better physical knowledge and for the realization of stable and precise control of processes. However, the demand is also for these models to be obtained quickly, especially in industrial environments where the production process must remain on the maximum possible scale, making the white box technique often impracticable.

An alternative to these challenges is system identification, also known as black box modeling because only the input and output data of the system are known, the information about the physics of the system is basically zero. Or, in some cases, gray box modeling is extremely viable, where the scientist has some physical information from the system and can improve the model obtained.

Due to the growth over the years of hybrid and nonlinear industrial processes, it is necessary to implement systems identification methods that meet the needs of modeling and controlling processes in a fast and effective way. Considering these factors, new and improved algorithms appear in the research that seek to meet the demand presented above, using hybrid methods and nonlinear modeling techniques (Neural Networks, Fuzzy Systems, Genetic Algorithm, etc.).

In short, the established algorithms use techniques capable of providing global models, which represent the system as a whole, generating a single regression model, or local models, characterized by the presence of several regression models for the same system or process - usually using clustering techniques for such purposes. Global models are more easily found in the literature, however, the interest in local models has increased due to their ease in characterizing systems that have different operating points or many nonlinearities. Obviously the best way to choose a model or modeling technique is to carry out research and tests.

Another approach that was recently proposed in the work of Júnior et al. (2015), is the use of regional models. The approach consists of using local modeling algorithms, performing a clustering at two levels Vesanto and Alhoniemi (2000), in order to separate the studied system in regions of more similar data and that can generate more reliable global regression models.

In the present work, linear and nonlinear regional models generated from the system identification process are used to characterize industrial system databases. The generated models, differently of the work that has proposed these techniques, are inverse models, obtaining the desired input for the generation of a certain output, a process that facilitates the subsequent control action of the systems. Robust models will also be produced in order to reduce

the effect of outliers in the modelling process, here is used the M-estimation method.

2. SYSTEM IDENTIFICATION

System identification is a process through which, provided only with the input information $u(k)$ and output $y(k)$ of a system, there is a mathematical model that, at least in an approximate way, can define a cause and effect relationship between them (Aguirre, 2014). This process consists of 5 steps, namely:

- dynamical tests and data collect;
- choose of mathematical representation to be used;
- determination of model structure;
- parameter estimation;
- model validation.

Exemplifying the process (Ljung, 1999)(Júnior et al., 2015), it is assumed that a system can be represented by the *autoregressive with exogenous inputs linear model* (ARX):

$$y(k) = \sum_{j=1}^{p} a_j y(k-j) + \sum_{l=1}^{q} b_l u(k-l). \tag{1}$$

After the stage of testing and collecting the data, it is desired to obtain the parameters $a_j \in R$ and $b_l \in R$. Thus, the input matrix $\mathbf{x}$(k) containing the data and the parameter vector θ is defined:

$$\mathbf{x}(k) = [y(k-1)...y(k-p)|u(k-1)...u(k-q)]^T, \tag{2}$$

$$\theta = [a_1...a_p|b_1...b_q]^T, \tag{3}$$

thus, the ARX model presented in (1) can be rewritten as:

$$y(k) = \theta^T \mathbf{x}(k). \tag{4}$$

Looking at the equation (4) it is noticed that, in order to arrive at a satisfactory regression model, a good estimation of the parameters θ is necessary. A common and well accepted technique among the scientific community is that of *Ordinary Least Squares* (OLS), some of its more recent applications and studies can be found at Nascimento et al. (2010), Wang and Lang (2016) and Huang (2018). There is versions of this technique in Batch or Recursive form, using the Kalman filter as in Borjas and Garcia (2011), or even robust versions of Least Squares, as will be presented in another topic of this article.

Using OLS, it is possible to arrive at the value of the parameters through the equation 5:

$$\theta = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}, \tag{5}$$

the matrix $\mathbf{X}$ is represented as shown in (6)

$$\mathbf{X} = \begin{bmatrix} y(p) & ... & y(1) & u(p) & ... & u(p-q+1) \\ y(p+1) & ... & y(2) & u(p+1) & ... & u(p-q+2) \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ y(N-1) & ... & y(N-p) & u(N-1) & ... & u(N-q) \end{bmatrix}, \tag{6}$$

where N represents the total number of input-output pairs used in the estimation process.

The nonlinear approach of the model presented in (1) constitutes a *nonlinear autoregressive model with exogenous inputs* (NARX), shown in (7)

$$\begin{aligned} y(k) &= f[y(k-1), ..., y(k-p); u(k-1), ..., u(k-q)|\Theta], \\ &= f[\mathbf{x}(k)|\Theta], \end{aligned} \tag{7}$$

where $f(\cdot)$ represents a nonlinear mapping of the system.

The parameters and mapping of the NARX model can be obtained using several computational intelligence approaches that have been studied and improved over time, due to the need to better understand and control a large demand for nonlinear dynamic systems. Among the methods used, we can mention neural networks (Billings and Chen, 1992) (Lawrence et al., 1996) (Barreto and Araújo, 2004), Takagi-Sugeno-Kang fuzzy models (Takagi and Sugeno, 1985) (Kim et al., 1998) (Rezaee and Zarandi, 2010) and hybrid models (Chen and Xi, 1998) (Azeem et al., 2000) (Babuška and Verbruggen, 2003) (Lima et al., 2007).

3. REGIONAL MODELLING

Global and local models commonly raise concerns about the capacity of generalization present in their regression curves. The concern regarding global models is the occurrence of overfitting, due to excessive training of the models. The concern with local models, on the other hand, is due to the fact that, often, models are stuck to a local minimum, without reaching the global minimum of the system. Joining the two approaches in a model that can be considered hybrid, an alternative to the problems presented is arrived at, this alternative is the regional models (Júnior et al., 2015).

This technique was developed from the 2-level clustering technique (Vesanto and Alhoniemi, 2000) and aims to partition the global data space into similar smaller spaces, in order to obtain global models for these regions, using neural networks as *Extreme Learning Machine* (ELM) (Huang et al., 2006) or *Multilayer Perceptron* (MLP) (Rumelhart et al., 1986), so that the system as a whole can be more faithfully represented. Here, the clustering techniques used will be the *Self-Organizing Map* neural network (SOM) (Kohonen, 2013) and the K-means algorithm, however other techniques such as *Winner Takes All* (WTA), *Radial Basis Function* (RBF), etc. can be used.

First of all, hyper-parameters must be defined, such as regression orders p e q, the value of the SOM prototypes C and the maximum number of regions K_{max}, being

indicated here as possible values $C = 5\sqrt{N_1}$ and $K_{max} = \sqrt{C}$, being N_1 the number of samples intended for training. After defining the hyper-parameters, the method used to generate regional models is basically five steps.

Step 1: SOM Training In this step, given an input vector $\mathbf{x}$, training of the SOM network is carried out using the C prototype vectors defined and the partition of the Voronoi cells, according to the rule presented in (8),

$$V_i = \{\mathbf{x} \in R^{p+q} | \parallel \mathbf{x} - \mathbf{w}_i \parallel \leq \parallel \mathbf{x} - \mathbf{w}_j \parallel, \forall j = 1, ..., C, j \neq i\}. \quad (8)$$

Step 2: Clustering of the SOM After SOM training, it is necessary to partition the prototype vectors into K regions. In this stage, cluster validation techniques can be used, such as Davies-Bouldin (DB) (Davies and Bouldin, 1979) to reach an optimum value of regions K_{opt}, among the values of $K = 1, ..., K_{max}$.

$$K_{opt} = \arg\min_{K=1,...,Kmax} DB(\mathbf{W}, \mathbf{P}^K), \quad (9)$$

where $\mathbf{P}^K$ denotes the set of K prototype vectors.

Step 3: Partitioning SOM prototypes into regions From the value found K_{opt}, SOM prototypes $\mathbf{w}_i$ must be distributed between the prototypes of the K-means algorithm $\mathbf{p}_r$, $r = 1, ..., K_{opt}$, following the rule given in (10):

$$\mathbf{W}_r = \{\mathbf{wi} \in R^{p+q} | \parallel \mathbf{w}_i - \mathbf{p}_r \parallel < \parallel \mathbf{w}_i - \mathbf{p}_j \parallel, \forall j = 1, ..., C, j \neq r\}. \quad (10)$$

Step 4: Mapping data points to regions. The fourth step is to map the training data to the defined regions, assigning the data to the region that the nearest SOM prototype belongs to the partition $\mathbf{W}_r$, thus generating the regions $\{\mathbf{X}_1\}$, $\{\mathbf{X}_2\}$,...,$\{\mathbf{X}_r\}$, that will be used to generate the regression models.

Step 5: Building regression models over the regions Finally, for each region, the necessary parameters for the construction of the models are calculated. The OLS method can be used directly to arrive at linear models of the system by calculating the parameter vector Θ_r for each region as shown in (11),

$$\Theta_r = (\mathbf{X}_r^T \mathbf{X}_r)^{-1} \mathbf{X}_r \mathbf{y}_r, \quad (11)$$

or, for a nonlinear approach, the ELM or MLP network can be used. Some uses of ELM and MLP in system identification may be seen in Soria-Olivas et al. (2011), Zhang et al. (2017), Zhao and Zhang (2009), Bhushan et al. (2012) and Rasouli et al. (2013).

4. M-ESTIMATION: ROBUST MODELS

Generally, when using regression algorithms, the same contribution value is assigned to all error samples. However, the error value can be generated from training data with outliers, which negatively affects the ability to generalize the algorithms. This problem can be solved by removing the outliers present in the data, but this process can be exhausting and complicated. A viable alternative is the realization of the so-called *robust regression*, using estimation methods that are not so sensitive to outliers, allowing the adaptation and re-weighting of algorithms such as ELM and OLS (Barreto and Barros, 2016)(Júnior et al., 2015).

The robust regression process is based on the M-estimation technique proposed by Huber in Huber (1964) - where M is related to the "maximum likehood", which allows to increase the robustness of the model by minimizing a cost function different from the sum of the quadratic errors and which allows the weighting and filtering of the contribution of each error to the function. Applying the technique to any OLS algorithm, you can increase the robustness of the algorithm by following the steps below:

Step 1 Generate an initial parameter array $\hat{\theta}(0)$ through the standard use of the OLS algorithm.

Step 2 At each t iteration of the algorithm, collect the residuals from the past iteration $e_n^{(t-1)}$, with $n = 1, ..., N_1$ and then store the values of the weights generated $w_n^{(t-1)} = w(e_n^{(t-1)})$. The value of the weights can be computed from several functions, such as Andrews, Cauchy, an example is that of Huber,

$$w(e_n) = \begin{cases} \frac{k_e}{e_n} \text{ if } |e_n| > k_e \\ 1 \text{ otherwise.} \end{cases} \quad (12)$$

The constant k_e is a tuning value, so it is understood that from the given condition $|e_n| > k_e$, the higher the value of the residue, the lower the value of the weight.

Step 3 In the last step is needed to solve the weighted least squares equation and find the vector of parameters

$$\hat{\theta}^{(t)} = (\mathbf{X}^T \mathbf{B}^{(t-1)} \mathbf{X})^{-1} \mathbf{X}^T \mathbf{B}^{(t-1)} \mathbf{y}, \quad (13)$$

where $\mathbf{B}^{(t-1)} = \text{diag}(w_n^{(t-1)})$ it's a matrix $N_1 \times N_1$.

This algorithm applied to OLS is called *iteratively reweighted least-squares* (IRLS), and also can be used in MLP, ELM, SOM (Fox and Weisberg, 2010) etc.

5. EXPERIMENTS AND RESULTS

In the research, four data sets obtained from the DaISy repository (Database for the Identication of Systems) were used, two industrial systems and two mechanical systems: *Hydraulic Actuator*, *Robot Arm*, *Heat Exchanger*, *Continuous Stirred Tank Reactor* (CSTR). The sets studied here have their characteristics presented and were modeled and analyzed using local models in the work of Souza (2012).

In the study carried out, 4 different techniques were used for the process of identifying the data sets. First, regional models were obtained using K_{opt} linear models obtained using the OLS technique, a method called the *Regional Linear Model* (RLM). Nonlinear models were also built using ELM algorithm, chosen in short due to its

speed of learning, since it does not require long training cycles such as the MLP network, this method was given the name *Regional Extreme Learning Machine* (RELM) (Júnior et al., 2015).

The RLM and RELM methods were also subjected to the IRLS technique, in order to reduce the influence of outliers on them through the interactive treatment that is performed. The robust regional models are then called Robust RLM (R2LM) and Robust RELM (R2ELM). The regression models generated are of the ARX type, and inverse regression models are built, that is, the result obtained through the tests performed is the system input necessary to obtain certain desired output values, thus facilitating the action of control.

For the training stage, an initial learning rate $\alpha_0 = 0.1$ and final $\alpha_T = 0.01$ were used as hyper-parameters for the SOM network. The neighborhood function chosen was Gaussian, having as initial neighborhood radius $\sigma_0 = 10$ and final $\sigma_T = 0.01$. The number of prototype vectors for each data set was defined in approximate values of $5\sqrt{N}$ and 50 training epochs were held. The data were normalized in the interval [-1,+1].

The value of K_{opt} and the number of neurons present in the hidden layer h used in the ELM network were defined through the training and validation process, ranging from K to a value of K_{max}, this being defined as an approximate value of $\sqrt{C}$. The number of neurons in hidden layer was varied from 10 to 100 and the tangent-hyperbolic activation function was used for the network. The data separation process for training, validation and testing had the following proportion: training data = 50%; validation data = 30%; test data = 20%.

The comparison between the algorithms for the same data set was performed through the analysis of the generalization error, comparing the value of the Normalized Mean Squared Error (NMSE):

$$\mathbf{NMSE} = \frac{\sum_{t=1}^{N_2} e^2(t)}{N_2 \widehat{\sigma}_u^2} \tag{14}$$

being N_2 the number of samples of the test set, $e(t)$ the predction error calculated as $e(t) = u(t) - \widehat{u}(t)$ and $\widehat{\sigma}_u^2$ the variance of the original test time series.

Next, the parameters and results for each data set will be presented, as well as the presentation of the residual analysis through the graph of the autocorrelation function.

5.1 Hydraulic Actuator

For the hydraulic actuator set, were selected the memory orders used by Sjöberg et al. (1995): p = 3 and q = 2. For nonlinear models based on ELM (RELM and R2ELM) the number of neurons used in the hidden layer was $h = 20$ and the value of K_{opt} obtained for all models was 2.

As can be seen from the quantitative analysis, the R2LM algorithm presented the best performance for the set, with a result very close to that of the RLM, but with less variance of the data.

Table 1. Results Hydraulic Actuator

Used Models	**NMSE**	
	Mean	*Variance*
R2ELM	0.0012	1.19e-006
RELM	0.0015	1.89e-006
R2LM	9.34e-005	2.32e-015
RLM	9.34e-005	2.41e-015

It is worth mentioning that in the work of Júnior et al. (2015), the analysis of the mean square error was also performed for the same data set, however, a direct model was generated, without the use of the inverse modeling technique as in this paper.

In Fig. 1 can be seen the regression curve of predicted data for the best case and the test set that originated it.

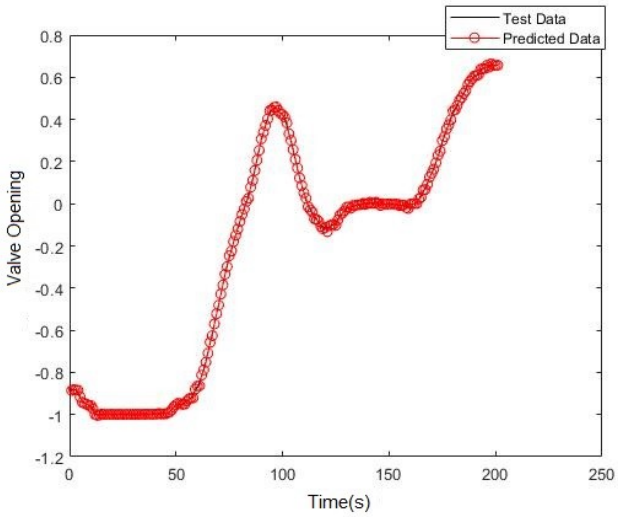

Figure 1. Regression Curve Hydraulic Actuator.

5.2 Robot Arm

For the robotic arm set, the following memory orders were selected: p = 5 and q = 4. For nonlinear models based on ELM (RELM and R2ELM) the number of neurons used in the hidden layer was $h = 100$ and the value of K_{opt} obtained for all models was 1.

Table 2. Results Robot Arm

Used Models	**NMSE**	
	Mean	*Variance*
R2ELM	0.0204	1.79e-005
RELM	0.0202	1.18e-005
R2LM	0.0042	7.78e-037
RLM	0.0042	3.11e-036

As can be seen through the analysis of the NMSE, the R2LM algorithm achieved the best results, considering the lowest rate of variance for it. In Fig. 2 can be seen the regression curve of predicted data for the best case and the test set that originated it.

5.3 Heat Exchanger

For the heat exchanger set, were used the memory orders: p = 6 and q = 3. For nonlinear models based on ELM,

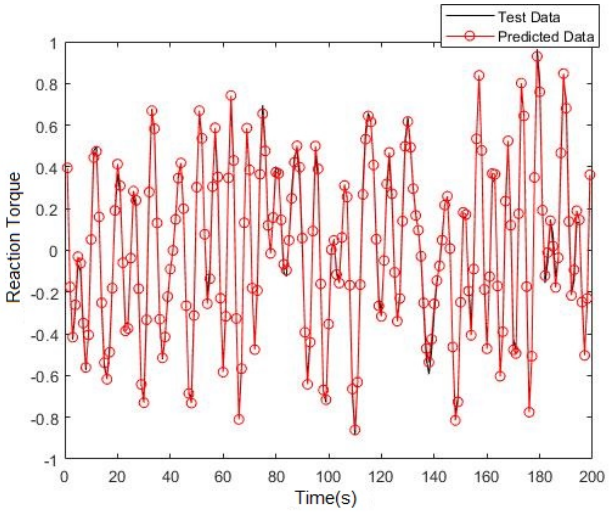

Figure 2. Regression Curve Robot Arm.

the number of neurons in the hidden layer used for the R2ELM algorithm was $h = 100$ e for the RELM algorithm it was $h = 70$. The value of K_{opt} obtained for all models was 4.

Table 3. Results Heat Exchanger

Used Models	NMSE	
	Mean	*Variance*
R2ELM	0.3611	0.0016
RELM	0.3527	7.32e-004
R2LM	0.3493	8.99e-007
RLM	0.3497	2.81e-007

As can be seen from the quantitative analysis, the R2LM algorithm presented the best performance for the set. In Fig. 3 can be seen the regression curve of predicted data for the best case and the test set that originated it.

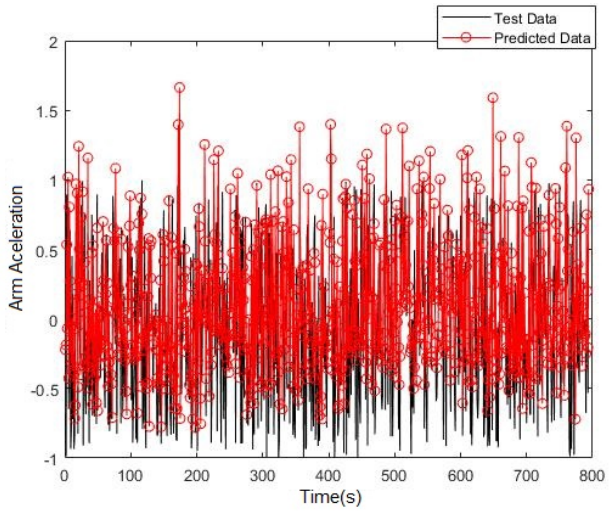

Figure 3. Regression Curve Heat Exchanger.

5.4 CSTR

The memory orders selected here were p = 4 and q = 3. For nonlinear models based on ELM (RELM and R2ELM) the number of neurons used in the hidden layer was $h = 60$ and $K_{opt}value = 3$. For the linear models (R2LM and RLM) the optimal number of regions found was $K_{opt} = 4$.

Table 4. Results CSTR

Used Models	NMSE	
	Mean	*Variance*
R2ELM	0.0314	2.15e-005
RELM	0.0315	2.54e-005
R2LM	0.0240	8.63e-011
RLM	0.0240	8.37e-011

As can be seen from the quantitative analysis, the R2LM algorithm presented the best performance for the set, with a result very close to that of the RLM, but with less variance of the data. In Fig. 4 can be seen the regression curve of predicted data for the best case and the test set that originated it.

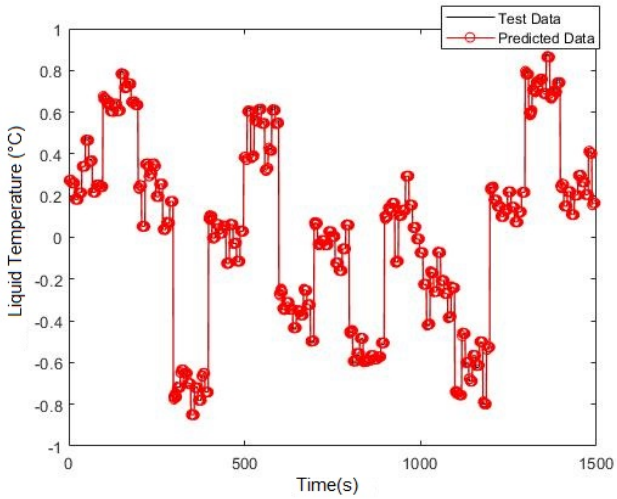

Figure 4. Regression Curve CSTR.

6. CONCLUSION

The use of regional models aims especially to understand and analyze certain more similar data spaces in a given set, being able to treat them in a similar way, avoiding analyzing them in a very generally or very sampled way, at the risk of losing certain characteristics.

Therefore, the models obtained for the data sets presented satisfactory results, considering a general panorama and also if compared to other models obtained for the same data, such as those of the work of Souza (2012), showing that the regional models presented are a via interesting to be explored and are able to compete with other systems identification algorithms.

In all the systems presented, the linear models were the ones that stood out the most, providing lower and more robust NMSE values, proving to be reliable even in systems with nonlinear characteristics, a process facilitated by the approach in regions.

It is important to note that during the process of validating the algorithms, it was noticed that the division of systems in many regions, often worsened the results of the models, often generating high values of NMSE, or generating

prototype vectors that were unable to represent any data of the sets.

In the future, it is intended to expand the scope of the models obtained, using other vector quantization algorithms in addition to the SOM network, also replacing the ARX regression model used by a NARX model. The authors of the work Júnior et al. (2015), are also developing adaptive variants of the algorithms (Fritzke, 1995), allowing an online version of regional models.

REFERENCES

Aguirre, L.A. (2014). *Introdução à Identificação de Sistemas: Técnicas Lineares e Não Lineares: Teoria e Aplicação*. Editora UFMG, Minas Gerais.

Azeem, M.F., Hanmandlu, M., and Ahmad, N. (2000). Generalization of adaptive neuro-fuzzy inference systems. *IEEE Transactions on Neural Networks*, 11(6), 1332–1346.

Babuška, R. and Verbruggen, H. (2003). Neuro-fuzzy methods for nonlinear system identification. *Annual Reviews in Control*, 27, 73–85.

Barreto, G.A. and Araújo, A.F.R. (2004). Identification and control of dynamical systems using the self-organizing map. *IEEE Transactions on Neural Networks*, 15, 1244–1259.

Barreto, G.A. and Barros, A.L.B. (2016). A robust extreme learning machine for pattern classification with outliers. *Neurocomputing*, 176, 3–13.

Bhushan, B., Singh, M., and Hage, Y. (2012). Identification and control using mlp, elman, narxsp and radial basis function networks: a comparative analysis. *Artificial Intelligence Review*, 37, 133–156.

Billings, S.A. and Chen, S. (1992). *Neural Netwoks for Systems and Control: Neural networks and system identification*, chapter 9, 181–205. Peter Peregrinus, London.

Borjas, M.D.S. and Garcia, C. (2011). Subspace identification for industrial processes. *TEMA Tend. Mat. Apl. Comput.*, 12:183–194.

Chen, J.Q. and Xi, Y.G. (1998). Nonlinear system modeling by competitive learning and adaptive fuzzy inference system. *IEEE Transactions on Systems*, 28, 231–238.

Davies, D.L. and Bouldin, D.W. (1979). A cluster separation measure. *IEEE Trans. Pattern Anal. Mach. Intell.*, 1, 224–227.

Fox, J. and Weisberg, S. (2010). *An R Companion to Applied Regression*. Sage Publication, Thousand Oaks, 2nd edition.

Fritzke, B. (1995). Growing grid—a self-organizing network with constant neighborhood range and adaptation strength. *Neural Process. Lett.*, 2, 9–13.

Huang, F.L. (2018). Multilevel modeling and ordinary least squares regression: How comparable are they? *The Journal of experimental education*, 86(2), 265–281.

Huang, G.B., Zhu, Q.Y., and Ziew, C.K. (2006). Extreme learning machine: Theory and applications. *Neurocomputing*, 1-3, 489–501.

Huber, P.J. (1964). Robust estimation of a location parameter. *Ann. Math. Stat.*, 35, 73–101.

Júnior, A.H.S., Barreto, G.A., and Corona, F. (2015). Regional models: A new approach for nonlinear system identification via clustering of the self-organizing map. *Neurocomputing*, 147, 31–46.

Kim, E., Lee, H., Park, M., and Park, M. (1998). A simply identified sugeno-type fuzzy model via double clustering. *Information Sciences*, 110, 25–39.

Kohonen, T. (2013). Essentials of the self-organizing map. *Neural Networks*, 37, 52–65.

Lawrence, S., Tsoi, A.C., and Back, A.D. (1996). Function approximation with neural networks and local methods: Bias, variance and smoothness. *Australian Conference on Neural Networks (ACNN)*, 16–21.

Lima, C.A.M., Coelho, A.L.V., and Zuben, F.J.V. (2007). Hybridizing mixtures of experts with support vector machines: Investigation into nonlinear dynamic systems identification. *Information Sciences*, 177, 2049–2074.

Ljung, L. (1999). *System Identification: Theory for the User*. Prentice-Hall, Englewood Cliffs, NJ, 2nd edition.

Nascimento, R.S., Froes, R.E., Silva, N.O.C.E., Naveira, R.L.P., Mendes, D.B.C., Neto, W.B., and Silva, J.B.B. (2010). Comparison between ordinary least squares regression and weighted least squares regression in the calibration of metals present in human milk determined by icp-oes. *Talanta (Oxford)*, 80(3), 1102–1109.

Rasouli, H., Rasouli, C., and Koohi, A. (2013). Identification and control of plasma vertical position using neural network in damavand tokamak. *Review of Scientific Instruments*, 84.

Rezaee, B. and Zarandi, M.F. (2010). Data-driven fuzzy modeling for takagi–sugeno–kang fuzzy system. *Information Sciences*, 180, 241–255.

Rumelhart, D.E., Hinton, G.E., and Williams, R. (1986). Learning representations by back-propagating errors. *Nature*, 323, 533–536.

Sjöberg, J., Zhang, Q., Ljung, L., Benveniste, A., Deylon, B., Glorennec, P.Y., Hjalmarsson, H., and Juditsky, A. (1995). Non-linear black-box modeling in system identification: a unified overview. *Automatica*, 31, 1691–1724.

Soria-Olivas, E., Gomez-Sanchis, J., Martin, J.D., Vila-Frances, J., Martinez, M., Magdalena, J.R., and Serrano, A.J. (2011). Belm: Bayesian extreme learning machine. *IEEE Transactions on Neural Networks*, 22, 505–509.

Souza, L.G.M. (2012). *Modelos Lineares Locais Para a Identificação de Sistemas Dinâmicos Usando Redes Neurais Competitivas*. Ph.D. thesis, Universidade Federal do Ceará.

Takagi, T. and Sugeno, M. (1985). Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man, and Cybernetics*, 15, 116–132.

Vesanto, J. and Alhoniemi, E. (2000). Clustering of the self-organizing map. *IEEE Trans. Neural Netw.*, 11, 586–600.

Wang, X. and Lang, C. (2016). High dimensional ordinary least squares projection for screening variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 78(3), 589–611.

Zhang, C., Liub, Q., Wuc, Q., Zhengb, Y., Zhoub, J., Tub, Z., and Chan, S.H. (2017). Modelling of solid oxide electrolyser cell using extreme learning machine. *Electrochimica Acta*, 251, 137–144.

Zhao, H. and Zhang, J. (2009). Nonlinear dynamic system identification using pipelined functional link artificial recurrent neural network. *Neurocomputing*, 72, 13–15.