IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received August 1, 2020, accepted August 14, 2020, date of publication August 27, 2020, date of current version September 9, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3019771

KPLS Optimization With Nature-Inspired Metaheuristic Algorithms

JORGE DANIEL MELLO-ROMÁN[1] **AND ADOLFO HERNÁNDEZ**[2]

[1]Faculty of Mathematical Science, Complutense University of Madrid, 28040 Madrid, Spain
[2]Financial and Actuarial Economics and Statistics Department, Complutense University of Madrid, 28223 Madrid, Spain

Corresponding author: Jorge Daniel Mello-Román (mellojorgepy@gmail.com)

ABSTRACT Kernel partial least squares regression (KPLS) is a technique used in several scientific areas because of its high predictive ability. This article proposes a methodology to simultaneously estimate both the parameters of the kernel function and the number of components of the KPLS regression to maximize its predictive ability. A metaheuristic optimization problem was proposed taking the cumulative cross-validation coefficient as an objective function to be maximized. It was solved using nature-inspired metaheuristic algorithms: the genetic algorithm, particle swarm optimization, grey wolf optimization and the firefly algorithm. To validate the results and have a reference measure of the efficiency of the nature-inspired metaheuristic algorithms, derivative-free optimization algorithms were also applied: Hooke-Jeeves and Nelder-Mead. The metaheuristic algorithms estimated optimal values of both of the kernel function parameters and the number of components in the KPLS regression.

INDEX TERMS Partial least squares regression, kernel-based method, cross-validation method, and nature-inspired metaheuristic algorithms.

I. INTRODUCTION

Partial least squares (PLS) regression is a linear method that seeks to predict a set of dependent variables (Y) from a set of predictors (X) by extracting orthogonal factors that maximize predictive ability, also called components [1], [2]. PLS is not appropriate for describing data structures when these exhibit nonlinear variations [3], so Rosipal and Trejo [4] proposed the kernel partial least squares regression method (KPLS), which transforms the original datasets into a space of arbitrary dimensionality characteristics, where the generation of a linear model is possible [5]. A recurring difficulty when implementing KPLS regression is determining both the number of components and kernel function parameters that maximize its predictive capacity [6], [7], generating an optimization problem.

In recent decades, so-called nature-inspired metaheuristic algorithms have emerged, which are powerful and effective in addressing various types of optimization problems, do not require assumptions to work well and are flexible and easy to implement [8]. In addition, according to Lin *et al.* [9], metaheuristic algorithms obtain better search subsets to determine optimal or approximate solutions in the transformed space, so they are suitable for determining both the kernel function parameters and number of components in the KPLS regression [10].

The aim of this work is to propose a methodology to optimize the predictive ability of KPLS regression through nature-inspired metaheuristic algorithms such as the genetic algorithm (GA), particle swarm optimization (PSO), grey wolf optimization (GWO) and the firefly algorithm (FFA). The relative performance of these algorithms is evaluated in terms of accuracy, convergence of estimates and computational speed. To validate the results and have a reference measure of the efficiency of the nature-inspired metaheuristic algorithms, the performance of two derivative-free algorithms is compared: Hooke-Jeeves (HJ) and Nelder-Mead (NM) [11]. This article is an extension of the conference proceeding presented for the International Workshop in Statistical Methods and Artificial Intelligence IWSMAI 20 [12], focused on the use of GA as an optimization agent. The scope of this article is broader in that it incorporates into the analysis a set of both nature-inspired metaheuristic algorithms and deterministic algorithms.

Several studies have chosen to use nature-inspired metaheuristic algorithms to improve the performance of PLS/KPLS regression, with different purposes and schemes than those presented in this article. In the literature review, it was found that the most recurrent metaheuristic algorithms in PLS/KPLS regression problems are GA [10], [13], [14],

The associate editor coordinating the review of this manuscript and approving it for publication was Zhenzhou Tang.

Algorithm 1 KPLS Algorithm by Rosipal and Trejo [4]

```
1. Randomly initialize $\mathbf{u}$ (any column of $\mathbf{Y}$)
2. $t = \mathbf{K}\, u$, $t \longleftarrow \frac{t}{||t||}$
3. $c = \mathbf{Y}^T\, t$
4. $u = \mathbf{Y}c$, $u \longleftarrow \frac{u}{||u||}$
5. Repeat steps 2 – 5 until convergence
6. Deflate $\mathbf{K}$,$\mathbf{Y}$ matrices: $\mathbf{K} \longleftarrow (\mathbf{I}\text{-}tt^T)\mathbf{K}(\mathbf{I}\text{-}tt^T)$; $\mathbf{Y} \longleftarrow$
   $(\mathbf{I}\text{-}tt^T\ \mathbf{Y})$ where $\mathbf{I}$ is an $n$-dimensional identity matrix
7. Repeat steps 2 – 7 until all the $h$-score vectors of $\mathbf{T}$
   and $\mathbf{U}$ are found.
```

PSO [15], [16], and less frequently FFA [17]. This article is organized as follows. Section 2 presents the KPLS regression method, its cross-validation coefficient Q^2_{cum} and describes the four nature-inspired metaheuristic algorithms to be implemented: GA, PSO, FFA and GWO. Section 3 presents the methodology. Sections 4 and 5 present the results and conclusions, respectively.

II. BACKGROUND

A. KERNEL PARTIAL LEAST SQUARES REGRESSION

Given an $(n \times N)$ matrix $\mathbf{X}$ of input variables $\{x_i\}_{(i=1)}^n \in \mathbb{R}^N$ and the $(n \times L)$ matrix $\mathbf{Y}$ of output variables $\{y_i\}_{(i=1)}^n \in \mathbb{R}^L$, assume a nonlinear transformation of the input variables $\{x_i\}_{(i=1)}^n$ into a feature space F, i.e., a mapping $\phi : x_i \in \mathbb{R}^N \longrightarrow \phi(x_i) \in F$. Denote by Φ an $(n \times M)$ matrix whose i-th row is the vector $\phi(x_i)$ in a M-dimensional feature space F, which can be very large [18]. The kernel function $k(x_i, x_j) = \phi(x_i)^T \phi(x_j)$ calculates the inner product in the feature space F. We can see that $\mathbf{K} = \Phi\Phi^T$ represents the $(n \times n)$ kernel matrix of the cross dot products between all mapped input data points $\{\phi(x_i)\}_{(i=1)}^n$, i.e., the element of the i-th row and j-th column of $\mathbf{K}$ is $k_{ij} = k(x_i, x_j)$ [19]. In this article, the Gaussian kernel function $k(x_i, x_j) = exp(-\sigma||x_i - x_j||^2)$ is used, where σ is a kernel parameter.

KPLS regression is essentially a technique based on the decomposition of this kernel matrix $\mathbf{K}$ into singular values [20]. Rosipal and Trejo [4] proposed the KPLS algorithm as shown in Algorithm 1.

After all the h components are extracted, we can write the matrix of the regression coefficients $\mathbf{B}$ in the following form $\mathbf{B} = \Phi^T\ \mathbf{U}(\mathbf{T}^T\mathbf{K}\mathbf{U})^{(-1)}\mathbf{T}^T\ \mathbf{Y}$, where the $(n \times h)$ matrices $\mathbf{T}$ and $\mathbf{U}$ represent the input and output scores, respectively.

To make predictions on the training data, we can write $\hat{\boldsymbol{Y}} = \mathbf{K}\ \mathbf{U}\ (\mathbf{T}^T\ \mathbf{K}\ \mathbf{U})^{(-1)}\ \mathbf{T}^T\ \mathbf{Y} = \mathbf{T}^T\ \mathbf{Y}$. For predictions made on testing points, i.e., $\hat{\boldsymbol{Y}}_{\boldsymbol{t}} = \mathbf{K}_t\ \mathbf{U}\ (\mathbf{T}^T\mathbf{K}\mathbf{U})^{(-1)}\ \mathbf{T}^T\mathbf{Y}$, where $\mathbf{K}_t$ is the $(n_t \times n)$ test matrix whose elements are $k_{ij} = k(x_i, x_j)$, where $\{x_i\}_{(i=n+1)}^{(n+n_t)}$ and $\{x_j\}_{(i=1)}^n$ are the testing and training points, respectively. Before applying KPLS, mean centering of the data should be carried out in the feature space [3].

KPLS regression predictive ability can be estimated for each included h-component by means of the cross-validation coefficient $Q_h^2 = 1 - \frac{PRESS_h}{RSS_{h-1}}$, where $PRESS_h$ denotes the predictive residual sum of squares including the component h and RSS_{h-1} the residual sum of squares with the previous components [21], [22]. The overall predictive performance of KPLS regression can be assessed by the cumulative cross-validation coefficient $Q^2_{cum} = 1 - \prod_1^h(1 - Q_h^2)$, which takes values between 0 and 1, and the higher the Q^2_{cum} value is, the better the predictive performance [23].

Cross-validation is probably one of the oldest resampling techniques [24]. The procedure divides the dataset into blocks of size equal to k and then use k - 1 blocks to fit the model and validate it in the remaining block. This is done for all possible combinations of k - 1 of the k blocks. The k-blocks are usually called folds in the cross-validation literature [25]. In this article, the cross-validation procedure was applied to randomly split the dataset into $k = 10$ segments of equal size to obtain the coefficients Q^2 and Q^2_{cum} [26].

We can list two main problems in KPLS: the selection of the kernel function and its parameters and the selection of the number of components. However, kernel function selection is still an open problem in KPLS [27]. This article proposes a metaheuristic tuning procedure to simultaneously estimate the kernel function parameter and the number of components, where the objective function is the cumulative cross-validation coefficient $Q^2_{cum}(h, \theta)$ for the case of KPLS regression is a function of both parameters. The optimization task can be written as follows.

$$(h, \theta) = \arg\max_{h,\theta} Q^2_{cum} \text{ where } h \in \mathbb{Z}^+ \leqslant N \wedge \theta \in S \subset \mathbb{R}$$

The domain of the number of components is the set of positive integers smaller than the number of columns of $\mathbf{X}$ and the domain of the kernel function parameter is a predefined subset $\mathbf{S}$ of real numbers.

The main differences with respect to [7] are both the optimization task formulation and the cross-validation approach based on grid-search that it proposes.This article does not include a comparison of the accuracy of the estimates and the computational costs associated with the two proposals due to differences in the databases evaluated and the software packages used.

An additional feature of this proposal is that it designs an algorithm already embedded in a general iterative optimizer, e.g., a nature-inspired metaheuristic or derivative-free algorithms. The proposed method is universal for any kernel function and easy to implement in any computer. Algorithm 2 describes the steps of the method to be used with nature-inspired metaheuristic algorithms.

To perform the statistical analysis of the estimates, several runs of this procedure are performed for each optimizer, and the convergence of the estimates with the gradual increase in the optimizer iterations is also evaluated.

B. NATURE-INSPIRED METAHEURISTICS ALGORITHMS

Metaheuristics, in their original definition, are solution methods that orchestrate an interaction between local improvement procedures and higher-level strategies to create

Algorithm 2 KPLS Parameter Selection

```
Given $\mathbf{X}_{n\times N}$ the input matrix, $\mathbf{Y}_{n\times L}$ the output matrix,
a kernel function $k_{ij}$ and its kernel parameter $\theta \in \mathbf{S}$,
and the number of components $h \in \mathbb{Z}^{+}$.
1. The optimizer randomly generates an initial
population $p$ of $\theta$ (individuals).
2. From $\mathbf{X}_{n\times N}$ the input matrix, $p$ kernel matrices
$\mathbf{K}_{n\times n}$ are generated, one matrix for each $\theta$.
3. KPLS regression (Algorithm 1) between $\mathbf{K}_{n\times n}$ and
$\mathbf{Y}_{n\times L}$ is executed with $h = N$ components.
4. $Q^2_{cum}$ is calculated by 10-fold cross-validation for
each KPLS.
5. Identify $\theta$ associated with the maximum $Q^2_{cum}$ and
extract $h$ for which this value is reached.
6. $g$ optimizer iterations are performed. The fitness
function of the optimizer is max $Q^2_{cum}$.
7. The optimizer determines the best values of $\theta$ and $h$
and the optimal $Q^2_{cum}$.
```

a process capable of escaping from local optima and performing a robust search of a solution space [28]. Metaheuristic algorithms are appropriate for solving problems with complex formulation because they do not use any specific information of the problem in the exploration of the space of feasible solutions [29]. An important set of nature-inspired metaheuristic algorithms is that they mimic the behavioral characteristics of biological agents [30]. The most popular are evolutionary algorithms and swarm intelligence algorithms, which have demonstrated their potential for solving important engineering decision-making problems [31].

The basic formulations of four nature-inspired metaheuristic algorithms: GA, PSO, FFA, and GWO [32]–[34], are described in this section. The derivative-free optimization algorithms used in this work are also briefly presented [11].

C. GENETIC ALGORITHMS

GA is a population-based metaheuristic algorithm widely used in search and optimization problems [35], [36]. GA works on a population of individuals (chromosomes). Each individual represents a potential solution to the problem to be solved [37]. In the algorithm, an individual (or chromosome) is represented by a binary bit string, an initial population of individuals is created randomly, the objective function for each individual is evaluated, and based on these values, the individuals of the next iteration are produced by selection, crossing and mutation operations. The basic steps of the algorithm can be consulted in [14].

For the implementation of a GA, it is necessary to previously define its parameters such as crossing probability p_c, mutation probability p_m [38], population size p and number of generations g [32]. The completion criterion can be determined by the number of generations (iterations) g or the convergence of results [39]. In Mello-Román *et al.* [12], the optimization of the KPLS with GA was deepened, and preliminary tests were carried out to determine the p and g values.

D. PARTICLE SWARM OPTIMIZATION

PSO is a population-based metaheuristic optimization technique inspired by the social behavior of bird flocks or the movement of fish shoals [40]. Each particle in the swarm is a potential solution to the optimization problem, which moves through the search space and updates its velocity and position with a learning mechanism based on its personal best experience and the population best experience [15]. In each iteration, every particle moves in a direction that is based on the previous best individual position *pbest* and the best overall position of the swarm *gbest*. For the dimension d of a particle i, the velocity and position can be updated according to the following equations:

$$v_{id}^{(t+1)} = \omega \cdot v_{id}^{t} + c_1 \cdot r_1(pbest_{id}^{t} - x_{id}^{t}) + c_2 \cdot r_2(gbest_{id}^{t} - x_{id}^{t}), \tag{1}$$

$$x_{id}^{(t+1)} = v_{id}^{(t+1)} + x_{id}^{t}, \tag{2}$$

where t is the iteration, v_{id} is the velocity, x_{id} is the particle position, ω is an inertia factor, c_1 and c_2 are learning factors, and r_1 and r_2 are random numbers between 0 and 1. More details about the steps followed by PSO can be found in [16].

E. FIREFLY ALGORITHM

FFA is a swarm intelligence technique inspired by imitating the light emissions of fireflies to attract each other [41]. To formulate the algorithm, the following conditions are established: a firefly can be attracted by any other firefly, the attraction is proportional to brightness for any pair of fireflies, the less bright firefly will move towards the brighter firefly, as the fireflies move away the perception of the light decreases, and the intensity of the light emitted by the firefly is determined by the value of the function to be optimized [42].

Each firefly (candidate solution) flashes its lights with some brightness (associated with the objective function) attracting other fireflies within its neighborhood. This attractiveness depends on the distance between the two fireflies and is determined by $\beta(r) = \beta_0 \, e^{(-r_{ij}^2)}$, where r or r_{ij} is the (Euclidean) distance between the i-th and j-th of two fireflies, β_0 is the initial attractiveness at $r = 0$ and γ is a fixed light absorption coefficient that controls the decrease in the light intensity [30]. The movement of the i-th firefly is attracted to another more attractive firefly j and is given by the following equation [42]:

$$x_i = x_i + \beta_0 e^{-\gamma r_{ij}^2}(x_j - x_i) + \alpha \epsilon_i, \tag{3}$$

where x_i is the particular location of the i-th firefly, α is the randomization parameter, and ϵ_i is a vector of random numbers drawn from a Gaussian or uniform distribution. The steps of the FFA algorithm can be found in [43].

F. GREY WOLF ALGORITHM

GWO is a recent metaheuristic optimization technique that simulates the hierarchical mechanism and predatory behavior of the gray wolf pack, where under the leadership of the main gray wolf, wolves capture prey through a series of processes: surrounding, hunting and attacking [44]. GWO is an optimization algorithm with a strong global search capability; however, studies continue to be conducted to verify its performance in different types of problems and suggest improvements [45], [46].

According to Mirjalili *et al.* [44], the process can be described as follows: first, the gray wolves track the prey, chase it and approach it; if the prey runs, then they chase it, surround it and harass it until it stops moving. Finally, the attack begins. Details of the steps followed by the GWO algorithm and its parameters can be found in Gao and Chao *et al.* [45].

G. DERIVATIVE-FREE OPTIMIZATION ALGORITHMS

The proposal of this work is to optimize the KPLS regression by means of nature-inspired metaheuristic algorithms. However, we intend to check the performance of some deterministic algorithms on the same optimization problem, both to validate the results and to have a reference measure of the efficiency of the set of nature-inspired metaheuristic algorithms evaluated. According to Sun *et al.* [47], when it is necessary to determine the optimum of a real-value function defined in n-dimensional space, but the derivatives are not available because they are computationally expensive or there is no explicit expression of the partial derivatives of the function, it is possible to resort to optimization methods called derivative-free or direct search methods. These methods use only function values and apply when no computer code can be produced for the derivative of the function. In general, they are robust methods, even for nonconvex or discontinuous functions [48].

Two popular derivative-free optimization algorithms are NM and HJ [11]. The NM algorithm is a direct search method for the unrestricted optimization of multidimensional functions, and it attempts to minimize a nonlinear scalar function of n variables by using only values of the function [49]. The HJ method is also a derivative-free optimization method that searches for directions of descent by exploratory movements through coordinate directions by performing two types of searches: an exploratory search and a pattern search, which are repeated until convergence [50].

III. METHODOLOGY

Taking as reference the good practices proposed by Osaba *et al.* [29] for the implementation and comparison of metaheuristics, in this work, we present for each algorithm the number of runs performed, the mean and standard deviation of both the objective function Q^2_{cum} and the KPLS parameters θ and h [7], the convergence behavior, the run times, and an appropriate statistical analysis with the results obtained.

The parameter estimation algorithm for the KPLS regression was programmed in the R programming language. This required the installation of the following packages: plsdepot by Sanchez [51], kernlab by Karatzoglou *et al.* [52], metaheuristicOpt by Riza and Nugroho [53] and dfoptim [54]. The Gaussian kernel function was selected for this work [7], [26].

Preliminary tests were performed to select the hyperparameters of the nature-inspired metaheuristic algorithms [32], [33]. Several sets of hyperparameters were evaluated while keeping both the population size and number of iterations constant. Considering the overall goal of this work, the sets of hyperparameters for which the algorithms showed higher mean values of Q^2_{cum} and with lower dispersion were selected [55].

The iterations of the metaheuristic algorithms must be stopped according to some criterion, which can be the convergence of the algorithm or a preset number of iterations [39]. In this article, we evaluated the convergence of estimates for a set of iterations $g = \{10, 30, 50, 100, 150\}$ keeping the population size constant at the value $p = 40$ [53].

The parameter estimation procedure for KPLS regression is presented in Algorithm 2. Thirty runs [43] of the algorithm were performed, extracting the maximum value of Q^2_{cum} and its associated values of both the number of components h and the parameter of the Gaussian kernel function σ for each run. Furthermore, the running time was calculated in minutes.

For the different runs, the maximum Q^2_{cum} reached, the Gaussian kernel function parameter value σ, the number of components h and the running time were obtained. The number of iterations of each metaheuristic algorithm was gradually increased in $g = \{10, 30, 50, 100, 150\}$, and the mean and standard deviation of these estimates were extracted [29]. To validate the results and have a benchmark of the efficiency of the GA, PSO, FFA and GWO algorithms in the optimization task, the performance of the HJ and NM algorithms was also evaluated. The NM and HJ algorithms establish a set of random starting points, taking into account the fact that these procedures are usually stuck in local optima [56]. The parameters of these algorithms were established with reference to recent research [57], [58].

Finally, a statistical analysis was carried out with the results obtained to test the hypothesis that there are no statistically significant differences between the estimates of the different metaheuristic algorithms [29]. One-way ANOVA was performed taking as a null hypothesis that the means of the estimates of the objective function Q^2_{cum} and the σ and h parameters were the same for the different algorithms evaluated. The analysis was complemented with variance homogeneity tests and multiple post hoc comparisons to identify the best-performing algorithms [67]. The Thamane T2 test was used for post hoc comparisons. This test is indicated when variances and/or sample sizes are unequal [68]. Additionally, the Kolmogorov-Smirnov normality test was applied to contrast that the distribution of σ estimates is normally distributed.

A. DATASET

For the implementation of the Algorithm 2, a dataset reported by Tsanas *et al.* [60] in the Machine Learning Repository at the University of Carolina at Irvine [61] was used. The dataset consists of biomedical voice measurements from 42 people with early Parkinson's disease recruited for remote monitoring of symptom progression. The predictor variables are subject number, subject age, subject gender, time interval from initial recruitment date, and 16 biomedical voice measurements. The data are used in predictive tasks for scores on the Unified Parkinson's Disease Rating Scale (UPDRS), motor and total scores [60], [62]. This scale is used to track Parkinson's disease [63].

Several research studies have used different techniques to attempt to predict the severity of Parkinson's disease symptoms from speech cues [64]. In particular, on the same dataset, feature selection methods [60], least squares regression techniques, nonparametric classification and regression trees were applied. In [63], artificial neural networks were tested. In both works, [63] and [64], despite the good results, the high correlation between some measures of dysphonia was highlighted, and it was suggested to further explore nonlinear methods.

Additionally, we used another dataset [65] hosted in the same repository [61] to implement Algorithm 2. It contains data on the academic performance of three hundred ninety-five (395) students of two Portuguese secondary schools in mathematics and Portuguese language, measuring forty-six (46) variables: grades of three periods, G1, G2 and G3, demographic, social and other variables related to the school, all compiled through reports and school questionnaires. More details about the variables can be found in [61]. This data file has been used in the past [65] for classification and ordinal regression purposes. More specifically, univariate ordinal regression was performed taking G3 as the response variable, comparing the results when G2 and G1 were also included as predictor variables and when they were excluded. It has been noted that feature selection methods should also be explored since only a small portion of the input variables considered appear to be relevant.

IV. RESULTS

Preliminary tests were first carried out to select the hyperparameters of the nature-inspired metaheuristic algorithms for the two datasets used [60], [65]. The results are presented in Table 1.

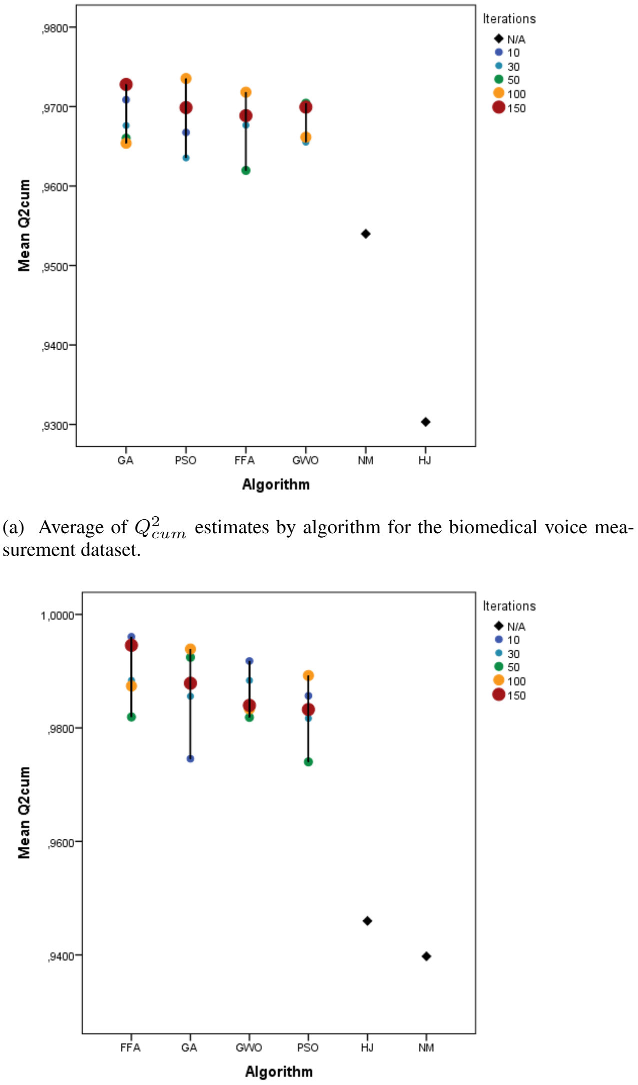
Below are the results of thirty runs of the parameter estimation algorithm in KPLS regression proposed in Algorithm 2 for the metaheuristic algorithms GA, FFA, PSO and GWO and the derivative-free algorithms NM and HJ. To generate the kernel matrix **K**, the Gaussian kernel function was used with the parameter $\sigma > 0$.

The population size of each metaheuristic algorithm was set at $p = 40$, and the iterations were gradually increased by $g = \{10, 30, 50, 100, 150\}$. The derivative-free NM and HJ algorithms took several random starting points $\sigma_0 > 0$ in given ranges for both datasets. The convergence criterion was the value 1.e-06 [54].

Figure 1 presents the behavior of the estimates of Q^2_{cum} for both datasets. The metaheuristic algorithms estimated higher average values of Q^2_{cum} and with a lower dispersion than the free derivative algorithms. Estimates of Q^2_{cum} were very similar for all four metaheuristic algorithms, with minimal

TABLE 1. Hyperparameters of nature-inspired metaheuristic algorithms by dataset.

Algorithms	Hyperparameters	Biomedical voice measurements dataset [60]	Academic performance dataset [66].
PSO	V_{max} maximum speed	2	4
	C_1 individual learning	2	2
	C_2 collective learning	2	2
	ω inertia factor	0.729	0.729
FFA	β_0 attraction coefficient	0.6	1
	γ absorption coefficient	1	1
	α randomization parameter	0.5	0.5
GA	P_m mutation probability	0.5	0.5
	P_c crossing probability	0.8	0.5
GWO	lb lower limit	0	0
	ub upper limit	100	10



(a) Average of Q^2_{cum} estimates by algorithm for the biomedical voice measurement dataset.

(b) Average of Q^2_{cum} estimates by algorithm for the academic performance dataset.

FIGURE 1. Average of Q^2_{cum} estimates by algorithm.

TABLE 2. Mean and standard deviation of estimates of Q^2_{cum}, σ, h and running time for the biomedical voice measurement dataset.

		Algorithm											
		GA		PSO		FFA		GWO		NM		HJ	
	g	Mean	St. Dev.	Mean	St. Dev.	Mean	St. Dev.	Mean	St. Dev.	Mean	St. Dev.	Mean	St. Dev.
Q^2_{cum}	10	0.971	0.016	0.967	0.02	0.969	0.016	0.97	0.015	-	-	-	-
	30	0.968	0.017	0.964	0.019	0.968	0.017	0.966	0.019	-	-	-	-
	50	0.966	0.017	0.97	0.015	0.962	0.02	0.97	0.013	-	-	-	-
	100	0.965	0.018	0.974	0.01	0.972	0.013	0.966	0.018	-	-	-	-
	150	0.973	0.012	0.97	0.017	0.969	0.015	0.97	0.013	-	-	-	-
	N/A	-	-	-	-	-	-	-	-	0.954	0.045	0.9303	0.0486
σ	10	49.17	3.32	47.43	3.49	48.67	2.8	48.13	3.33	-	-	-	-
	30	49	3.31	47.35	2.99	47.42	3.3	47.04	3.15	-	-	-	-
	50	48.83	3.27	47.06	2.3	47.1	3.23	47.2	2.73	-	-	-	-
	100	48.58	2.74	47.51	2.95	48.14	2.68	47.18	3.12	-	-	-	-
	150	48.44	2.57	47.09	2.49	48.68	3.12	45.27	3.01	-	-	-	-
	N/A	-	-	-	-	-	-	-	-	40.86	14.2	51.53	27.17
h	10	6	0	6	0	6	0	6	0	-	-	-	-
	30	6	0	6	0	6	0	6	0	-	-	-	-
	50	6	0	6	0	6	0	6	0	-	-	-	-
	100	6	0	6	0	6	0	6	0	-	-	-	-
	150	6	0	6	0	6	0	6	0	-	-	-	-
	N/A	-	-	-	-	-	-	-	-	6	0	6	0
Time (min)	10	124.3	4.6	119.7	3.1	70.6	2.9	71.1	3.8	-	-	-	-
	30	352.9	22.5	513.6	92.8	201.5	14	208.1	4	-	-	-	-
	50	571.2	55.8	800.6	117.2	332.9	27.7	334.8	19.6	-	-	-	-
	100	1135.4	80.7	1141.8	474.4	616.6	59.5	664.9	32.5	-	-	-	-
	150	1509.7	347.1	1734.7	727.1	940.4	83.7	852	156.4	-	-	-	-
	N/A	-	-	-	-	-	-	-	-	5.1	0.9	18	2.6

TABLE 3. Mean and standard deviation of estimates of Q^2_{cum}, σ, h and running time for the academic performance dataset.

		Algorithm											
		GA		PSO		FFA		GWO		NM		HJ	
	g	Mean	St. Dev.	Mean	St. Dev.	Mean	St. Dev.	Mean	St. Dev.	Mean	St. Dev.	Mean	St. Dev.
Q^2_{cum}	10	0.975	0.056	0.986	0.045	0.996	0.003	0.992	0.032	-	-	-	-
	30	0.986	0.041	0.982	0.047	0.988	0.035	0.988	0.036	-	-	-	-
	50	0.992	0.026	0.974	0.061	0.982	0.047	0.982	0.049	-	-	-	-
	100	0.994	0.019	0.989	0.031	0.987	0.035	0.983	0.05	-	-	-	-
	150	0.988	0.034	0.983	0.045	0.995	0.003	0.984	0.041	-	-	-	-
	N/A	-	-	-	-	-	-	-	-	0.94	0.22	0.946	0.21
σ	10	3.9	0.65	4.07	0.21	3.86	0.71	4.11	0.13	-	-	-	-
	30	4.08	0.37	4.16	0.13	4.05	0.55	4.08	0.16	-	-	-	-
	50	4.04	0.39	4.07	0.16	3.98	0.75	4.11	0.12	-	-	-	-
	100	3.87	0.66	4.06	0.14	3.73	0.82	4.08	0.17	-	-	-	-
	150	3.7	0.82	4.11	0.13	4.02	0.58	4.09	0.13	-	-	-	-
	N/A	-	-	-	-	-	-	-	-	7.11	4.96	6.01	2.89
h	10	7.4	0.7	7.1	0.4	7.2	0.8	7.1	0.3	-	-	-	-
	30	7.2	0.6	7	0.3	7.1	0.8	7.1	0.4	-	-	-	-
	50	7.2	0.5	7.2	0.4	7.1	1.1	7	0.2	-	-	-	-
	100	7.4	0.7	7.1	0.4	7.4	1	7.1	0.4	-	-	-	-
	150	7.5	0.9	7.1	0.3	7.1	0.8	7.1	0.3	-	-	-	-
	N/A	-	-	-	-	-	-	-	-	6.5	1.4	6.4	1.3
Time (min)	10	56.9	4.1	109.5	6.1	32	1.8	38.3	2.7	-	-	-	-
	30	155.8	8.5	274.2	5.5	90.9	3.5	99.5	8.1	-	-	-	-
	50	261.2	12.6	483	21.3	145.2	8.3	146.3	21.3	-	-	-	-
	100	467.2	60.7	822.8	106.7	274.6	20.7	236.1	26.6	-	-	-	-
	150	576.1	123.3	978.7	395.7	368.1	79.5	304.6	93.1	-	-	-	-
	N/A	-	-	-	-	-	-	-	-	1.8	0.4	7.4	0.9

variations observed with the increasing numbers of iterations. Despite the above differences, the estimates of the deterministic algorithms do not differ considerably from the values obtained by the metaheuristic algorithms. Details can be seen in Tables 2 and 3.

A statistical analysis of Q^2_{cum} estimates was carried out [29]. First, one-way ANOVA was executed taking as a null hypothesis that the means of Q^2_{cum} were equal for the different algorithms. The analysis was complemented by a variance homogeneity test and multiple post hoc comparisons to identify the best-performing algorithms [67]. The results presented in Table 4 indicate that there is statistical evidence to reject the null hypothesis that the average estimates of Q^2_{cum} are the same for the different tested algorithms for both datasets. Similarly, the result given by Levene's statistic indicates the nonhomogeneity of variances.

Estimates of the parameter of the Gaussian kernel function σ were on average similar for the algorithms evaluated in

TABLE 4. ANOVA and test of homogeneity of variances – Q^2_{cum} and σ by algorithms.

	Dataset	One -Way ANOVA		Homogeneity of variances	
		F	Sig.	Levene statistic	Sig.
Q^2_{cum}	Biomedical voice measurement	25.453	0	25.788	0
	Academic performance	3.247	0.007	14.079	0
σ	Biomedical voice measurement	6.329	0	169.87	0
	Academic performance	2.793	0.017	21.905	0

TABLE 5. T2 Tamhane's post hoc comparisons – Q^2_{cum} by algorithm.

Algorithms		Biomedical voice measurement dataset			Academic performance dataset		
		Mean difference*	St. Error	Sig.	Mean difference*	St. Error	Sig.
GA	PSO	0	0.002	1	0.0042	0.005	0.999
GA	FFA	0.0007	0.0019	1	0.0026	0.0041	1
GA	GWO	0.0001	0.0019	1	0.001	0.0046	1
GA	NM	0.0145	0.0095	0.894	0.0472	0.0551	1
GA	HJ	0.0382	0.0077	0	0.041	0.0484	1
PSO	FFA	0.0007	0.002	1	0.0069	0.0047	0.901
PSO	GWO	0.0001	0.0019	1	0.0032	0.0052	1
PSO	NM	0.0145	0.0095	0.894	0.043	0.0551	1
PSO	HJ	0.0382	0.0077	0	0.0367	0.0484	1
FFA	GWO	0.0006	0.0019	1	0.0036	0.0043	1
FFA	NM	0.0138	0.0095	0.925	0.0498	0.055	0.999
FFA	HJ	0.0375	0.0077	0	0.0436	0.0483	0.999
GWO	NM	0.0144	0.0095	0.898	0.0462	0.0551	1
GWO	HJ	0.0381	0.0077	0	0.0399	0.0484	1
NM	HJ	0.0237	0.0121	0.574	0.0062	0.0732	1

* Mean difference in absolute value

TABLE 6. Estimates of *h* for the academic performance dataset by algorithm.

h	Algorithm					
	FFA	GA	GWO	PSO	HJ	NM
5	0.70%	0.00%	0.00%	0.00%	42.10%	37.50%
6	18.10%	2.10%	0.70%	0.70%	0.00%	0.00%
7	54.30%	73.40%	89.70%	86.80%	47.40%	50.00%
8	16.70%	14.00%	9.60%	12.50%	0.00%	0.00%
9	10.10%	10.50%	0.00%	0.00%	10.50%	12.50%

TABLE 7. Tests of normality - Kolmogorov-Smirnov of σ parameter.

Biomedical voice measurement when h =6			Academic performance when h = 7		
Algorithm	Statistic	Sig.	Algorithm	Statistic	Sig.
FFA	0,074	0,048	FFA	0,13	0,003
GA	0,044	0,2	GA	0,1	0,011
GWO	0,063	0,2	GWO	0,05	0,2
PSO	0,037	0,2	PSO	0,053	0,2
HJ	0,092	0,2	HJ	0,275	0,048
NM	0,274	0	NM	0,369	0,002

both datasets. Details can be seen in Tables 2 and 3. NM and HJ showed mean values of σ close to those obtained by the metaheuristic algorithms, but with a much larger standard deviation. Table 4 shows the results of the ANOVA carried out to contrast the null hypothesis of equality of means of σ obtained by the different algorithms implemented. The ANOVA test indicates that there is statistical evidence to reject the equal-means hypothesis at a significance level of $\alpha = 0.05$. Likewise, Levene's statistic indicates the nonhomogeneity of variances.

For post hoc comparisons, the T2 Thamane test was used to indicate when variances and/or sample sizes are unequal [68]. The results for the voice biomedical dataset indicate a significant difference between the Q^2_{cum} estimates of the HJ algorithm and the other algorithms. In the academic performance dataset, multiple comparisons found no statistical evidence to state that Q^2_{cum} estimates were different for the algorithms evaluated. In this dataset, pairwise comparisons were not significant, while the overall effect was weakly significant [69]. This last claim could be justified by both the weak overall effect and the conservative characteristics of the T2 Thamane test. The results are presented in Table 5.

The number of h components estimated was equal to 6 for all algorithms evaluated in the biomedical voice measurement dataset. For the academic performance dataset, the estimates were variable; however, the most frequent estimate for all the algorithms was $h = 7$. The results are presented in Table 6.

A result that is favorable to the estimation in the interval of the σ parameter is the approximately normal distribution of the estimates for certain algorithms. Taking $h = 6$ for the biomedical voice measurement dataset and $h = 7$ for the academic performance dataset, the Kolmogorov - Smirnov normality test was applied. The null hypothesis of the contrast was that the distribution of sigma estimates is normally distributed. At a significance level $\alpha = 0.05$, the hypothesis for the algorithms for PSO and GWO in both datasets was not rejected. Details are shown in Table 7. The distributions of the distribution of σ parameter estimates by the nature-inspired metaheuristics algorithm are presented in Figure 2 for both datasets.

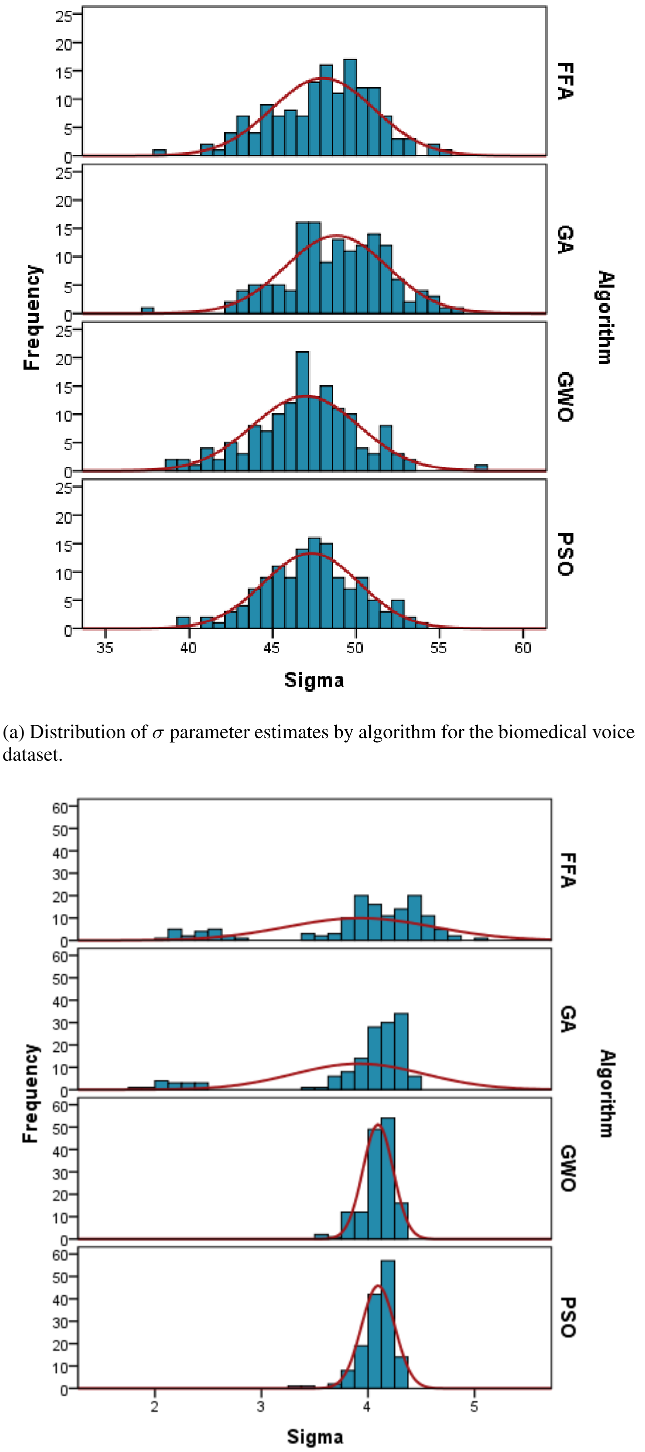
According to [8], in a complex optimization problem where there is no theory to find or verify the optimal, if several nature-inspired metaheuristic algorithms with tuned hyperparameters converge to a solution, this solution is most likely the optimal solution.

A. CONVERGENCE ANALYSIS

The convergence of estimates of Q^2_{cum}, σ and h was evaluated with respect to the variation in the number of iterations of each nature-inspired metaheuristic algorithm. Tables 2 and 3 do not show an overall effect in the increase in the iterations in the means and standard deviations obtained for Q^2_{cum}, σ and h. A statistical test was carried out taking as a null hypothesis the equality of means of Q^2_{cum}, σ and h for the different levels of iterations. Levene's statistic was also calculated to verify the homogeneity of variances. Table 8 presents the results.

There is no statistical evidence of the effect of increasing iterations in nature-inspired metaheuristic algorithms on estimates of Q^2_{cum}, σ and h in the two datasets evaluated at an α significance level $= 0.05$. However, for the academic performance dataset and at a significance level of $\alpha = 0.10$, the null hypothesis of equality of means and variances in σ estimates with the increase in the number of iterations is rejected.

A relevant point in the comparison of the optimization algorithms is the computational cost associated with each algorithm. Tables 2 and 3 show a direct relationship between the number of iterations and the running time in minutes for the four metaheuristic algorithms. GA and PSO have a higher associated computational cost in both datasets and for any



(a) Distribution of σ parameter estimates by algorithm for the biomedical voice dataset.

(b) Distribution of σ parameter estimates by algorithm for the academic performance dataset.

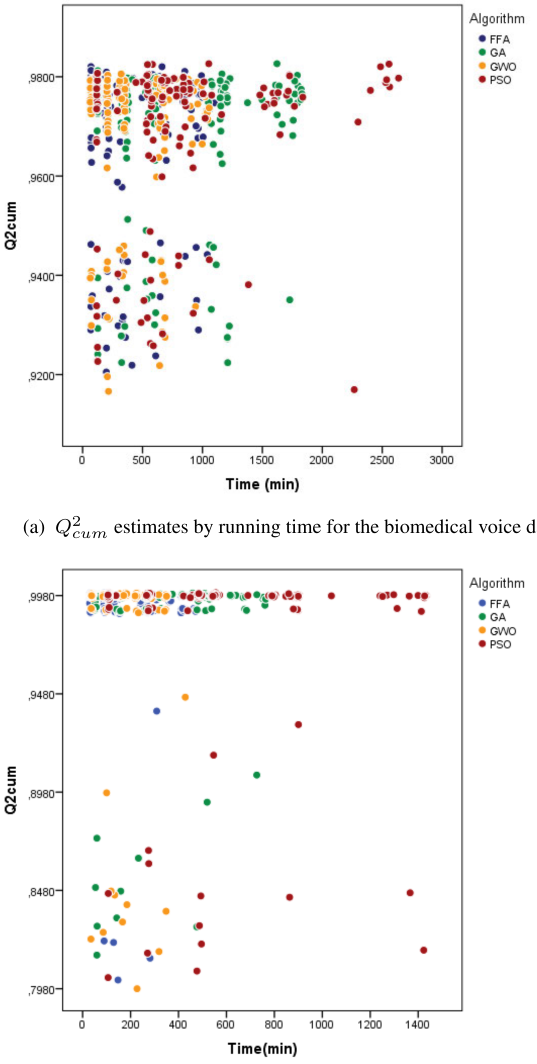
FIGURE 2. Distribution of σ parameter estimates by algorithm.

number of iterations. However, the reduced running time of the NM and HJ algorithms is noteworthy.

TABLE 8. ANOVA and test of homogeneity of variances – Q^2_{cum} and σ by iterations.

	Dataset	One -Way ANOVA		Homogeneity of variances	
		F	Sig.	Levene statistic	Sig.
Q^2_{cum}	Biomedical voice measurement	1.335	0.256	3.839	0.004
	Academic performance	0.405	0.805	1.842	0.119
σ	Biomedical voice measurement	1.659	0.158	0.844	0.497
	Academic performance	2.037	0.088	2.389	0.05
h	Biomedical voice measurement*	-	-	-	-
	Academic performance	1.32	0.261	3.1	0.015

*h is a constant value



(a) Q^2_{cum} estimates by running time for the biomedical voice dataset.

(b) Q^2_{cum} estimates by running time for the academic performance dataset.

FIGURE 3. Q^2_{cum} estimates by running time for datasets.

For the evaluated datasets and this specific optimization task, the results are inconclusive regarding the contribution to the accuracy of the estimates made by the increased iterations and running time of the nature-inspired metaheuristic algorithms. Figure 3 shows the estimates of Q^2_{cum} in both datasets with respect to the running time of the nature-inspired metaheuristic algorithms. No direct relationship between the two variables is observed in either cases.

These results are consistent with those of previous publications [8], which suggest that deterministic algorithms are more computationally efficient than nature-inspired metaheuristic algorithms and refer to the slow convergence of metaheuristic algorithms.

Due to differences in objectives and methodologies, it was not possible to compare the results obtained with the biomedical voice measurement dataset with other studies conducted on the same dataset [60], [63]. The results obtained in this article regarding the academic performance dataset first verified the estimates obtained in [12], which exclusively used GA as the optimizer to improve the predictive capacity of the KPLS regression. However, the results in terms of convergence of estimates with the increasing iterations have not been repeated for the voice biomedical measurement dataset or for the other nature-inspired metaheuristic algorithms.

V. CONCLUSION

This article proposes a methodology to simultaneously estimate both the kernel function parameter θ and the number of components h, which maximize the predictive capacity in the KPLS regression. An optimization problem is posed that takes as an objective function the cumulative cross-validation coefficient Q^2_{cum}. To solve the problem, an algorithm is designed that embeds nature-inspired metaheuristic algorithms: GA, PSO, FFA and GWO. To validate the results and have a reference measure of the efficiency of the set of algorithms, the performance of derivative-free optimization algorithms NM and HJ on the indicated problem is also evaluated.

The results indicate that the nature-inspired metaheuristic algorithms allow the estimation of approximate optimal values of the KPLS regression parameters. In the two datasets, the mean estimates of Q^2_{cum}, θ and h made by the four metaheuristic algorithms were similar with low dispersion. The derivative-free algorithms obtained a similar performance in the mean of the estimates, validating the results obtained, but the dispersion in the estimates was generally higher. Under the conditions established in this work, it is not possible to draw a conclusion about the convergence of the estimates with the increase in the iterations in the metaheuristic algorithms. Additionally, the derivative-free algorithms have shown a much lower computational time than the metaheuristic algorithms.

The conclusions reached are limited to the optimization problem posed and the datasets used. Future studies can explore other approaches to optimize the predictive ability of the KPLS regression, reduce the computational cost and dispersion in the estimates, include more kernel functions in the analysis, and test with more datasets.

REFERENCES

[1] P. Geladi and B. R. Kowalski, "Partial least-squares regression: A tutorial," *Analytica Chim. Acta*, vol. 185, pp. 1–17, 1986.

[2] H. Abdi, "Partial least squares regression and projection on latent structure regression (PLS Regression)," *WIREs Comput. Statist.*, vol. 2, no. 1, pp. 97–106, Jan. 2010.

[3] Y. Gao, X. Kong, C. Hu, Z. Zhang, H. Li, and L. Hou, "Multivariate data modeling using modified kernel partial least squares," *Chem. Eng. Res. Des.*, vol. 94, pp. 466–474, Feb. 2015.

[4] R. Rosipal and L. J. Trejo, "Kernel partial least squares regression in reproducing kernel Hilbert space," *J. Mach. Learn. Res.*, vol. 2, pp. 97–123, Mar. 2001.

[5] H. Zhang, S. Liu, S. Moraca, and R. Ojstersek, "An effective use of hybrid metaheuristics algorithm for job shop scheduling problem," *Int. J. Simul. Model.*, vol. 16, no. 4, pp. 644–657, Dec. 2016.

[6] V. E. de Almeida, A. de Araújo Gomes, D. D. de Sousa Fernandes, H. C. Goicoechea, R. K. H. Galvão, and M. C. U. Araújo, "Vis-NIR spectrometric determination of Brix and sucrose in sugar production samples using kernel partial least squares with interval selection based on the successive projections algorithm," *Talanta*, vol. 181, pp. 38–43, May 2018.

[7] Y. Fu, U. Kruger, Z. Li, L. Xie, J. Thompson, D. Rooney, J. Hahn, and H. Yang, "Cross-validatory framework for optimal parameter estimation of KPCA and KPLS models," *Chemometrics Intell. Lab. Syst.*, vol. 167, pp. 196–207, Aug. 2017.

[8] R. García-Ródenas, J. C. García-García, J. López-Fidalgo, J. Á. Martín-Baos, and W. K. Wong, "A comparison of general-purpose optimization algorithms for finding optimal approximate experimental designs," *Comput. Statist. Data Anal.*, vol. 144, Apr. 2020, Art. no. 106844.

[9] S.-W. Lin, K.-C. Ying, S.-C. Chen, and Z.-J. Lee, "Particle swarm optimization for parameter determination and feature selection of support vector machines," *Expert Syst. Appl.*, vol. 35, no. 4, pp. 1817–1824, Nov. 2008.

[10] P. Žuvela, K. Macur, J. Jay Liu, and T. Baczek, "Exploiting nonlinear relationships between retention time and molecular structure of peptides originating from proteomes and comparing three multivariate approaches," *J. Pharmaceutical Biomed. Anal.*, vol. 127, pp. 94–100, Aug. 2016.

[11] L. M. Rios and N. V. Sahinidis, "Derivative-free optimization: A review of algorithms and comparison of software implementations," *J. Global Optim.*, vol. 56, no. 3, pp. 1247–1293, Jul. 2012.

[12] J. D. Mello-Román and A. Hernández, "KPLS optimization approach using genetic algorithms," in *Proc. Int. Workshop Stat. Methods Artif. Intell. (IWSMAI)*, Warsaw, Poland, Apr. 2020, pp. 1153–1160.

[13] S. Liu, J. Tang, and D. Yan, "Multi-Kernel Partial Least Squares Regression based on Adaptive Genetic Algorithm," in *Proc. Int. Conf. Autom., Mech. Control Comput. Eng.*, 2015, pp. 141–144.

[14] M. Jalali-Heravi and A. Kyani, "Application of genetic algorithm-kernel partial least square as a novel nonlinear feature selection method: Activity of carbonic anhydrase II inhibitors," *Eur. J. Medicinal Chem.*, vol. 42, no. 5, pp. 649–659, May 2007.

[15] Y. Wang, J.-J. Huang, N. Zhou, D.-S. Cao, J. Dong, and H.-X. Li, "Incorporating PLS model information into particle swarm optimization for descriptor selection in QSAR/QSPR," *J. Chemometrics*, vol. 29, no. 12, pp. 627–636, Dec. 2015.

[16] Y. Ying, W. Jin, H. Yu, B. Yu, J. Shan, S. Lv, D. Zhu, Q. Jin, and Y. Mu, "Development of particle swarm optimization-support vector regression (PSO-SVR) coupled with microwave plasma torch-atomic emission spectrometry for quality control of ginsengs," *J. Chemometrics*, vol. 31, no. 1, pp. e2862, Nov. 2016.

[17] M. Goodarzi and L. dos Santos Coelho, "Firefly as a novel swarm intelligence variable selection method in spectroscopy," *Analytica Chim. Acta*, vol. 852, pp. 20–27, Dec. 2014.

[18] H. Jin, X. Chen, J. Yang, and L. Wu, "Adaptive soft sensor modeling framework based on just-in-time learning and kernel partial least squares regression for nonlinear multiphase batch processes," *Comput. Chem. Eng.*, vol. 71, pp. 77–93, Dec. 2014.

[19] X. Wang, P. Wang, X. Gao, and Y. Qi, "On-line quality prediction of batch processes using a new kernel multiway partial least squares method," *Chemometric Intell. Lab. Syst.*, vol. 158, pp. 138–145, Nov. 2016.

[20] J. Shawe-Taylor and C. Nello, *Kernel Methods for Pattern Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[21] M. Tenenhaus, *La Régression PLS: Théorie et Pratique*. Paris, France: Editions Technip, 1998.

[22] S. Wold, M. Sjöström, and L. Eriksson, "PLS-regression: A basic tool of chemometrics," *Chemometric Intell. Lab. Syst.*, vol. 58, no. 2, pp. 109–130, Oct. 2001.

[23] L. Eriksson, E. Johansson, H. Antti, and E. Holmes, "Multi-and megavariate data analysis," in *Metabonomics Toxicity Assessment*. Boca Raton, FL, USA: CRC Press, Mar. 2005, pp. 263–336.

[24] M. Stone, "Cross-validatory choice and assessment of statistical predictions," *J. Roy. Stat. Soc., B (Methodol.)*, vol. 36, no. 2, pp. 111–133, Jan. 1974.

[25] B. Bischl, O. Mersmann, H. Trautmann, and C. Weihs, "Resampling methods for meta-model validation with recommendations for evolutionary computation," *Evol. Comput.*, vol. 20, no. 2, pp. 249–275, Jun. 2012.

[26] S. Xue and X. Yan, "A new kernel function of support vector regression combined with probability distribution and its application in chemometrics and the QSAR modeling," *Chemometrics Intell. Lab. Syst.*, vol. 167, pp. 96–101, Aug. 2017.

[27] H. Huang, B. Chen, and C. Liu, "Safety monitoring of a super-high dam using optimal kernel partial least squares," *Math. Problems Eng.*, vol. 2015, pp. 1–13, 2015.

[28] M. Gendreau and J.-Y. Potvin, Eds., *Handbook of Metaheuristics* (International Series in Operations Research & Management Science). New York, NY, USA: Springer, 2010.

[29] E. Osaba, R. Carballedo, F. Diaz, E. Onieva, A. D. Masegosa, and A. Perallos, "Good practice proposal for the implementation, presentation, and comparison of metaheuristics for solving routing problems," *Neurocomputing*, vol. 271, pp. 2–8, Jan. 2018.

[30] D. E. Kvasov and M. S. Mukhametzhanov, "Metaheuristic vs. deterministic global optimization algorithms: The univariate case," *Appl. Math. Comput.*, vol. 318, pp. 245–259, Feb. 2018.

[31] A. P. Piotrowski, M. J. Napiorkowski, J. J. Napiorkowski, and P. M. Rowinski, "Swarm intelligence and evolutionary algorithms: Performance versus speed," *Inf. Sci.*, vol. 384, pp. 34–85, Apr. 2017.

[32] J. Yi, D. Huang, H. He, W. Zhou, Q. Han, and T. Li, "A novel framework for fault diagnosis using kernel partial least squares based on an optimal preference matrix," *IEEE Trans. Ind. Electron.*, vol. 64, no. 5, pp. 4315–4324, May 2017.

[33] K. A. M. Attia, M. W. I. Nassar, M. B. El-Zeiny, and A. Serag, "Firefly algorithm versus genetic algorithm as powerful variable selection tools and their effect on different multivariate calibration models in spectroscopy: A comparative study," *Spectrochim. Acta A, Mol. Biomol. Spectrosc.*, vol. 170, pp. 117–123, Jan. 2017.

[34] Z.-M. Gao and J. Zhao, "An improved grey wolf optimization algorithm with variable weights," *Comput. Intell. Neurosci.*, vol. 2019, Jun. 2019, Art. no. 2981282.

[35] J. H. Holland, *Adaptation in Natural and Artificial Systems. An Introductory Analysis With Application to Biology, Control, and Artificial Intelligence*. Ann Arbor, MI, USA: Univ. Michigan Press, 1975, pp. 439–444.

[36] D. B. Fogel, *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence*. vol. 1. Hoboken, NJ, USA: Wiley, 2006.

[37] O. Roeva and S. Fidanova, "Comparison of different Metaheuristic algorithms based on InterCriteria analysis," *J. Comput. Appl. Math.*, vol. 340, pp. 615–628, Oct. 2018.

[38] H. Noorizadeh, S. Sobhan Ardakani, T. Ahmadi, S. S. Mortazavi, and M. Noorizadeh, "Application of genetic algorithm-kernel partial least square as a novel non-linear feature selection method: Partitioning of drug molecules," *Drug Test. Anal.*, vol. 5, no. 2, pp. 89–95, Mar. 2011.

[39] A. B. E. L. Fiszelew, R. García-Martínez, and T. de Buenos Aires, "Generación automática de redes neuronales con ajuste de parámetros basado en algoritmos genéticos," *Revista del Instituto Tecnológico de Buenos Aires*, vol. 26, pp. 76–101, Feb. 2002.

[40] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw.*, vol. 4, Nov. 1995, pp. 1942–1948.

[41] X. S. Yang, *Nature-Inspired Metaheuristic Algorithms*. Frome, U.K.: Luniverpress, 2010.

[42] I. Fister, I. Fister, X.-S. Yang, and J. Brest, "A comprehensive review of firefly algorithms," *Swarm Evol. Comput.*, vol. 13, pp. 34–46, Dec. 2013.

[43] P. J. Vidal and A. C. Olivera, "DNA fragment assembling using a novel GPU firefly algorithm," *Dyna*, vol. 85, no. 204, pp. 108–116, 2018.

[44] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Mar. 2014.

[45] Z.-M. Gao and J. Zhao, "An improved grey wolf optimization algorithm with variable weights," *Comput. Intell. Neurosci.*, vol. 2019, pp. 1–13, Jun. 2019.

[46] P. Niu, S. Niu, N. liu, and L. Chang, "The defect of the Grey Wolf optimization algorithm and its verification method," *Knowl.-Based Syst.*, vol. 171, pp. 37–43, May 2019.

[47] Y. Sun, N. V. Sahinidis, A. Sundaram, and M.-S. Cheon, "Derivative-free optimization for chemical product design," *Current Opinion Chem. Eng.*, vol. 27, pp. 98–106, Mar. 2020.

[48] J. Larson, M. Menickelly, and S. M. Wild, "Derivative-free optimization methods," *Acta Numerica*, vol. 28, pp. 287–404, May 2019.

[49] J. A. Nelder and R. Mead, "A simplex method for function minimization," *Comput. J.*, vol. 7, no. 4, pp. 308–313, Jan. 1965.

[50] R. Hooke and T. A. Jeeves, "'Direct Search' Solution of Numerical and Statistical problems," *J. ACM*, vol. 8, no. 2, pp. 212–229, Apr. 1961.

[51] G. Sanchez. (2012). *Plsdepot: Partial Least Squares (PLS) Data Analysis Methods*. R package version 0.1.17. [Online]. Available: https://cran.r-project.org/web/packages/plsdepot/index.html

[52] A. Karatzoglou, A. Smola, K. Hornik, and A. Zeileis, "Kernlab-an S4 package for kernel methods in R," *J. Stat. Softw.*, vol. 11, no. 9, PP. 1–20, 2004.

[53] L. S. Riza and E. P. Nugroho, "MetaheuristicOpt: An R package for optimisation based on meta-heuristics algorithms," *Pertanika J. Sci. Technol.*, vol. 26, no. 3, pp. 1401–1412, 2018.

[54] R. Varadhan and H. W. Borchers, "Dfoptim: Derivative-free optimization," R package version 2016.7-1, Tech. Rep., 2016. [Online]. Available: https://cran.r-project.org/web/packages/dfoptim/index.html

[55] C. D. Lin, C. M. Anderson-Cook, M. S. Hamada, L. M. Moore, and R. R. Sitter, "Using genetic algorithms to design experiments: A review," *Qual. Rel. Eng. Int.*, vol. 31, no. 2, pp. 155–167, Jan. 2014.

[56] S. Asghari and N. J. Navimipour, "Nature inspired meta-heuristic algorithms for solving the service composition problem in the cloud environments," *Int. J. Commun. Syst.*, vol. 31, no. 12, pp. e3708, May 2018.

[57] J. Wang and D. Zhu, "Derivative-free restrictively preconditioned conjugate gradient path method without line search technique for solving linear equality constrained optimization," *Comput. Math. Appl.*, vol. 73, no. 2, pp. 277–293, Jan. 2017.

[58] F. Xue, W. Lu, C. J. Webster, and K. Chen, "A derivative-free optimization-based approach for detecting architectural symmetries from 3D point clouds," *ISPRS J. Photogramm. Remote Sens.*, vol. 148, pp. 32–40, Feb. 2019.

[59] J. Derrac, S. García, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm Evol. Comput.*, vol. 1, no. 1, pp. 3–18, Mar. 2011.

[60] A. Tsanas, M. Little, P. McSharry, and L. Ramig, "Accurate telemonitoring of Parkinson's disease progression by noninvasive speech tests," *IEEE Trans. Biomed. Eng.*, vol. 57, no. 4, pp. 884–893, Apr. 2010, doi: 10.1109/TBME.2009.2036000.

[61] D. Dua and C. Graff. (2019). *UCI Machine Learning Repository*. Irvine, CA, USA: Univ. of California, School of Information and Computer Science. [Online]. Available: http://archive.ics.uci.edu/ml and http://archive.ics.uci.edu/ml/citation_policy.html

[62] M. Little, P. McSharry, E. Hunter, J. Spielman, and L. Ramig, "Suitability of dysphonia measurements for telemonitoring of Parkinson's disease," *IEEE Trans. Biomed. Eng.*, vol. 56, no. 4, pp. 1015–1022, Apr. 2009, doi: 10.1109/TBME.2008.2005954.

[63] E. Benmalek, J. Elmhamdi, and A. Jilbab, "UPDRS tracking using linear regression and neural network for Parkinson's disease prediction," *Int. J. Emerg. Trends Technol. Comput. Sci.*, vol. 4, no. 6, pp. 189–193, 2015.

[64] B. E. Sakar, M. E. Isenkul, C. O. Sakar, A. Sertbas, F. Gurgen, S. Delil, H. Apaydin, and O. Kursun, "Collection and analysis of a Parkinson speech dataset with multiple types of sound recordings," *IEEE J. Biomed. Health Inform.*, vol. 17, no. 4, pp. 828–834, Jul. 2013.

[65] P. Cortez and A. M. G. Silva, "Using data mining to predict secondary school student performance," in *Proc. 5th Annu. Future Bus. Technol. Conf.*, A. Brito and J. Teixeira, Eds. Porto, Portugal: EUROSIS, Apr. 2008, pp. 5–12.

[66] W. H. Kruskal and W. A. Wallis, "Use of ranks in one-criterion variance analysis," *J. Amer. Stat. Assoc.*, vol. 47, no. 260, pp. 583–621, Dec. 1952.

[67] S. Lee and D. K. Lee, "What is the proper way to apply the multiple comparison test?" *Korean J. Anesthesiol.*, vol. 71, no. 5, pp. 353–360, Oct. 2018.

[68] M. C. Shingala and A. Rajyaguru, "Comparison of post hoc tests for unequal variance," *Int. J. New Technol. Sci. Eng.*, vol. 2, no. 5, 22-33, 2015.

[69] P. H. Ramsey and P. P. Ramsey, "Power of pairwise comparisons in the equal variance and unequal sample size case," *Brit. J. Math. Stat. Psychol.*, vol. 61, no. 1, pp. 115–131, May 2008.

JORGE DANIEL MELLO-ROMÁN received the master's degree in advanced mathematics–statistics and operations research from the National University of Distance Education (UNED), Spain. He is currently pursuing the Ph.D. degree in mathematical engineering with the Statistics and Operations Research Program, Complutense University of Madrid. He is currently a Professor and the Dean of the Faculty of Exact and Technological Sciences, National University of Concepción, Paraguay.

ADOLFO HERNÁNDEZ received the Ph.D. degree in mathematical sciences (statistics) from the Carlos III University of Madrid. He is currently an Associate Professor with the Complutense University of Madrid, where he is also the Head of the Research Group "Statistical Methods and Big Data Applied to the Economy, Tourism, and other Social Sciences." He was a Former Lecturer and a Senior Lecturer with the Carlos III University of Madrid, the Autonoma University of Madrid, the University of Salamanca, and the University of Exeter, U.K.