New Hybrid Methodology Based on Particle Swarm Optimization with Genetic Algorithms to Improve the Search of Parsimonious Models in High-Dimensional Databases

17th International Conference on Hybrid Artificial Intelligent Systems, HAIS 2022 (September, 2022)

J. Divasón*, A. Pernia and F.J. Martinez-de-Pison

University of La Rioja, Spain

UNIVERSIDAD DE LA RIOJA

HAIS 2022

Index

Introduction

AutoML

- Nowadays, **there is a growing demand in auto machine learning (AutoML) tools** to automatize tedious tasks such as hyperparameter optimization (HO), model selection (MS), feature selection (FS) and feature generation (FG).
- Companies like *DataRobot*, *Strong Analytics*, *Mighty AI*, *Akkio*, *CloudZero* or *Unity Technologies*, among others, are currently providing services to automate a multitude of tasks in machine learning and artificial intelligence.
- In addition, new AutoML suites have emerged such as *Google Cloud AutoML*, *Microsoft Azure ML*, *Alteryx Intelligence Suite*, or *H2O AutoML*.
- Free software is also available, such as `Auto-WEKA`, `Auto-Sklearn 2.0`, `Hyperopt`, `TPOT`, or, for Deep Learning, `Auto-Keras` and `Auto-PyTorch`.

HAIS 2022

UNIVERSIDAD DE LA RIOJA

Introduction

Our Previous Works

- Our work over the years has focused on the development of machine learning models with small but high dimensional databases.
- The goal was to find accurate low complexity models by reducing the number of features and the internal complexity of the models.
- **The search for low complexity models (more parsimonious), among different accurate solutions, is usually an important strategy for finding models that will be robust against perturbations or noise. These kinds of models will be also easier to maintain and understand.**
- In previous HAIS and SOCO conferences, we have introduced two methodologies GA-PARSIMONY and PSO-PARSIMONY.

HAIS 2022

UNIVERSIDAD DE LA RIOJA

Introduction

Our Previous Works: GA-PARSIMONY

- GA-PARSIMONY optimizes with GA the Hyperparameter Optimization (HO) and the Feature Selection (FS) by executing a parsimonious model selection (PMS), **which is based on criteria that considers complexity and accuracy separately**.
- This methodology has been successfully applied in a variety of real applications such as steel industrial processes, hotel room booking forecasting, mechanical design, solar radiation forecasting, and demand energy prediction.
- However, it requires a high number of individuals and also several repetitions of the method to guarantee an optimal solution (because it is unstable). **It often converges to local minima instead of global minima**.
- In addition, it **sometimes generates twin individuals that do not contribute new information to the search process**.

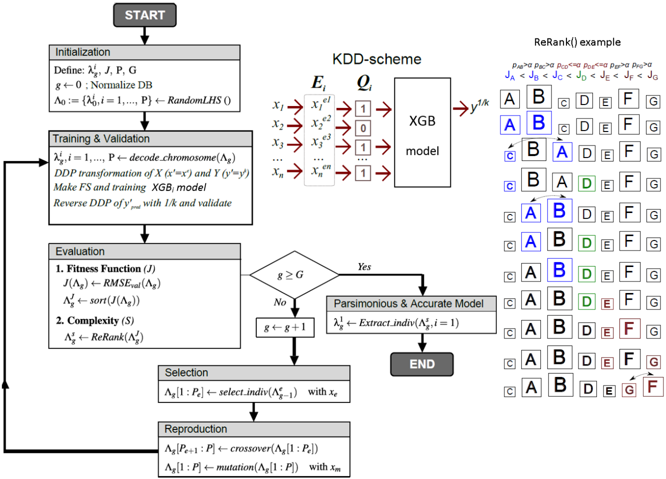
HAIS 2022

UNIVERSIDAD DE LA RIOJA

Introduction

Our Previous Works: PSO-PARSIMONY R and Python Packages

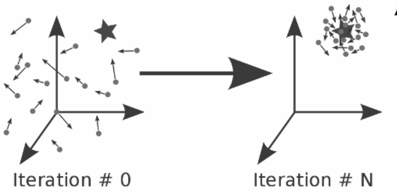
The *GAparsimony* packages for R and Python are available from official repositories (CRAN and pip)



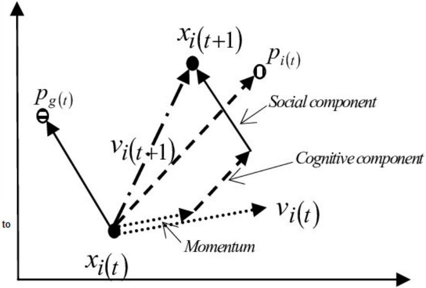
Introduction

Our Previous Works: PSO-PARSIMONY

- PSO-PARSIMONY methodology was developed to improve the results of GA-PARSIMONY by combining particle swarm optimization technique (PSO) and the parsimony criteria to obtain high-accuracy and low complexity models.
- The algorithm improved GA-PARSIMONY on small dimensional databases, but was stuck in local minima on large databases.



Tokhi, MO and Alam, MS (2009). Particle swarm optimisation algorithms and their application to controller design for flexible structure systems. IST Transactions of Control Engineering-Theory and Applications. 1 (3(9)), pp. 12-25



$$\vec{V}_i^{t+1} = \omega \vec{V}_i^t + \vec{r}_1 \varphi_1 \times \left(\vec{P}_{best,i}^t - \vec{X}_i^t\right) + \vec{r}_2 \varphi_2 \times \left(\vec{L}_{best,i}^t - \vec{X}_i^t\right) \quad (1)$$

$$\vec{X}_i^{t+1} = \vec{X}_i^t + \vec{V}_i^{t+1} \quad (2)$$

Index

HAIS 2022

UNIVERSIDAD DE LA RIOJA

PSO-PARSIMONY

How PSO works (I)

- The PSO algorithm works by having a population (called a swarm) of possible solutions (called particles).
- The position of a particle is simply a vector $X = (H, F)$ where H corresponds to the values of model's hyperparameters and F is a vector with values between 0 and 1 for selecting the input features.
- These particles are moved around in the search-space of the combinational problem according to simple formulas:

$$V_i^{t+1} = \omega V_i^t + r_1\varphi_1 \times \left(pbest_i^t - X_i^t\right) + r_2\varphi_2 \times \left(lbest_i^t - X_i^t\right) \tag{1}$$

$$X_i^{t+1} = X_i^t + V_i^{t+1} \tag{2}$$

where V_i^t and X_i^t denote the velocity and position of the i-th particle in iteration t, respectively.

HAIS 2022

UNIVERSIDAD DE LA RIOJA

PSO-PARSIMONY

How PSO works (and II)

- Such formulas just state that the movement of a particle is influenced by three components: its previous velocity, its own experience (its best position achieved so far, $pbest_i$) and also by the experience of other particles (the best position within a neighborhood, $lbest_i$).
- This permits particles to explore the search space based on their current momentum, each individual particle thinking (cognitive component) and the collaborative effect of the particles (cooperation component).
- More concretely, ω is the inertia weight used to control the displacement of the current velocity. φ_1 and φ_2 are positive constant parameters that balance the global exploration and local exploitation.
- r_1 and r_2 are uniformly distributed random variables within range $[0, 1]$ used to maintain the diversity of the swarm.

HAIS 2022

UNIVERSIDAD DE LA RIOJA

PSO-PARSIMONY

How PSO-PARSIMONY works

Our modified PSO includes a strategy where the best position of each particle (thus, also the best position of each neighborhood) **is computed considering not only the goodness-of-fit, but also the principle of parsimony**.

Algorithm 1 Pseudo-code of the modified PSO algorithm

Initialization of positions $\mathbf{X}^{\mathbf{0}}$ using a random and uniformly distributed Latin hypercube within the ranges of feasible values for each input parameter
Initialization of velocities according to $\mathbf{V}^{\mathbf{0}} = \frac{random_{LHS}(s,\ D)-\mathbf{X}^{\mathbf{0}}}{2}$
for $t = 1$ to T **do**
    Train each particle $\mathbf{X}_i$ and validate with CV
    Fitness evaluation and complexity evaluation of each particle
    Update $pbest_i$, $pbest_{p,i}$ and the $gbest$
    if early stopping is satisfied **then**
        return $gbest$
    end if
    Generation of new neighborhoods if $gbest$ did not improve
    Update each $lbest_i$
    Update positions and velocities according the formulas
    Mutation of features
    Limitation of velocities and out-of-range positions
end for
return gbest

HAIS 2022

UNIVERSIDAD DE LA RIOJA

PSO-PARSIMONY

PSO-PARSIMONY vs GA-PARSIMONY (I)

Experiments of PSO-PARSIMONY vs GA-PARSIMONY were conducted with 10 databases (results shows average of 5 runs):

Table 1: PSO-PARSIMONY vs GA-PARSIMONY with 10 databases (results are the average of 5 runs with each methodology and $tol = 10^{-3}$).

Database	*#rows*	*#feats*	$\overline{PSO_J}$	$\overline{GA_J}$	$\overline{PSO_{NFS}}$	$\overline{GA_{NFS}}$	$\overline{PSO_{time}}$	$\overline{GA_{time}}$	$\overline{PSO_{iters}}$	$\overline{GA_{iters}}$
strike	625	7	**0.83856**	0.86479	**1.8**	3.0	105.1	**16.3**	88.0	**39.6**
no2	500	8	**0.65608**	0.66007	6.0	6.0	59.8	**17.5**	102.8	**46.4**
concrete	1030	9	**0.28943**	0.29526	**7.4**	7.8	468.0	**160.9**	107.2	**41.6**
housing	506	14	**0.31261**	0.32559	11.0	**10.0**	215.8	**74.6**	104.0	**61.0**
bodyfat	252	15	**0.10709**	0.10806	3.4	**2.0**	97.3	**40.2**	128.2	**69.2**
cpu_act	8192	22	**0.12405**	0.12473	14.8	**13.4**	1241.7	**788.3**	84.6	**50.0**
bank	8192	33	**0.63420**	0.63792	23.6	**19.8**	1298.8	**629.1**	177.8	**101.6**
puma	8192	33	**0.18049**	0.18097	4.6	**4.2**	2188.6	**1028.4**	96.0	**51.2**
ailerons	13750	41	**0.38228**	0.38258	17.8	**10.4**	2663.2	**1144.7**	115.2	**65.6**
crime	2215	128	**0.59336**	0.59565	49.8	**19.8**	797.8	**410.5**	185.6	**143.6**

HAIS 2022

UNIVERSIDAD DE LA RIOJA

PSO-PARSIMONY

PSO-PARSIMONY vs GA-PARSIMONY: Conclusions (and II)

- PSO-PARSIMONY **always improved accuracy with respect to GA**.
- In databases with a low number of features, the difference with respect to parsimony was usually small (GA found solutions about 10% simpler).
- However, datasets with a larger number of features caused trouble for PSO, **which found better solutions than GA but with twice as many features**.
- Moreover, **GA required much less computational effort**: approximately, GA was three times faster and needed halving the iterations.
- **This showed that PSO-PARSIMONY is a good alternative to GA-PARSIMONY if the number of features is relatively small but fails to obtain parsimonious solutions in high dimensional datasets**.

HAIS 2022

UNIVERSIDAD DE LA RIOJA

Index

HAIS 2022

UNIVERSIDAD DE LA RIOJA

Objective of this Work

The main goal was **to promote the parsimonious behavior of GA in PSO**. To increase parsimony in PSO, two new variants of the algorithm were proposed:

- First, **PSO is combined with an aggressive mutation strategy** to foster parsimonious models.
- Second, an hybrid model between PSO and genetic algorithms is proposed, **in which the particles with worse fitness are replaced in each iteration by new ones generated from typical genetic algorithm operations**: selection, crossover and mutation.
- Accuracy and complexity of the new proposals are tested with public databases of different sizes and compared with GA-PARSIMONY.
- They have been implemented in Python and are available at https://github.com/jodivaso/Hybrid-PSOGAParsimony.

Index

New Proposals

1. PSO with a New Mutation (I)

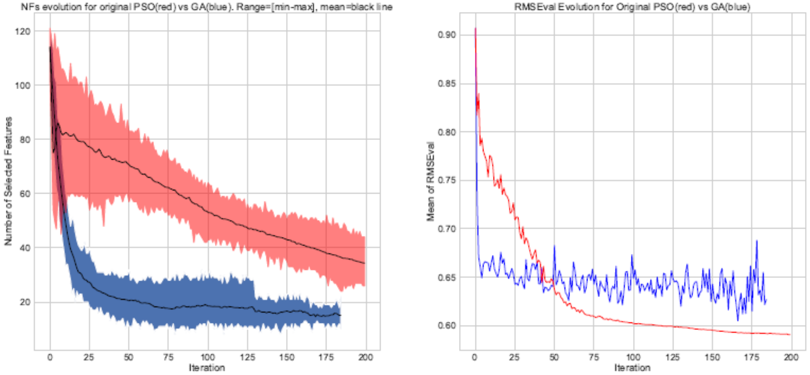
- PSO-PARSIMONY already includes a mutation rate which was set to $1/D$ by default, where D is the dimensionality of the problem.
- GA **has a much more aggressive strategy excluding a** 9% **of the features in each iteration**.
- In contrast, only $1/2D$ of the features are excluded in each step with PSO, **which are very few if the dataset has many variables**. This explains why the PSO algorithm performs worse in terms of parsimony with high-dimensional databases.
- To solve this problem PSO algorithm was modified to include the mutation step as is done in GA.

HAIS 2022

UNIVERSIDAD DE LA RIOJA

New Proposals

1. PSO with a New Mutation (and II)



(a) Features evolution in GA and PSO (b) Evolution of RMSE in GA and PSO

Fig. 1: PSO-PARSIMONY (red) vs. GA-PARSIMONY (blue) in terms of number of selected features and accuracy with the crime dataset.

New Proposals

2. Hybrid method: PSO with crossover and mutation (I)

- To further encourage parsimony in PSO and make its behavior closer to GA, especially in the first iterations, **a crossover phase is added just after calculating the local bests of the neighborhoods**. The crossover function was implemented by using heuristic blending for hyperparameters and random swapping for features.
- To perform this crossover, a selection phase is also added at that point, with a **Michalewicz nonlinear-rank selection**.
- The selection of other individuals in addition to the best ones **maintains the diversity of the population and prevents premature convergence**. Furthermore, the **best individuals are more likely to be selected for crossover**. Thus, they are selected for breeding more times to foster good offspring.

HAIS 2022

UNIVERSIDAD DE LA RIOJA

New Proposals

2. Hybrid method: PSO with crossover and mutation (II)

- The way of replacing the particles differs from the typical GA crossover. In this case, **the new particles created from the crossover replace the worst particles** (those with the worst fitness value) that appeared in the population.
- Parameter `pcrossover` fixes the percentage of worst individuals to be substituted from crossover. This parameter can be either a constant (such a percentage of particles is substituted in all iterations) or **a vector to indicate a different percentage in each iteration**.
- Once the crossover step is done, PSO algorithm requires updating the positions and velocities according the formulas. **This step is only applied to the particles that have not been substituted by the crossover**.
- Mutation phase is also modified to include the changes proposed in the previous subsection: a more aggressive mutation strategy to encourage parsimony.
- The rest of steps of the PSO-PARSIMONY algorithm are preserved.

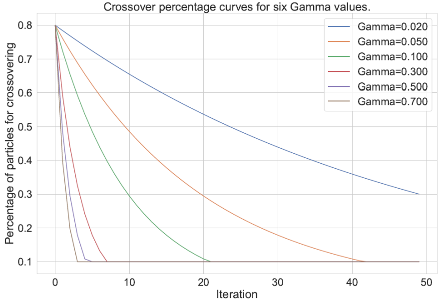
HAIS 2022

UNIVERSIDAD DE LA RIOJA

New Proposals

2. Hybrid method: PSO with crossover and mutation (and III)

- For the hybrid method, we suggested the following equation $\%particles = max(0.80 \cdot e^{(-\Gamma \cdot iter)}, 0.10)$ that was defined **to calculate the percentage of particles to be substituted by crossover in each iteration** *iter*.
- Figure shows six curves obtained with different Γ values. In the first iterations, the hybrid method performs the substitution by crossing a high percentage of particles. As the optimization process progresses, the number of substituted particles is reduced exponentially until it ends up fixed at a percentage of 10%. Thus, **the hybrid method begins by facilitating the search for parsimonious models using GA-based mechanisms and ends up using more PSO optimization**.



HAIS 2022

UNIVERSIDAD DE LA RIOJA

Index

HAIS 2022

UNIVERSIDAD DE LA RIOJA

Experiments
Settings

- Databases with a high number of features were selected to test the capacity of the proposed methodologies to find accurate and parsimonious models.
- Experiments compared the PSO-PARSIMONY with the new mutation (New-PSO), the new hybrid HYB-PARSIMONY (HYB), GA-PARSIMONY (GA) and the previous PSO-PARSIMONY method (Old-PSO).
- All experiments were similar to previous works with a population size of $P = 40$, $tol = 0.001$, a maximum number of generations of $G = 200$, and an early stopping of 35.
- Experiments were implemented in 9 separately 24-core servers from the Beronia Cluster at the University of La Rioja.

HAIS 2022

UNIVERSIDAD DE LA RIOJA

Experiments

NEW PSO-PARSIMONY vs HYBRID-PARSIMONY (*crime* database)

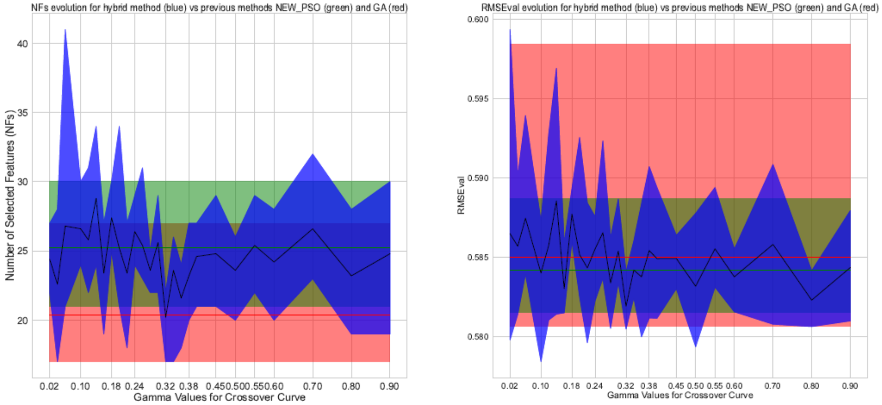
The hybrid method with $\Gamma = 0.10$ **obtained the best model reducing J to 0.57844 versus the previous best model achieved with GA ($J = 0.58070$).** However, the improvement in J involved the selection of 24 features (5 more) versus 19 in GA. On the other hand, **the hybrid method with $\Gamma = 0.04$ obtained the most parsimonious model with only 17 features and an error of $J = 0.58142$, slightly higher than the J of GA**. With respect to the mean values obtained from the five runs of each algorithm, it is observed **that the hybrid method with $\Gamma = 0.32$ obtained the best mean values of J and N_{FS}**.

Table 2: Hybrid with different Γ vs previous methods for *crime* database.

Method	Γ	J_{best}	N_{FSbest}	$\overline{J}$	$\overline{N_{FS}}$	$\overline{iters}$	$\overline{time}$
GA	0.00	0.58070	19	0.58503	20.4	146.8	86.2
OLD PSO	0.00	0.58333	29	0.58773	33.0	200.0	121.0
NEW PSO	0.00	0.58155	26	0.58419	25.2	362.8	211.9
HYB	0.02	0.57981	22	0.58650	24.4	229.6	132.0
HYB	0.04	0.58142	**17**	0.58571	22.6	200.8	118.8
HYB	0.06	0.58397	23	0.58747	26.8	206.6	119.1
HYB	0.10	**0.57844**	24	0.58402	26.6	200.4	115.6
HYB	0.12	0.58106	24	0.58576	25.8	184.4	97.6
HYB	0.14	0.58143	24	0.58856	28.8	190.0	109.9
HYB	0.16	0.58151	19	0.58304	23.4	228.8	121.5
HYB	0.18	0.58603	26	0.58773	27.4	148.6	85.9
HYB	0.20	0.58251	23	0.58517	25.2	185.6	111.0
HYB	0.22	0.57964	23	0.58434	23.4	241.4	138.9
HYB	0.24	0.58229	25	0.58554	26.4	167.4	98.0
HYB	0.26	0.58368	23	0.58656	25.4	176.0	101.2
HYB	0.28	0.58054	24	0.58340	23.6	235.6	135.0
HYB	0.30	0.58343	29	0.58540	25.6	**143.8**	**82.7**
HYB	0.32	0.58050	22	**0.58193**	**20.2**	242.2	139.1
HYB	0.34	0.58247	**17**	0.58421	23.6	233.2	123.5
HYB	0.36	0.58001	23	0.58378	21.6	221.6	127.4
HYB	0.38	0.58119	20	0.58544	23.2	197.2	117.9
HYB	0.40	0.58117	27	0.58493	24.6	209.8	120.6
HYB	0.45	0.58304	22	0.58494	24.8	176.8	101.9
HYB	0.50	0.57938	24	0.58319	23.6	213.4	123.0
HYB	0.55	0.58314	24	0.58555	25.4	193.8	111.5
HYB	0.60	0.58158	22	0.58378	24.2	218.2	115.8
HYB	0.70	0.58080	24	0.58582	26.6	195.8	116.5
HYB	0.80	0.58065	19	0.58232	23.2	215.0	123.7
HYB	0.90	0.58101	25	0.58437	24.8	187.4	108.5

Experiments

NEW PSO-PARSIMONY vs HYBRID-PARSIMONY (*crime* database)



HAIS 2022

UNIVERSIDAD DE LA RIOJA

Experiments

NEW PSO-PARSIMONY vs HYBRID-PARSIMONY (*crime* database)

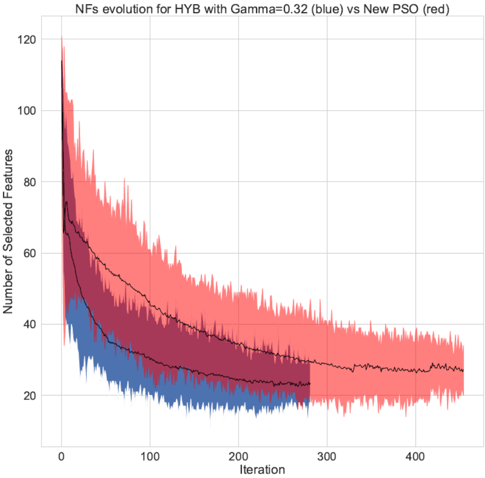


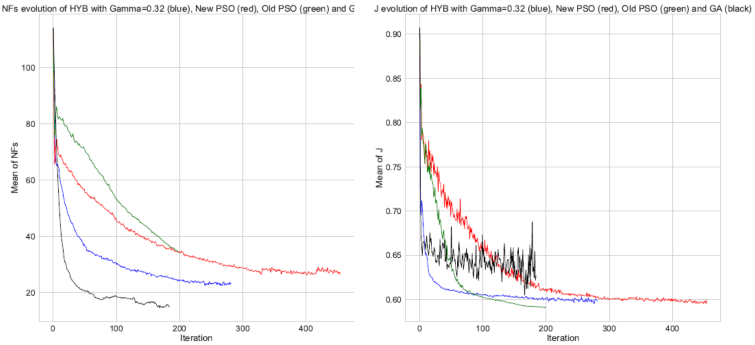
Fig. 4: Comparison of N_{FS} evolution between the Hybrid method (blue) (with $\Gamma = 0.32$) and the new PSO-PARSIMONY (red).

HAIS 2022

UNIVERSIDAD DE LA RIOJA

Experiments

NEW PSO-PARSIMONY vs HYBRID-PARSIMONY (*crime* database)



(a) $\overline{N_{FS}}$ evolution

(b) $\overline{J}$ evolution

Fig. 5: Comparison with *crime* dataset of $\overline{N_{FS}}$ and $\overline{J}$ between the four methods: the Hybrid method (blue) (with $\Gamma = 0.32$), the new PSO-PARSIMONY (red), the old PSO-PARSIMONY (green) and GA-PARSIMONY (black).

Experiments

NEW PSO-PARSIMONY vs HYBRID-PARSIMONY (7 Databases)

- Similar results can be observed with other high-dimensional databases.
- Tables 3 and 4, show respectively the average results and the best model obtained with the Hybrid Method and the New-PSO.
- In almost all databases, the hybrid method obtained more accurate models with less complexity, although it was necessary to find a suitable Γ value.

Experiments

NEW PSO-PARSIMONY vs HYBRID-PARSIMONY (7 Databases)

Table 3: NEW PSO-PARSIMONY vs HYBRID-PARSIMONY with a population size of $P = 40$ and $tol = 0.001$ (results are the average of the 5 runs).

Dataset	*#rows*	*#feats*	Γ	$\overline{PSO_J}$	$\overline{HYB_J}$	$\overline{PSO_{N_{FS}}}$	$\overline{HYB_{N_{FS}}}$	$\overline{PSO_{time}}$	$\overline{HYB_{time}}$
slice	5000	379	0.34	0.0238	**0.0231**	146.8	**132.2**	819.4	**609.0**
blog	4999	277	0.70	0.4087	**0.3983**	127.6	**113.8**	1117.5	**1051.6**
crime	2215	128	0.32	0.5842	**0.5819**	25.2	**20.2**	211.9	**139.1**
tecator	240	125	0.50	0.0331	0.0331	55.0	**48.6**	11.9	**8.7**
ailerons	5000	41	0.70	0.3947	**0.3934**	10.6	**10.2**	473.4	**466.1**
bank	8192	33	0.50	0.6514	**0.6511**	21.4	21.4	2146.4	**1536.6**
puma	8192	33	0.50	0.1817	0.1817	4.0	4.0	1063.8	**933.2**

Table 4: Best individual obtained with PSO-PARSIMONY vs HYBRID-PARSIMONY using a population size of $P = 40$ and $tol = 0.001$.

Dataset	Γ	PSO_J	HYB_J	$PSO_{J_{TST}}$	$HYB_{J_{TST}}$	$PSO_{N_{FS}}$	$HYB_{N_{FS}}$	PSO_{time}	HYB_{time}
slice	0.70	0.0228	**0.0218**	**0.0012**	0.0017	124	**112**	1050.1	**627.5**
blog	0.38	0.3948	**0.3879**	0.2523	**0.2023**	**115**	129	1304.0	**1277.4**
crime	0.10	0.5815	**0.5784**	0.5021	**0.4780**	26	**24**	263.5	**138.5**
tecator	0.38	0.0328	**0.0327**	0.0207	**0.0206**	**48**	51	16.3	**10.7**
ailerons	0.15	0.3935	**0.3922**	**0.3675**	0.3698	13	**10**	**484.2**	494.3
bank	0.70	0.6510	**0.6507**	**0.5839**	0.5865	22	**21**	2428.5	**1675.0**
puma	0.38	0.1817	0.1817	0.1776	0.1776	4	4	1191.8	**712.9**

Index

HAIS 2022

UNIVERSIDAD DE LA RIOJA

Conclusions (I)

- The present work includes two new proposals to improve our previous PSO-PARSIMONY methodology for the simultaneous search of the best model hyperparameters and input features, with a balance between accuracy and complexity.

HAIS 2022

UNIVERSIDAD DE LA RIOJA

Conclusions (I)

- The present work includes two new proposals to improve our previous PSO-PARSIMONY methodology for the simultaneous search of the best model hyperparameters and input features, with a balance between accuracy and complexity.
- The main novelty relies on the hybrid model where the optimization is based on the PSO formulas, but the common genetic operations of selection, crossover and mutation are included to replace the worst particles.

Conclusions (I)

- The present work includes two new proposals to improve our previous PSO-PARSIMONY methodology for the simultaneous search of the best model hyperparameters and input features, with a balance between accuracy and complexity.
- The main novelty relies on the hybrid model where the optimization is based on the PSO formulas, but the common genetic operations of selection, crossover and mutation are included to replace the worst particles.
- The percentage of variables to be substituted in each iteration is customized with a function that depends on a Γ parameter. This function promotes parsimony in the first iterations (a high percentage of particles is substituted), but in further iterations the percentage is decreased.

Conclusions (and II)

- This differs from other hybrid methods where the crossover is applied between each particle's individual best positions or other approaches where the worst particles are also substituted by new ones, but at extreme positions.

Conclusions (and II)

- This differs from other hybrid methods where the crossover is applied between each particle's individual best positions or other approaches where the worst particles are also substituted by new ones, but at extreme positions.
- Experiments show that, in general and once the appropriate Gamma is fixed, the HYB-PARSIMONY methodology allows one to obtain better, more parsimonious and more robust models compared to our previous PSO-based methodology and the PSO with mutation. The computational effort is also reduced, since it requires less time.

HAIS 2022

UNIVERSIDAD DE LA RIOJA

Conclusions (and II)

- This differs from other hybrid methods where the crossover is applied between each particle's individual best positions or other approaches where the worst particles are also substituted by new ones, but at extreme positions.
- Experiments show that, in general and once the appropriate Gamma is fixed, the HYB-PARSIMONY methodology allows one to obtain better, more parsimonious and more robust models compared to our previous PSO-based methodology and the PSO with mutation. The computational effort is also reduced, since it requires less time.
- Although it is a promising method, further research is required to provide an explicit formula that fixes the appropriate Γ value for each dataset, for instance, depending on the number of instances and features or by means of adaptive strategies.

HAIS 2022

UNIVERSIDAD DE LA RIOJA

References

- Ceniceros, J.F., Sanz-Garcia, A., Pernia-Espinoza, A., Martinez-de Pison, F.J.: PSO-PARSIMONY: A new methodology for searching for accurate and parsimonious models with particle swarm optimization. application for predicting the force-displacement curve in t-stub steel connections. In: Sanjurjo González, H., Pastor López, I., García Bringas, P., Quintián, H., Corchado, E. (eds.) Hybrid Artificial Intelligent Systems. pp. 15 – 26. Springer, Cham (2021).
- Martinez-de Pison, F.J., Ferreiro, J., Fraile, E., Pernia-Espinoza, A.: A comparative study of six model complexity metrics to search for parsimonious models with GAparsimony R Package. Neurocomputing 452, 317 – 332 (2021).
- 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).

HAIS 2022

UNIVERSIDAD DE LA RIOJA

THANK YOU VERY MUCH

FOR YOUR ATTENTION

UNIVERSIDAD
DE LA RIOJA

Questions?

HAIS 2022

UNIVERSIDAD
DE LA RIOJA