Hybrid methodology based on Bayesian Optimization and GA-PARSIMONY for searching parsimony models by combining hyperparameter optimization and feature selection

F.J. Martinez-de-Pison[a,*], R. Gonzalez-Sendino[a], A. Aldama[a], J. Ferreiro[a], E. Fraile[a]

[a]*EDMANS Group, Department of Mechanical Engineering, University of La Rioja, Logroño, Spain.*

Abstract

This paper presents a Hybrid methodology that combines Bayesian Optimization (BO) with a constrained version of the GA-PARSIMONY method to obtain parsimony models. The proposal is designed to reduce the big computational efforts associated to the use of GA-PARSIMONY alone. The method is initialized with BO to obtain favorable initial model parameters. With these parameters, a constrained GA-PARSIMONY is implemented to generate accurate parsimony models using feature reduction, data transformation and parsimonious model selection. Experiments with Extreme Gradient Boosting Machines (XGBoost) and ten UCI databases demonstrate that the Hybrid methodology obtains models analogous to those of GA-PARSIMONY while achieving significant reductions on the elapsed time in seven of the ten datasets.

Keywords: GA-PARSIMONY, bayesian optimization, hyperparameter optimization, parsimony models, genetic algorithms

1. Introduction

Hyperparameter optimization (HO) is extremely important for finding accurate models. Also, feature selection (FS) is useful for seeking the less complex models among solutions with similar accuracy. These parsimonious models are more robust against perturbations or noise, easier to maintain, and besides, they mitigate the effects of the curse of dimensionality.

In the last years, there is an increasing interest in reducing the human efforts in HO and FS because these tasks are time-consuming and quite tedious. Newest learning methods such as deep learning or gradient boosting machines have up to a dozen of tuning parameters, also known as hyper-parameters, which hinders the use of traditional optimization methods such as

*Corresponding author
Email address: fjmartin@unirioja.es (F.J. Martinez-de-Pison)

grid or random search. Therefore, companies are demanding new methodologies to automatize these processes, because they prefer to invest their efforts in other critical KDD tasks such as data transformation or feature engineering that are harder to automatize [13].

Among the different existing methods to tackle this issue, soft computing (SC) seems to be an effective approach to reduce the computational costs [23, 35, 4, 7]. There is an increasing number of studies reporting SC strategies that combine FS and HO applied to multiple fields [15, 9, 33, 14, 8, 34, 5, 25]. New libraries are emerging to perform HO with Bayesian Optimization (BO) like *Hyperopt* [2] in Python, or *mlr* [3] and *rBayesianOptimization* in R. In addition, there are other tools that are focused on the optimization of more KDD stages such as algorithm selection (AS), data transformation (DT), dimensional reduction (DR), model selection (MS) or feature construction (FC). For example, the *SUMO-Toolbox* [12] from MATLAB adopts different plugins for each of the different KDD stages. They can be optimized with other 'meta' plugins available in the toolbox. The *Auto-WEKA* [30] from *Weka* suite also combines MS and HO. TPOT [18] is another library in Python that automatically optimizes machine learning pipelines using genetic programming. These pipelines consist on several KDD tasks as FS, DT, FC or MS, among others.

In this context, we proposed GA-PARSIMONY [31, 24], a Genetic Algorithm (GA) methodology whose main objective is to obtain accurate parsimonious models. It optimizes HO, DT, and FS with a new model selection process based on a double criteria that considers accuracy and complexity in two steps. Despite the fact that the methodology has been successfuly applied in several practical fields [1, 10, 32], it might be too computationally expensive when implemented with large and high dimensional databases. Our main objective here is to obtain models as accurate as those obtained with GA-PARSIMONY but reducing the reduced computational effort. For that, we develop a new Hybrid methodolody that combines BO and GA-PARSIMONY, and we test this new approach in ten UCI datasets.

The rest of the paper is organized as follows: Section 2 presents a brief description of BO, GA-PARSIMONY and the Hybrid method. Section 3 describes the experiments performed with the three methods to obtain parsimonious XGBoost models in ten UCI datasets. In Section 4 analysis of the experiment results are shown. Finally, Section 5 presents the conclusions and suggestions for further research.

2. Materials and Methods

2.1. Extreme Gradient Boosting Machines

eXtreme Gradient Boosting (XGBoost) [6] is one of the most popular machine learning methods. This powerful method is based on gradient boosting machines (GBM) [11]. GBM use a gradient-descent based algorithm that optimizes a differentiable loss function to create a boosting ensemble of weak prediction models. The main idea is to construct each new additive base-learner to be maximally correlated with the negative gradient of the loss function of the ensemble. However, XGBoost with tree-based learners is computationally more efficient and scalable than GBM. It incorporates more regularization strategies to reduce over-fitting and control model complexity, such us the limitation of the minimum loss reduction at each tree partition, the sum of instances weight per leaf or the depth of each tree. It also incorporates Lasso (L1) and Ridge (L2) penalties, similar to other machine learning methods. Moreover, it integrates "random subspaces" and "random subsampling" parameters to shrink the variance.

The high number of model parameters increases the computational efforts of the tuning process. Besides, despite the fact that tree-based ensemble methods have good performance with high-dimensional data, the inclusion of irrelevant or noisy features can degrade the accuracy of these models [19]. Therefore, there is an increasing interest in developing new SC methods to efficiently optimize HO and FS and obtain models with good generalization capabilities.

2.2. Bayesian optimization

Since mid of 2000s, *Bayesian optimization* (BO) has become one interesting alternative among other HO classical alternatives like random search or grid search [22]. BO uses Bayesian models based on *Gaussian processes* (GP) to formalize the relationship between model error/accuracy (y_n) with its parameters by means of a sequential design strategy. According to GP, any finite set of N points, where $\{\mathbf{x}_n \in \boldsymbol{\varnothing}\}_{n=1}^{N}$, induces a multivariate Gaussian distribution on $\Re^n$. Then, GP defines a powerful prior distribution on functions $f : \boldsymbol{\varnothing} \rightarrow \Re$ where the nth model performance is obtained from $f(\mathbf{x}_n)$ and the marginals and conditionals are calculated by the marginalization properties of the Gaussian distribution. These properties are determined by a predefined mean function $m\colon \chi \rightarrow \Re$ and a positive-definitive kernel or covariance function $k\colon \chi \times \chi \rightarrow \Re$.

From a practical point of view [28], BO starts with the evaluation of a small number of N models with a random set of parameters $\mathbf{x}_n$ where $y_n \sim \mathcal{N}(f(\mathbf{x}_n, v))$ is the n^{th} measured model performance and v is the variance of functions' noise. Thus, considering that $f(\mathbf{x})$ is obtained from a Gaussian process prior and with the precomputed experiments, a posterior over function

$a(\mathbf{x})$ is induced. This function, denoted acquisition function, depends on the model through its predictive mean function $\mu(\mathbf{x};\{\mathbf{x}_n,y_n\},\theta)$ and predictive variance function $\sigma^2(\mathbf{x};\{\mathbf{x}_n,y_n\},\theta)$. Therefore, next point is evaluated by $\mathbf{x}_{next} = argmax_{\mathbf{x}}\, a(\mathbf{x})$ balancing the search of places with high variance (exploration) and places with low mean (exploitation).

Among the available acquisition functions [27], *GP Upper Confidence Bound* (GP-UCB) has shown a good performance in *hyperparameter tuning* [29]. This acquisition function can be expressed as:

$$a_{LCB} = \mu(\mathbf{x}) - \kappa\sigma(\mathbf{x}) \quad , \tag{1}$$

where κ balances exploration and exploitation. Also, *squared exponential kernel* (Eq. 2) is often a default choice as covariance function for Gaussian process regression.

$$K_{SE}(\mathbf{x},\mathbf{x}') = \theta_0\, exp\{\frac{1}{2}r^2(\mathbf{x},\mathbf{x}')\} \qquad r^2(\mathbf{x},\mathbf{x}') = \sum_{d=1}^{D}(x_d - x'_d)^2/\theta_d^2 \quad , \tag{2}$$

2.3. GA-PARSIMONY methodology

GA-PARSIMONY is a SC methodology based on Genetic Algorithms (GA) and designed for obtaining precise overall parsimonious models automatically [31, 24]. It includes HO, FS, and DT in the GA optimization process and it has a flowchart similar to other classical GA methods. The main novelty is the design of a *parsimonious model selection* process (PMS) arranged in two stages. First, the best models are sorted by their fitness function (J), which is an error or accuracy metric, and next, individuals with similar Js are rearranged based on their complexities. Models with less complexity are therefore promoted to the top positions of each generation. This choice of less complex solutions among those with similar accuracy fosters the generation of robust solutions with better generalization capabilities.

GA-PARSIMONY has successfully been applied to obtain accurate parsimonious models with the most popular machine learning techniques such as Support Vector Regression (SVR), Random Forest (RF) or Artificial Neural Networks (ANNs) in different fields: mechanical design [10], solar radiation forecasting [1], industrial processes [26], and hotel room demand estimation [32]. Additionaly, a preliminary evaluation of the methodolgy was perfomed with XGboost using several high dimensional databases and different complexity metrics [20]. GA-PARSIMONY performed well only with HO, but previous experiments have demonstrated that, choosing the number of features as measure of the model complexity is a good metric to obtain better parsimonious solutions when HO, FS and PMS are used with this method.

2.4. Hybrid method based on Bayesian Optimization and GA-PARSIMONY

Although GA-PARSIMONY is able to generate accurate and parsimonious models, the implementation of this methodology with large and/or high dimensional database can be too computationally expensive even using parallel computing techniques. A Hybrid method that combines BO and GA-PARSIMONY is presented here to reduce the computational costs (Fig. 1) associated to the GA-PARSIMONY. The main idea is to use BO in a first stage with all features to obtain the best model parameters. Next, GA-PARSIMONY with FS and PMS is used for seeking the best features of the parsimonious model with the fixed parameters obtained in the first step.

3. Experiments

3.1. Datasets and validation process

The Hybrid methodology with XGBoost was evaluated against the use of either BO or GA-PARSIMONY alone. The experiments were conducted with ten UCI datasets (Table 1), which were split into a validation set (80% of samples) and a testing set (20% of samples), in order to check the generalization capability of each model. The validation was made in terms of the mean of the Root Mean Squared Error ($RMSE$) calculated with a 5 repeated 4-fold CV ($RMSE_{val}^{mean}$).

3.2. GA-PARSIMONY settings

The fitness function selected was $J = RMSE_{val}^{mean}$ while the maximum difference of J to consider similar individuals and promote parsimonious solutions into the re-ranking process was set to 0.01%. The elitism percentage was set to 25%, the selection method, *random uniform*, and crossing was perfomed with *heuristic blending* [17]. A mutation percentage of 10% was used except for the best two elitists of each generation that were not mutated. The population size was set to $P = 64$ and the maximum number of generations to $G = 100$. However, an early stopping strategy was implemented when the J of the best individual did not decrease more than 0.01% in 10 generations, $G_{early} = 10$.

XGBoost parameters were defined within the following ranges: number of trees, $nrounds =$ [10, 2000], maximum depth of a tree, $max\_depth =$ [2, 20], minimum sum of instance weight needed in a child, $min\_child\_weight =$ [1, 20], *lasso* regularization term on weights, $alpha =$ [0.0, 1.00], *ridge* regularization term on weights, $lambda =$ [0.0, 1.00], subsample ratio of the training instances, $subsample =$ [0.60, 1.00], and subsample ratio of columns when constructing each tree, $colsample\_bytree = [0.80,\ 1.00]$. Random seed was fixed to 1234 and learning rate, *eta*, to 0.01.

Also, k exponent to transform the dependent variable was used in the following way $y^* = y^k$. In this case, the range set for this parameter was $k = [0.20,\ 1.79]$.

The representation of each individual (i) and generation (g) was a chromosome (Eq. 3).

$$\begin{aligned} \lambda_g^i = [&nrounds,\ max\_depth,\ min\_child\_weight,\ alpha,\\ &lambda,\ subsample,\ colsample\_bytree,\ k,\ Q] \end{aligned} \tag{3}$$

where the first seven values are the XGBoost parameters, k is the exponent to transform the dependent variable and Q is a binary-coded array that included the selected features.

3.3. Bayesian optimization settings

BO parameter bounds were identical to GA-PARSIMONY settings. The acquisition function selected was the GP-UCB while the covariance function was the squared exponential kernel with $\kappa = 2.576$. The number of initial points was set to 10, and the number of iterations for the optimization process to 50.

3.4. Hybrid method settings

The first stage of the Hybrid method was based on the same BO settings as those described in Section 3.3. In the second stage, GA-PARSIMONY performed FS and PMS with the best model parameters obtained during the first stage. Chromosomes at each generation were only defined by the binary-coded array $\lambda_g^i = Q$ because HO was disabled. Except λ_g^i, the rest of GA settings were similar to those described in Section 3.2.

3.5. Computational resources

All the experiments were implemented in 28-core servers of the *Beronia* cluster at the Universidad de La Rioja, using the statistical software `R` [21] and the following contributing packages: `XGBoost` [6] and `GAparsimony` [16].

4. Results and Discussion

Table 1 summarizes the results obtained with the ten UCI high-dimensional datasets. Among the three methods, GA-PARSIMONY obtains parsimonious models with the best $RMSE_{tst}^{mean}$ in six of the ten datasets, while having similar errors to those of the Hybrid method in the other four datasets. However, the elapsed time required by the Hybrid method was considerably reduced for large datasets.

Comparing GA-PARSIMONY with BO, an improvement of $RMSE_{tst}^{mean}$ is observed for all datasets in general and for *Housing*, *Pol* and *Puma* in particular. Also, *#FT* is reduced in five datasets: *Ailerons*, *Bank*, *Blog*, *Elevators*, and *Puma*. Otherwise, the Hybrid methodology generates analogous $RMSE_{tst}^{mean}$ to the GA-PARSIMONY in nine datasets but with a significant reduction on the elapsed time for the largest ones.

Figure 2 depicts the evolution of the $RMSE_{val}$ and $RMSE_{tst}$ for the elitist individuals using the GA-PARSIMONY and *Bank* database, without using early stopping to observe the optimization convergence errors. Figure 3 shows the same evolution for the second stage of the Hybrid method where GA-PARSIMONY is used without HO. In this second optimization, XGBoost parameters were obtained from the previous BO process (stage 1) computed with all the database features. Comparing both figures, it can be observed than the optimization process converge faster in the Hybrid methodology than in GA-PARSIMONY. With this database and using an early stopping criteria of 10 generations ($G_{early} = 10$), the Hybrid solution stops at the $20th$ generation while the GA-PARSIMONY does at the $35th$, leading to the observed reduction of the elapsed time.

Table 2 shows the *p-values* obtained with the Wilcoxon test for the three methodologies. Despite the fact that the GA-PARSIMONY obtains a smaller $RMSE_{tst}^{mean}$ than that from BO, the differences are only statistically significant in four databases: *Blog*, *Housing*, *Pol* and *Puma*. However, there is an important reduction of *#FT* for all databases, leading to parsimonious models with similar or better accuracy. With respect to the Hybrid methodology, errors are similar to those of GA-PARSIMONY. The only exception appears in *Pol* dataset, although *p-value* is close to the 95% of confidence level in this case (*p-value*=0.05).

The stages of the Hybrid proposal are summarized in Table 3. The last column includes the time reduction in the Stage 2 of the Hybrid method compared to the GA-PARSIMONY. Both of them were parallelized in 28-Core servers.

In the first step, BO is applied for extracting the best model parameters with all features of the database. In some cases, the execution time is large because BO cannot be parallelized. In the second stage and with these parameters, FS is performed with GA-PARSIMONY but without HO. Thus, the *#Gen* is substantially reduced compared to the use of GA-PARSIMONY with FS and HO in nine of the ten databases. Therefore, the most important reduction in the elapsed time is obtained in this stage, with a relative reduction in the execution time exceeding

46% in these databases. Besides, it is important to highlight the big elapsed time contraction for large databases such as *Elevators* or *Pol*.

Figure 4 shows the relative reduction of the execution time between the Hybrid methodology and GA-PARSIMONY. A significant reduction was achieved with the Hybrid proposal in seven of the ten databases. The exceptions were *cpu*, in which GA-PARSIMONY stopped earlier than the Hybrid method, and small databases such as *housing*, where non-parallelizable BO was more computational expensive than stage 2. However, it can be observed that the Hybrid methodology clearly reached important time reductions for large databases such as *Ailerons*, *Bank*, *Elevators*, *Pol* or *Puma*.

5. Conclusions

This article presents a new Hybrid methodology that combines Bayesian Optimization and GA-PARSIMONY to seek high accuracy and parsimonious models while reducing the execution time. Although GA-PARSIMONY obtains better models than BO by combining Hyperparameter Optimization (HO), parsimonious model selection (PMS), feature selection (FS), and data transformation (DT), the computational efforts with large and high dimensional databases are still significant. The Hybrid proposal uses BO to obtain good initial model parameters previous to the FS, DT and PMS, which are optimized with GA-PARSIMONY without HO.

Experiments with ten UCI databases demonstrate that the Hybrid methodology generates similar parsimonious solutions than the GA-PARSIMONY while reducing the execution time in eight of the ten datasets. Further experiments are still required with additional high dimensional databases to obtain more detailed conclusions.

Acknowledgements

We are greatly indebted to *Banco Santander* for the APPI16/05 fellowship and to the University of La Rioja for the EGI16/19 fellowship. This work used the Beronia cluster (Universidad de La Rioja), which is supported by FEDER-MINECO grant number UNLR-094E-2C-225.

[1] Antonanzas-Torres, F., Urraca, R., Antonanzas, J., Fernandez-Ceniceros, J., de Pison, F.M.: Generation of daily global solar irradiation with support vector machines for regression. Energy Conversion and Management 96, 277 – 286 (2015) 1, 2.3

[2] Bergstra, J., Komer, B., Eliasmith, C., Yamins, D., Cox, D.D.: Hyperopt: a python library for model selection and hyperparameter optimization. Computational Science & Discovery 8(1), 014008 (2015) 1

[3] Bischl, B., Lang, M., Kotthoff, L., Schiffner, J., Richter, J., Studerus, E., Casalicchio, G., Jones, Z.M.: mlr: Machine learning in R. Journal of Machine Learning Research 17(170), 1–5 (2016) 1

[4] Caamaño, P., Bellas, F., Becerra, J.A., Duro, R.J.: Evolutionary algorithm characterization in real parameter optimization problems. Applied Soft Computing 13(4), 1902–1921 (2013) 1

[5] Chen, N., Ribeiro, B., Vieira, A., Duarte, J., Neves, J.C.: A genetic algorithm-based approach to cost-sensitive bankruptcy prediction. Expert Systems with Applications 38(10), 12939–12945 (2011) 1

[6] Chen, T., He, T., Benesty, M.: XGBoost: Extreme Gradient Boosting machines. (2015), `https://github.com/dmlc/xgboost`, R package version 0.4-3 2.1, 3.5

[7] Corchado, E., Wozniak, M., Abraham, A., de Carvalho, A.C.P.L.F., Snásel, V.: Recent trends in intelligent data analysis. Neurocomputing 126, 1–2 (2014) 1

[8] Dhiman, R., Saini, J., Priyanka: Genetic algorithms tuned expert model for detection of epileptic seizures from EEG signatures. Applied Soft Computing 19(0), 8 – 17 (2014) 1

[9] Ding, S.: Spectral and wavelet-based feature selection with particle swarm optimization for hyperspectral classification. Journal of Software 6(7), 1248–1256 (2011) 1

[10] Fernandez-Ceniceros, J., Sanz-Garcia, A., Antonanzas-Torres, F., de Pison, F.M.: A numerical-informational approach for characterising the ductile behaviour of the T-stub component. Part 2: Parsimonious soft-computing-based metamodel. Engineering Structures 82, 249 – 260 (2015) 1, 2.3

[11] Friedman, J.H.: Greedy function approximation: A gradient boosting machine. The Annals of Statistics 29(5), 1189–1232 (2001) 2.1

[12] Gorissen, D., Couckuyt, I., Demeester, P., Dhaene, T., Crombecq, K.: A surrogate modeling and adaptive sampling toolbox for computer based design. J. Mach. Learn. Res. 11, 2051–2055 (2010) 1

[13] Hashem, I.A., Yaqoob, I., Anuar, N.B., Mokhtar, S., Gani, A., Ullah Khan, S.: The rise of big data on cloud computing: Review and open research issues. Information Systems 47, 98–115 (2015) 1

[14] Huang, C.L., Dun, J.F.: A distributed PSO-SVM hybrid system with feature selection and parameter optimization. Applied Soft Computing 8(4), 1381 – 1391 (2008) 1

[15] Huang, C.J., Chen, Y.J., Chen, H.M., Jian, J.J., Tseng, S.C., Yang, Y.J., Hsu, P.A.: Intelligent feature extraction and classification of anuran vocalizations. Applied Soft Computing 19(0), 1 – 7 (2014) 1

[16] Martínez-De-Pisón, F.J.: GAparsimony: GA-based optimization R package for searching accurate parsimonious models. (2017), `https://github.com/jpison/GAparsimony`, R package version 0.9-1 3.5

[17] Michalewicz, Z., Janikow, C.Z.: Handling constraints in genetic algorithms. In: ICGA. pp. 151–157 (1991) 3.2

[18] Olson, R.S., Bartley, N., Urbanowicz, R.J., Moore, J.H.: Evaluation of a tree-based pipeline optimization tool for automating data science. In: Proceedings of the Genetic and Evolutionary Computation Conference 2016. pp. 485–492. GECCO '16, ACM, New York, NY, USA (2016) 1

[19] Perner, P.: Improving the accuracy of decision tree induction by feature preselection. Applied Artificial Intelligence 15(8), 747–760 (2001) 2.1

[20] Martinez-de Pison, F.J., Fraile-Garcia, E., Ferreiro-Cabello, J., Gonzalez, R., Pernia, A.: Searching Parsimonious Solutions with GA-PARSIMONY and XGBoost in High-Dimensional Databases, pp. 201–210. Springer International Publishing, Cham (2017) 2.3

[21] R Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2013) 3.5

[22] Rasmussen, C.E., Williams, C.K.I.: Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning). The MIT Press (2005) 2.2

[23] Reif, M., Shafait, F., Dengel, A.: Meta-learning for evolutionary parameter optimization of classifiers. Machine Learning 87(3), 357–380 (2012) 1

[24] Sanz-Garcia, A., Fernandez-Ceniceros, J., Antonanzas-Torres, F., Pernia-Espinoza, A., Martinez-de Pison, F.J.: GA-PARSIMONY: A GA-SVR approach with feature selection and parameter optimization to obtain parsimonious solutions for predicting temperature settings in a continuous annealing furnace. Applied Soft Computing 35, 13–28 (2015) 1, 2.3

[25] Sanz-Garcia, A., Fernández-Ceniceros, J., Fernández-Martínez, R., Martínez-De-Pisón, F.J.: Methodology based on genetic optimisation to develop overall parsimony models for predicting temperature settings on annealing furnace. Ironmaking & Steelmaking 41(2), 87–98 (2014) 1

[26] Sanz-García, A., Fernández-Ceniceros, J., Antoñanzas-Torres, F., Martínez-de Pisón, F.J.: Parsimonious support vector machines modelling for set points in industrial processes based on genetic algorithm optimization. In: International Joint Conference SOCO13-CISIS13-ICEUTE13, Advances in Intelligent Systems and Computing, vol. 239, pp. 1–10. Springer International Publishing (2014) 2.3

[27] Shahriari, B., Swersky, K., Wang, Z., Adams, R.P., de Freitas, N.: Taking the human out of the loop: A review of bayesian optimization. Tech. rep., Universities of Harvard, Oxford, Toronto, and Google DeepMind (2015) 2.2

[28] Snoek, J., Larochelle, H., Adams, R.P.: Practical bayesian optimization of machine learning algorithms. In: Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 25, pp. 2951–2959. Curran Associates, Inc. (2012) 2.2

[29] Srinivas, N., Krause, A., Kakade, S.M., Seeger, M.W.: Gaussian process bandits without regret: An experimental design approach. CoRR abs/0912.3995 (2009) 2.2

[30] Thornton, C., Hutter, F., Hoos, H.H., Leyton-Brown, K.: Auto-weka: Combined selection and hyperparameter optimization of classification algorithms. In: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 847–855. KDD '13, ACM, New York, NY, USA (2013) 1

[31] Urraca, R., Sodupe-Ortega, E., Antonanzas, J., Antonanzas-Torres, F., de Pison, F.M.: Evaluation of a novel GA-based methodology for model structure selection: The GA-PARSIMONY. Neurocomputing 271(Supplement C), 9 – 17 (2018) 1, 2.3

[32] Urraca-Valle, R., Sanz-García, A., Fernández-Ceniceros, J., Sodupe-Ortega, E., de Pisón Ascacibar, F.J.M.: Improving hotel room demand forecasting with a hybrid GA-SVR methodology based on skewed data transformation, feature selection and parsimony tuning. In: Onieva, E., Santos, I., Osaba, E., Quintián, H., Corchado, E. (eds.) Hybrid Artificial Intelligent Systems - 10th International Conference, HAIS 2015, Bilbao, Spain, June 22-24, 2015, Proceedings. Lecture Notes in Computer Science, vol. 9121, pp. 632–643. Springer (2015) 1, 2.3

[33] Vieira, S.M., Mendonza, L.F., Farinha, G.J., Sousa, J.M.: Modified binary PSO for feature selection using SVM applied to mortality prediction of septic patients. Applied Soft Computing 13(8), 3494 – 3504 (2013) 1

[34] Winkler, S.M., Affenzeller, M., Kronberger, G., Kommenda, M., Wagner, S., Jacak, W., Stekel, H.: Analysis of selected evolutionary algorithms in feature selection and parameter optimization for data based tumor marker modeling. In: Moreno-Diaz, R.Z, R., Pichler, F., Quesada-Arencibia, A. (eds.) EUROCAST (1). Lecture Notes in Computer Science, vol. 6927, pp. 335–342. Springer (2011) 1

[35] Xue, B., Zhang, M., Browne, W.N.: Particle swarm optimisation for feature selection in classification: Novel initialisation and updating mechanisms. Applied Soft Computing 18(0), 261 – 276 (2014) 1

321 ## Figures

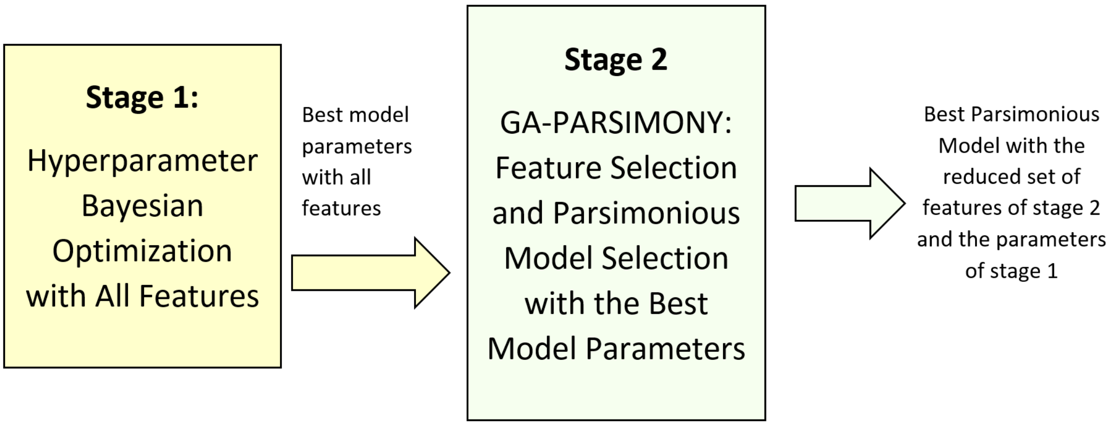

Figure 1: Descritpion of the Hybrid methodology that combines BO and GA-PARSIMONY.

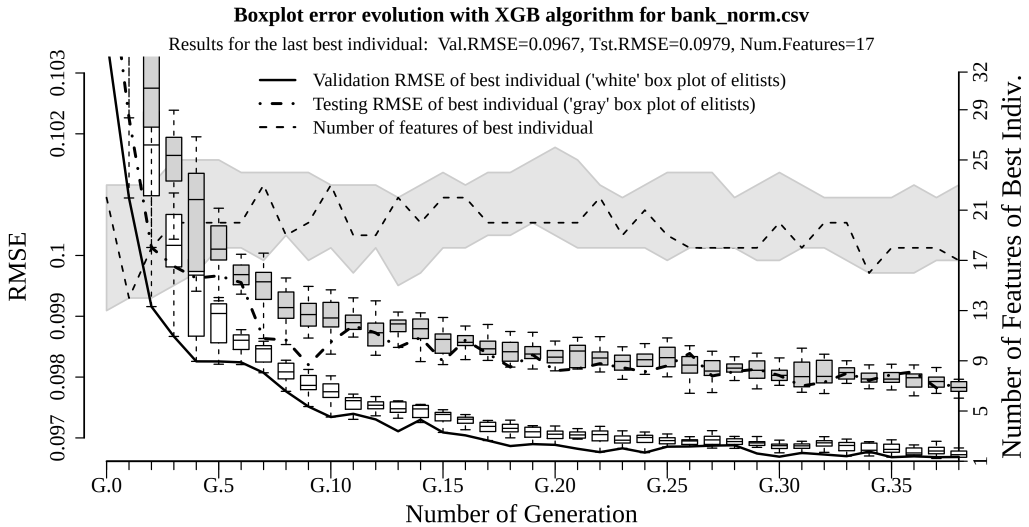

Figure 2: Evolution of elitist individuals in Bank database using GA-PARSIMONY for HO, FS, DT and PMS. White and gray box-plots represent $RMSE_{val}$ and $RMSE_{tst}$ evolution respectively. Discontinuous lines represent the best individual. The shaded area delimits the maximum and minimum N_{FS}.

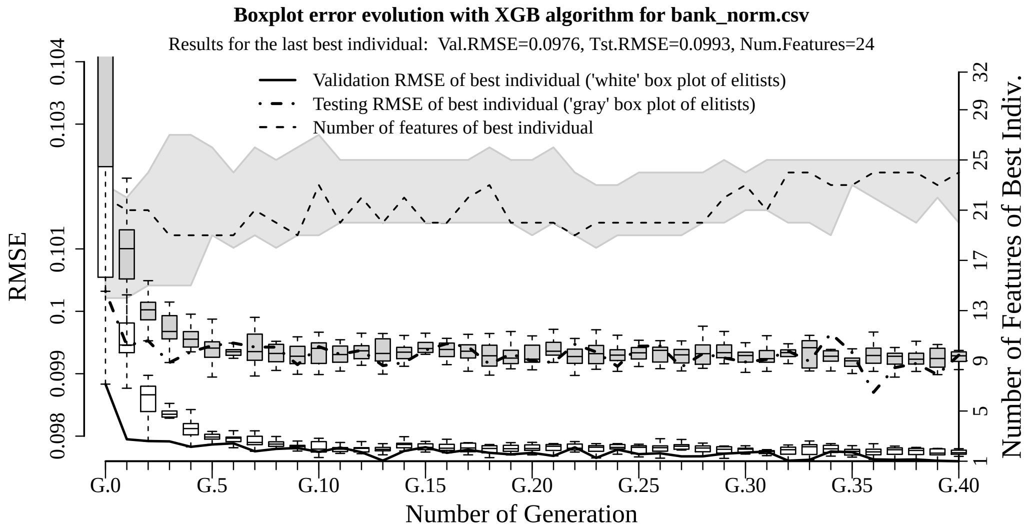

Figure 3: Evolution of elitist individuals in Bank database of Stage 2 of Hybrid methodology which uses GA-PARSIMONY with XGBoost parameters fixed to the best ones obtained with BO. White and gray box-plots represent $RMSE_{val}$ and $RMSE_{tst}$ evolution respectively. The shaded area delimits the maximum and minimum N_{FS}.

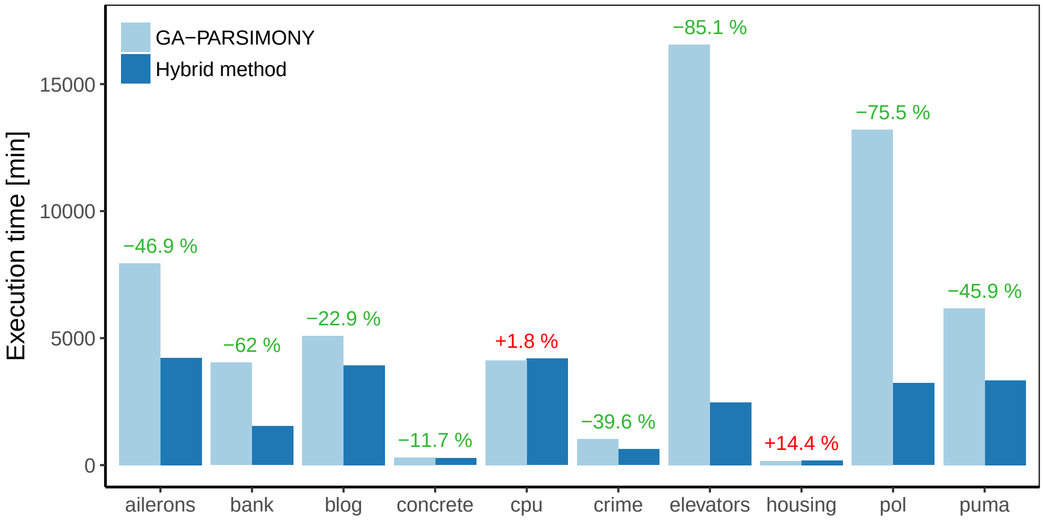

Figure 4: Execution times of the GA-PARSIMONY and the Hybrid methodology.

Tables

Table 1: Results obtained with the BO, GA-PARSIMONY and the Hybrid proposal. FT stands for the number of features of the best model, $RMSE_{tst}^{mean}$ is the mean testing error and $Time$ the elapsed time in minutes. Best results for each database are depicted in bold.

Database		*Bayesian Optim.*			*GA-PARSIMONY*				*Hybrid Method*		
Name	# Inst	#FT	Time	$RMSE_{tst}^{mean}$	#Gen	#FT	Time	$RMSE_{tst}^{mean}$	#FT	Time	$RMSE_{tst}^{mean}$
Ailerons	13750	40	295	0.0428	23	**13**	7949	**0.0425**	14	4221	**0.0425**
Bank	8192	32	104	0.0995	35	**18**	4036	**0.0980**	20	1533	0.0991
Blog	52397	276	1186	0.0155	13	**100**	5097	0.0148	108	3930	**0.0147**
Concrete	1030	8	152	0.0532	100	**7**	308	0.0521	8	272	**0.0519**
Cpu	8192	21	189	0.0232	20	**16**	4121	**0.0220**	**16**	4194	0.0231
Crime	2215	127	206	0.0612	100	**38**	1037	**0.0576**	40	626	**0.0576**
Elevators	16599	18	343	0.0322	39	**9**	16554	**0.0314**	12	2466	0.0319
Housing	506	13	136	0.0737	100	**10**	167	**0.0586**	55	191	0.0589
Pol	15000	26	176	0.0476	66	**16**	13203	**0.0400**	20	3231	0.0465
Puma	8192	32	209	0.0433	25	**4**	6168	0.0337	**4**	3337	**0.0336**

Table 2: Testing RMSE obtained with the three methodologies. Last column in Bayesian Optimization and the Hybrid method shows the p-value obtained with the Wilcoxon test when comparing each method against GA-PARSIMONY.

Database	GA-PARSIMONY		Bayesian Optim.			Hybrid Methodology		
Name	$RMSE_{tst}^{mean}$	$RMSE_{tst}^{sd}$	$RMSE_{tst}^{mean}$	$RMSE_{tst}^{sd}$	p-value	$RMSE_{tst}^{mean}$	$RMSE_{tst}^{sd}$	p-value
Ailerons	**0.0425**	0.042429	0.0428	0.000947	=(0.700)	**0.0425**	0.000784	=(1.000)
Bank	**0.0980**	0.097594	0.0995	0.001253	=(0.100)	0.0991	0.001149	=(0.200)
Blog	0.0148	0.014595	0.0155	0.010170	+(0.039)	**0.0147**	0.000994	=(1.000)
Concrete	0.0521	0.052261	0.0532	0.013800	=(0.100)	**0.0519**	0.013542	=(0.750)
Cpu	**0.0220**	0.021727	0.0232	0.002806	=(0.100)	0.0231	0.002863	=(0.100)
Crime	**0.0576**	0.058036	0.0612	0.004623	=(0.300)	**0.0576**	0.003234	=(0.834)
Elevators	**0.0314**	0.031355	0.0322	0.000641	=(0.100)	0.0319	0.000679	=(0.400)
Housing	**0.0586**	0.057918	0.0737	0.005727	+(0.000)	0.0589	0.005402	=(0.757)
Pol	**0.0400**	0.040358	0.0476	0.002647	+(0.008)	0.0465	0.001483	+(0.030)
Puma	0.0337	0.000420	0.0433	0.001411	+(0.008)	**0.0336**	0.000648	=(0.200)

Table 3: Summary of the stages of Hybrid method

Database	*Stage 1*			*Stage 2*				*Stage 2* vs *GA-PARSIMONY*
Name	#FT	Time	$RMSE_{tst}^{mean}$	#Gen	#FT	Time	$RMSE_{tst}^{mean}$	Diff. Time (%)
Ailerons	40	295	0.0428	14	14	3926	0.0420	3568 min. (50.61%)
Bank	32	104	0.0995	13	20	1429	0.0991	2607 min. (64.59%)
Blog	276	1186	0.0155	7	108	2744	0.0147	2353 min. (46.16%)
Concrete	8	152	0.0532	20	8	120	0.0519	188 min. (61.03%)
Cpu	21	189	0.0232	26	16	4005	0.0231	116 min. (02.81%)
Crime	127	206	0.0612	22	40	420	0.0576	617 min. (59.50%)
Elevators	18	343	0.0322	5	12	2123	0.0319	14431 min. (87.18%)
Housing	13	136	0.0737	16	9	55	0.0589	112 min. (67.07%)
Pol	26	176	0.0476	17	20	3055	0.0465	9972 min. (75.53%)
Puma	32	209	0.0433	13	4	3128	0.0336	3040 min. (49.29%)