Hybrid Method Based on NARX models and Machine Learning for Pattern Recognition

Pedro H. O. Silva * **Augusto S. Cerqueira** *
Erivelton G. Nepomuceno **

* *Department of Electrical Engineering, Federal University of Juiz de Fora (UFJF), Juiz de Fora, MG, Brazil, (e-mail: silva.pedro@engenharia.ufjf.br, augusto.santiago@ufjf.edu.br).*
** *Department of Electrical Engineering, Federal University of São João del-Rei (UFSJ), São João del-Rei, MG, Brazil, (e-mail: nepomuceno@ufsj.edu.br).*

Abstract: This work presents a novel technique that integrates the methodologies of machine learning and system identification to solve multiclass problems. Such an approach allows to extract and select sets of representative features with reduced dimensionality, as well as predicts categorical outputs. The efficiency of the method was tested by running case studies investigated in machine learning, obtaining better absolute results when compared with traditional classification algorithms.

Resumo: O presente trabalho apresenta uma nova técnica que integra as metodologias de aprendizado de máquinas e identificação de sistemas na solução de problemas multiclasses. A abordagem permite extrair e selecionar conjuntos de características representativas com dimensionalidade reduzida, da mesma forma que prediz saídas categóricas. A eficiência do método é testada pela aplicação em estudos de casos estudados no aprendizado de máquina, obtendo melhores resultados absolutos em comparação aos algoritmos clássicos de classificação.

Keywords: machine learning; system identification; NARX model; feature extraction; dimensionality reduction.

Palavras-chaves: aprendizado de máquina; identificação de sistemas; modelos NARX; extração de características; redução de dimensionalidade.

1. INTRODUCTION

The progressive development of modern technology, comprised of computer and internet applications, generates large amounts of data at an unprecedented speed, such as videos, photos, texts, voices, and data obtained from the emergence of the Internet of Things (IoT) and cloud computing. Data often have large attributes, presenting sets of redundant, noisy, and irrelevant features that can degrade the performance of machine learning algorithms, posing a major challenge for data analysis and decision making. Therefore, there is a need to use techniques that allow reducing the dimensionality of data, which is a step that helps data mining and machine learning algorithms to be more efficient (Brunton and Kutz, 2019).

The dimensionality reduction problem can be solved using feature extraction techniques. Feature extraction deals with the problem by generating a new reduced set of features with k dimensions, coming from combinations of the original set with d dimensions (Cai et al., 2018). The new reduced dataset has high discriminatory power, which can increase algorithm performance, reduce processing time, and simplify results. Furthermore, in some cases, feature extraction promotes an increase in the understanding of the results and leads to an improvement in precision, as it avoids excessive adjustments to the data sample.

In machine learning, conventional classifiers largely lack processes to handle overfitting more efficiently. Therefore, if the input variables (features) have a larger number compared to the number of training data, in some cases it can result in complex and ineffective models. Basically, the generalizability of the classifier may not be enough, being necessary to extract and select features to improve the generalizability. In recent decades, several feature extraction algorithms have been created and employed, being Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) the most widely known (Abdi and Williams, 2010).

Parameter extraction methods (Jiménez et al., 2019) assist in data pre-processing, and are later used to increase the performance of traditional classification algorithms such as Support Vector Machine (SVM) (Nagata et al., 2020), Neural Network (NN) (Naik and Kundu, 2014), Random Forest (RF) (Zhang et al., 2003) and K-Nearest Neighbors (KNN) (Pan et al., 2017). Classification algorithms are widely used, however, they have some disadvantages, such as the dependence on auxiliary extraction algorithms to obtain satisfactory performance on certain types of data. Furthermore, the models resulting from the classifiers generally have low interpretability, i.e., understanding the relationship between the inputs and outputs of the predictor may not be simple. Even though classifiers are very

efficient in solving several problems, the performance of a classifier depends on the nature of the data to be classified (Silva et al., 2020). Since there is no single classifier that works best for all the problems provided.

Thus, this work proposes a hybrid algorithm that integrates machine learning and system identification methodologies in solving multinomial classification problems. One of the main motivations is the combination of methods and techniques from different areas in order to improve their performance. As a result, combinations are generated that usually provide more efficient hybrid algorithms. The approach allows to extract and select sets of representative features with reduced dimensionality, in the same way as it efficiently predicts categorical outputs. Furthermore, the algorithm allows the inclusion of lagged terms directly and deals with multicollinearity, resulting in more interpretable models, something that is not achievable using other popular classification techniques.

2. NONLINEAR SYSTEM IDENTIFICATION

System Identification is an experimental approach that aims to identify and adjust a mathematical model of a system, based on experimental data that record the behavior of system inputs and outputs (Billings, 2013; Aguirre, 2007). In particular, the interest in nonlinear system identification has received a lot of attention from researchers since the 1950s and many relevant results have been developed (Wei et al., 2004; Nepomuceno and Martins, 2016; Ferreira et al., 2017). A model representation constantly employed is the NARX model (*Nonlinear AutoRegressive with eXogenous inputs*), consisting of a mathematical model based on differential equations. In general, system identification consists of several steps, including data collection and processing, choice of mathematical representation, determination of model structure, parameter estimation, and model validation (Söderström and Stoica, 1989). For nonlinear systems, there are numerous techniques to determine the structure of the model, such as: Clustering Algorithms (Aguirre and Jácome, 1998), Genetic Programming (Sette and Boullart, 2001) and the method Orthogonal Forward Regression (OFR) using the Error Reduction Ratio (ERR) approach (Wei et al., 2004).

2.1 NARX Representation

The NARX representation is a discrete-time model that explain the output value $y(k)$ as a function of previous values of the output and input signals:

$$\begin{aligned} y(k) = f^l(&y(k-1), \cdots, y(k-n_y), \\ &u(k-1), \cdots, u(k-n_u)) + e(k), \end{aligned} \tag{1}$$

where f^l represents a nonlinear function of the model with nonlinearity degree $l \in \mathbb{N}$, $y(k) \in \mathbb{R}$ is the output of the system, and $u(k) \in \mathbb{R}$ is the input to the system in discrete time $k = 1, 2, \ldots, N$; N is the number of observations, $e(k) \in \mathbb{R}$ represents the uncertainties and possible noise in discrete time k, $n_y \in \mathbb{N}$ and $n_u \in \mathbb{N}$ describes the maximum lags for the output and input sequences, respectively. Most approaches assume that the function f^l can be approximated by a linear combination of a predefined set of functions $\phi_i(\varphi(k))$, so that Equation (1) can be expressed in the following parametric form:

$$y(k) = \sum_{i=1}^{m} \theta_i \phi_i(\varphi(k)) + e(k), \tag{2}$$

where θ_i are the coefficients to be estimated, $\phi_i(\varphi(k))$ are the predefined functions that depend on the regression vector:

$$\begin{aligned} \varphi(k) = [&y(k-1), \cdots, y(k-n_y), \\ &u(k-1), \cdots, u(k-n_u)]^\mathsf{T}, \end{aligned} \tag{3}$$

where $\varphi(k)$ represents the previous outputs and inputs, and m is the number of functions in the set. One of the most used NARX models is the polynomial representation, where Equation (2) can be denoted as follows:

$$\begin{aligned} y(k) = \theta_0 + \sum_{i_1=1}^{n} \theta_{i_1} x_{i_1}(k) + \sum_{i_1=1}^{n} \sum_{i_2=i_1}^{n} \theta_{i_1 i_2} x_{i_1}(k) x_{i_2}(k) + \\ \sum_{i_1=1}^{n} \cdots \sum_{i_l=i_l-1}^{n} \theta_{i_1 i_2 \cdots i_l} x_{i_1}(k) x_{i_2}(k) \cdots x_{i_l}(k) + e(k), \end{aligned} \tag{4}$$

considering $n = n_y + n_u$,

$$x_i(k) = \begin{cases} y(k-i), & 1 \le i \le n_y, \\ u(k-i+n_y), & n_y + 1 \le i \le n, \end{cases} \tag{5}$$

being l the nonlinearity degree. The NARX model of order l means that the order of each term in the model is not greater than l. The total number of potential terms in a polynomial NARX model is given by:

$$M = \frac{(n+l)!}{n! \cdot l!}. \tag{6}$$

Finally, NARX models can be used to describe a wide variety of systems, simply obtaining analytical information about dynamic models. Another advantage is parsimony, meaning that a wide range of behaviors can be concisely represented using just a few terms from the vast search space formed by candidate regressors, as well as a small data set is needed to estimate a model, which can be crucial in applications where it is difficult to acquire a large amount of data.

2.2 Orthogonal Forward Regression

In general, the determination of the model structure and parameter estimation are performed together. One of the most popular algorithms for performing the two-step NARX modeling is the Orthogonal Forward Regression (OFR) algorithm (Billings, 2013) . The algorithm transforms a set of candidate terms into orthogonal vectors and classifies them based on their contribution to the output data, identifying and fitting a deterministic and parsimonious NARX model that can be expressed in a form of generalized linear regression. The original Orthogonal Forward Regression algorithm uses the Error Reduction Rate (ERR) as a dependency metric. The criterion associates to each candidate term an index corresponding to the contribution in explaining the variance of the system output data. The error reduction rate is defined as the

Pearson correlation coefficient $C(x,y)$ between two associated vectors x and y:

$$C(x,y) = \frac{(x^T y)^2}{(x^T x)(y^t y)}. \tag{7}$$

3. LOGISTIC-NARX MULTINOMIAL CLASSIFICATION

Classification problems occur in the most diverse areas of knowledge, such as finance, healthcare, and engineering, where the goal is to identify a model that is capable of classifying observations or measurements into different categories or classes. A widely used approach is logistic regression, which uses concepts of statistics and probability to categorize variables by classes. In the logistic regression method, the predicted values are probabilities, so they are restricted to values between 0 and 1, and use the logistic function defined as:

$$f(x) = \frac{1}{1 + \exp(-x)}, \tag{8}$$

which $x \in \mathbb{R}$ and $f(x)$ is restricted to the range between 0 and 1. A problem found in logistic regression is multicollinearity, in which independent variables have exact or approximately exact linear relationships. If the variables are highly correlated, inferences based on the regression model may be erroneous or unreliable.

In this work, a hybrid multinomial classification method is presented, which allows the extraction and selection of features during the process. One of the advantages of using NARX modeling methodologies is the orthogonalization procedures, which address the multicollinearity problem by verifying the correlations between the predictor variables. The method is based on the Orthogonal Forward Regression (Ayala Solares et al., 2019) algorithm that selects the terms and combines the logistic function with the NARX representation to obtain a probability model:

$$p(x) = \frac{1}{1 + \exp\left[\sum_{m=1}^{M} \theta_m \phi_m(\varphi(k))\right]}. \tag{9}$$

The multiclass problems are usually more complex than the binary classification, due to their decision boundaries. Direct extension of the binary algorithm to a multiclass version is not always possible or easy to accomplish. Therefore, the forms most explored by the scientific community are based on the binarization of multiclass problems. One of the most employed decomposition methods is One-Versus-All (OVA), which makes use of C binary classifiers to solve a classification problem involving C classes. For the v-th binary classifier, a distinction is made between the class w_v and the other classes. In this way, a x pattern is classified by the following decision rule:

$$x \in w_v \Leftrightarrow \underset{1 \le v \le C}{\arg\max}\ f_v(x), \tag{10}$$

defining f_v as the result given by the model referring to the class v, meaning the probability between 0 and 1 of belonging to the v-th class with respect to an instance x. Then, it is verified which class is most likely given as a result in Equation (10).

Algorithm 1 Logistic-NARX Multinomial

1: **Input:** $\{y(k), k = 1, \ldots, N\}$, $\mathcal{M} = \{\phi_i, i = 1, \ldots, m\}$, l, n_y, n_u, k
2: **Output:** $\boldsymbol{\alpha} = \{\alpha_i, i = 1, \ldots, k\}$, $\boldsymbol{\theta} = \{\theta_i, i = 1, \ldots, k\}$
3: **for** $i = 1 : m$ **do**
4: $\quad w_i \leftarrow \frac{\phi_i}{\|\phi_i\|_2}$
5: $\quad r_i \leftarrow$ Logistic regression accuracy in w_i and y
6: $j \leftarrow \arg\max_{1 \le i \le m} \{r(w_i, y)\}$
7: $q_1 \leftarrow w_j$
8: $\alpha_1 \leftarrow \phi_j$
9: Train logistic model with α_1 and y
10: Compute cross-validation
11: Remove ϕ_j from $\mathcal{M}$
12: **for** $s = 2 : k$ **do**
13: $\quad$ **for** $i = 1 : m$ **do**
14: $\quad\quad w_i^{(s)} \leftarrow$ Orthogonalize ϕ_i in $[q_1, \ldots, q_{(s-1)}]$
15: $\quad\quad$ **if** $w_i^{\mathsf{T}} w_i < 10^{-10}$ **then**
16: $\quad\quad\quad$ Remove ϕ_i from $\mathcal{M}$
17: $\quad\quad\quad$ Next iteration
18: $\quad\quad r_i \leftarrow$ Logistic regression accuracy in w_i and y
19: $\quad j \leftarrow \max_{1 \le i \le m-s+1} \{r^{(i)}(w_i, y)\}$
20: $\quad q_s \leftarrow w_j$
21: $\quad \alpha_s \leftarrow \phi_j$
22: $\quad$ Remove ϕ_j from $\mathcal{M}$
23: $\quad \boldsymbol{\alpha} \leftarrow [\alpha_1, \ldots, \alpha_{(s)}]$
24: $\quad$ Train logistic model with $\boldsymbol{\alpha}$ and y
25: $\quad$ Compute cross-validation
26: $\boldsymbol{\alpha} \leftarrow [\alpha_1, \ldots, \alpha_{(k)}]$ ▷ matrix of selected terms
27: $\boldsymbol{\theta} \leftarrow [\theta_1, \ldots, \theta_{(k)}]$ ▷ estimated coefficients vector

To combine the methodologies of NARX models and multinomial classification, some aspects of the Orthogonal Forward Regression algorithm were adapted. The OFR algorithm relies on the error reduction rate given by Equation (7) to determine the contribution of each candidate term. However, this metric is no longer useful as the output is a categorical variable due to the multiclass addressed. To solve this problem, a simple logistic model using maximum likelihood estimate (MLE) is used. Basically, the accuracy of the prediction of the categorical variable resulting from the logistic model based on continuous variables is calculated. Since the resulting predictor has a high degree of accuracy, it can be concluded that the two variables are correlated. To calculate the accuracy, the K-Fold Cross Validation was used, in order to assess the generalizability of the model.

In Algorithm 1, line (1) represents the inputs composed by the vector $y(k)$ of classes (labels), the matrix $\mathcal{M}$ constitutes the regressors formed by combinations of feature vectors, k is the maximum number of selected terms and

Table 1. Summary of the datasets.

Dataset	Classes	Features	Samples
Iris	3	4	150
Wine	3	13	178
Glass	6	9	214
Wave	3	40	5000

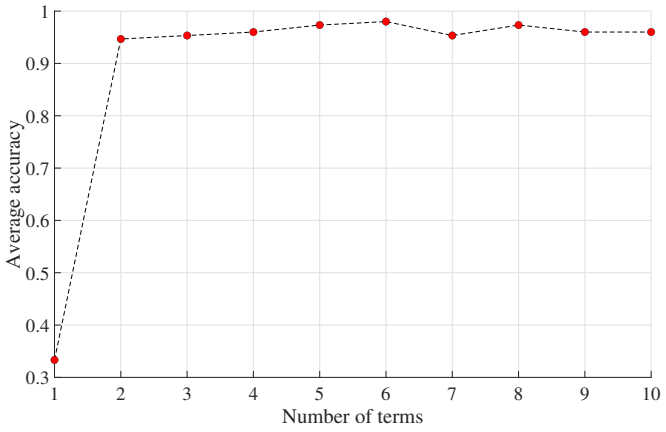

Figure 1. Relation between average accuracy and the number of selected terms associated with *Iris* data.

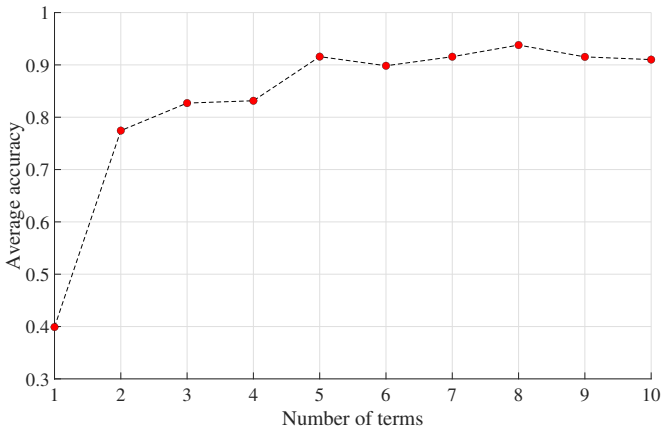

Figure 2. Relation between average accuracy and the number of selected terms associated with *Wine* data.

(l, n_y, n_u) are the parameters of the NARX model (Equation 1). Lines (3-8) aim to select the candidate terms ϕ_i with greater discriminatory power, based on the prediction accuracy of the logistic model. In lines (9-10) the logistic model is trained using the selected regressors α and the class vector $y(k)$, calculating the cross-validation of the classification. The selected terms are transformed at each step into a new group of orthogonal bases in lines (14-18), using the Gram-Schmidt Orthogonalization procedures. The process is repeated on lines (12-25) until it reaches the specified maximum number of model terms selection k. Finally, in lines (26-27) the matrix of selected terms $\boldsymbol{\alpha}$ and the vector of coefficients $\boldsymbol{\theta}$ of the estimated model are obtained. Since the number of terms in the model is not known in advance, the parameter k can be selected heuristically, executing the Algorithm 1 and checking the resulting accuracy curve.

4. RESULTS

In this section, simulations are carried out to evaluate the dimensionality reduction capacity and classification accuracy of the proposed method, with tests being carried out in traditional databases in the literature. Furthermore,

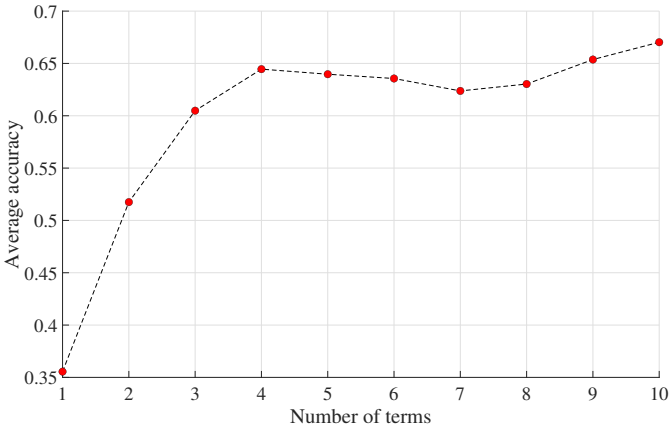

Figure 3. Relation between average accuracy and the number of selected terms associated with *Glass* data.

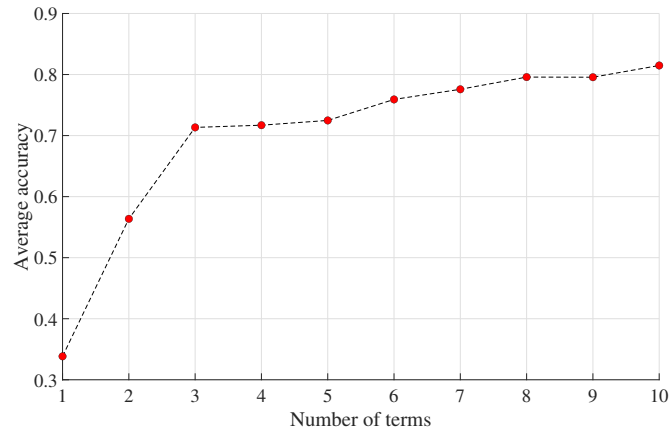

Figure 4. Relation between average accuracy and the number of selected terms associated with *Wave* data.

the effectiveness of the new methodology in comparison to popular classification methods will be analyzed.

In order to analyze the performance of the algorithm, four multivariate data sets available in the UCI Machine Learning [1] were selected, which are often studied in machine learning. The choice of data sets was based on the range of diversity, considering the number of features, samples, classes, and nature of the data. The *Iris* dataset is a classic example from the literature, which has simple discrimination. *Wine* data has more features for analysis and has imbalanced classes. The *Glass* problem is composed of 6 classes and their sets have minority classes and data constituted as outliers. Finally, *Wave* sets have a large number of features and samples composed of noise. A summary of the datasets is shown in Table 1.

The algorithms were implemented in Matlab and executed using a machine with Intel Core i5-7300HQ, CPU 2.5 GHz and 8 Gb RAM. In all methods presented, the same datasets of training (80%) and validation (20%) were applied, employing the 5-fold cross-validation. The features of the data sets were normalized to zero mean and standard deviation equal to 1, being the only prepro-

[1] Repository of Iris, Wine, Glass and Wave datasets (UCI Machine Learning Repository): `https://archive.ics.uci.edu/ml`.

Table 2. Selected model terms and score values for each dataset.

Iris		Wine		Glass		Wave	
Terms	Score	Terms	Score	Terms	Score	Terms	Score
$u_3(k-1)$	0.9533	Constant	0.7759	Constant	0.7287	Constant	0.7690
Constant	0.7714	$u_7(k-1)$	0.7695	$u_4(k-1)$	0.5183	$u_7(k-1)$	0.5632
$u_4(k-1)u_4(k-1)$	0.6067	$u_{10}(k-1)u_{13}(k-1)$	0.6680	$u_3(k-1)u_7(k-1)$	0.5434	$u_{11}(k-1)$	0.5386
$u_3(k-1)u_3(k-1)$	0.4933	$u_{13}(k-1)$	0.6527	$u_3(k-1)u_3(k-1)$	0.4814	$u_8(k-1)u_{11}(k-1)$	0.4518
$u_2(k-1)u_4(k-1)$	0.4733	$u_7(k-1)u_{11}(k-1)$	0.5189	$u_4(k-1)u_6(k-1)$	0.4773	$u_{12}(k-1)u_{16}(k-1)$	0.4264
$u_1(k-1)u_4(k-1)$	0.4067	-	-	$u_3(k-1)u_4(k-1)$	0.4399	$u_{10}(k-1)$	0.4266
-	-	-	-	-	-	$u_{15}(k-1)u_{16}(k-1)$	0.4286
-	-	-	-	-	-	$u_{16}(k-1)$	0.4142
-	-	-	-	-	-	$u_5(k-1)u_{10}(k-1)$	0.4052
-	-	-	-	-	-	$u_{16}(k-1)$	0.4014

cessing performed. In the classification using the proposed method, the following parameters were considered: degree of nonlinearity $l = 2$, maximum lags $n_u = n_y = 2$ and maximum number of selected terms $k = 10$. For comparison purposes, the parameters inserted in the classification methods were chosen based on tests with the databases, selecting the configurations with greater accuracy using 5-fold cross-validation. The configuration used in each dataset is presented below:

- Iris - (RF) with division criterion Gini's diversity index and maximum number of decision divisions equal to 5, (SVM) with Polynomial kernel and order 2 and (KNN) with metric Minkowski distance using exponent equal to 3 and number of nearest neighbors equal to 10;
- Wine - (RF) with division criterion Gini's diversity index and maximum decision divisions equal to 20, (SVM) with Polynomial kernel and order 2 and (KNN) with metric Euclidean distance and number of nearest neighbors equal to 10;
- Glass - (RF) with division criterion Gini's diversity index and maximum decision divisions equal to 100, (SVM) with Polynomial kernel and order 3 and (KNN) with metric Euclidean distance and number of nearest neighbors equal to 10;
- Wave - (RF) with division criterion Gini's diversity index and maximum decision divisions equal to 20, (SVM) with Gaussian kernel and (KNN) with metric Euclidean distance and nearest neighbors equal to 10.

Figure 1 represents the application of the algorithm to the Iris dataset. The results suggest that the selection of 2 terms is sufficient to represent the classifier model, obtaining an average accuracy of 0.94. Furthermore, selecting more terms did not significantly increase the accuracy. On the other hand, in Figure 2 that represents the results applied to the Wine data, the selection of terms gradually increased the accuracy. Therefore, 5 terms were chosen to represent the classifier model, resulting in an average accuracy of 0.91. In Figure 3 it is possible to observe 2 local maximums in 4 and 10 terms, showing that in some cases the increase in the number of terms can result in an accuracy decrease. The simulation associated with the Wave data resulted in increased accuracy in the addition of new terms (see Figure 4), indicating that the maximum selection of $k = 10$ can be further increased for greater accuracy. In summary, the method was successful in selecting the features, ranking the most relevant in the classification process.

Table 2 represents the selected terms and their importance score for the classifier model using the proposed method. The values found explain the curves obtained in relation between accuracy ratio and the number of terms, since the growth rate is higher in the insertion of the most significant terms and degrades in the inclusion of the least significant terms (see Figure 3). The comparison between the average and maximum accuracy between the classification methods is shown in Table 3. The method obtained higher average accuracy in the tests of the sets of Iris and Wine. While the KNN and SVM methods achieved better results in the Glass and Wave sets respectively. However, the proposed method obtained better results compared to the RF and SVM methods in the Glass set, as well as surpassing the RF and KNN methods in the Wave sets. Another important aspect is that the method in the Wave sets used only 10 terms or features compared to the others that used 40. The purpose of the comparison is to reveal that the method competes with other classical algorithms, making it an alternative that can obtain gains in certain data sets.

Table 3. Comparison between average and maximum accuracy resulting from cross-validation.

		Iris	Wine	Glass	Wave
NARX	$\overline{x}$	0.9800	0.9382	0.6682	0.8148
	max	1.0000	1.0000	0.7907	0.8228
RF	$\overline{x}$	0.9467	0.8427	0.6449	0.7448
	max	0.9680	0.9189	0.7209	0.7538
SVM	$\overline{x}$	0.9533	0.9213	0.6682	0.8650
	max	1.0000	0.9730	0.7727	0.8749
KNN	$\overline{x}$	0.9533	0.8652	0.7150	0.7944
	max	1.0000	0.9412	0.7674	0.8208

Table 4. Performance of the proposed method in feature extraction.

Dataset	Features		Reduction	Accuracy	
	Data	Model	(%)	$\overline{x}$	max
Iris	4	2	50.00	0.9467	0.9667
Wine	13	5	61.53	0.9158	0.9444
Glass	9	4	55.55	0.6445	0.7381
Wave	40	10	75.00	0.8148	0.8228

The performance of the proposed method in feature extraction, together with the dimensionality reduction capability, is summarized in Table 4. The algorithm proposed in the tests significantly reduced the size of the data, resulting in average accuracy similar to those obtained by the other methods (see Table 3). In emphasis, the *Wine* test reduced the dimension by 61.53%, maintaining an average accuracy of 0.9158 and resulting in greater accuracy than those found in RF and KNN in Table 3. Likewise, in the *Wave* data there was a reduction of 75% and an average accuracy of 0.8148, consisting of better results compared to the RF and KNN techniques.

5. CONCLUSION

In this work, a hybrid technique was proposed that incorporates system identification and machine learning methodologies in the prediction of categorical variables. The presented algorithm performs the extraction and selection of features, ordering the most significant terms to compose a multiclass classification model. The model obtained is relatively simple and intuitive to interpret, providing insights with reduced dimensionality that clearly explain the incremental impact of a predictor variable on the response variable. The results show that the method excels in maximum accuracy by 3 of the applied case studies. In terms of average accuracy, the method obtained better results in 2 case studies. However, the proposed technique reduced the dimensionality of the data in the analyzed sets by more than 50%, keeping the accuracy equivalent to the other approached techniques. In general, the method is an interesting alternative for extraction and classification, and can achieve significant gains in certain datasets. The results are promising and in future proposals we want to evaluate the method in comparison with other extraction and feature selection techniques. Another approach is to use heuristic methods in the term selection process, to obtain optimal values for the number of terms in the model concerning the average accuracy.

ACKNOWLEDGEMENTS

We thank CAPES, CNPq, INERGE, FAPEMIG and Federal University of Juiz de Fora (UFJF) for the support.

REFERENCES

Abdi, H. and Williams, L.J. (2010). Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(4), 433–459.

Aguirre, L. (2007). *Introdução à identificação de sistemas - Técnicas lineares e não-lineares aplicadas a sistemas reais*. Editora UFMG.

Aguirre, L. and Jácome, C. (1998). Cluster analysis of NARMAX models for signal-dependent systems. *IEE Proceedings - Control Theory and Applications*, 145(4), 409–414.

Ayala Solares, J.R., Wei, H.L., and Billings, S.A. (2019). A novel logistic-NARX model as a classifier for dynamic binary classification. *Neural Computing and Applications*, 31(1), 11–25.

Billings, S.A. (2013). *Nonlinear system identification: NARMAX methods in the time, frequency, and spatio-temporal domains*. John Wiley & Sons.

Brunton, S.L. and Kutz, J.N. (2019). *Data-Driven Science and Engineering*. Cambridge University Press.

Cai, J., Luo, J., Wang, S., and Yang, S. (2018). Feature selection in machine learning: A new perspective. *Neurocomputing*, 300, 70–79.

Ferreira, D.D., Nepomuceno, E.G., Cerqueira, A.S., and Mendes, T.M. (2017). A model validation scale based on multiple indices. *Electrical Engineering*, 99(1), 325–334.

Jiménez, A.A., García Márquez, F.P., Moraleda, V.B., and Gómez Muñoz, C.Q. (2019). Linear and nonlinear features and machine learning for wind turbine blade ice detection and diagnosis. *Renewable Energy*, 132, 1034–1048.

Nagata, E.A., Ferreira, D.D., Bollen, M.H., Barbosa, B.H., Ribeiro, E.G., Duque, C.A., and Ribeiro, P.F. (2020). Real-time voltage sag detection and classification for power quality diagnostics. *Measurement*, 164.

Naik, C.A. and Kundu, P. (2014). Power quality disturbance classification employing S-transform and three-module artificial neural network. *International Transactions on Electrical Energy Systems*, 24(9), 1301–1322.

Nepomuceno, E.G. and Martins, S.A.M. (2016). A lower bound error for free-run simulation of the polynomial NARMAX. *Systems Science & Control Engineering*, 4(1), 50–58.

Pan, D., Zhao, Z., Zhang, L., and Tang, C. (2017). Recursive clustering K-nearest neighbors algorithm and the application in the classification of power quality disturbances. In *2017 IEEE Conference on Energy Internet and Energy System Integration (EI2)*, 1–5. IEEE.

Sette, S. and Boullart, L. (2001). Genetic programming: principles and applications. *Engineering Applications of Artificial Intelligence*, 14(6), 727–736.

Silva, P.H.O., Cerqueira, A.S., Nepomuceno, E.G., and Oliveira, A.F. (2020). Classificação de Distúrbios na Qualidade de Energia Usando Modelagem Logística-NARX Multinomial. In *Anais do Congresso Brasileiro de Automática 2020*. SBA.

Söderström, T. and Stoica, P. (1989). *System identification*. Prentice-Hall International.

Wei, H.L., Billings, S.A., and Liu, J. (2004). Term and variable selection for non-linear system identification. *International Journal of Control*, 77(1), 86–110.

Zhang, H., Liu, P., and Malik, O. (2003). Detection and classification of power quality disturbances in noisy conditions. *IEE Proceedings - Generation, Transmission and Distribution*, 150(5), 567.