Integration of Stacking Case-Based Reasoning with a Multi-Agent System Applied to Regression Problems

Daniel SOTO FORERO[1][a], Marie-Laure BETBEDER[1][b] and Julien HENRIET[1][c]

[1] *Université Marie et Louis Pasteur, FEMTO-ST, DISC, 16 route de Gray, Besançon, FRANCE*

{*daniel.soto_forero, marie-laure.betbeder, julien.henriet*}*@univ-fcomte.fr*

Keywords: Case-based reasoning, Intelligent tutoring system, Multi-Agent Systems, Stacking model, Bayesian learning

Abstract: This paper presents an integration model between CBR (Case Based Reasoning) and a MAS (Multi-Agent System) that improves our previously proposed model called ESCBR (Ensemble Case Based Reasoning). The integration model is designed to find approximate solutions to generic regression problems of one or more dimensions. The integration pretends to explore and exploit the solutions space better by using an individual cognitive process with social interactions and Bayesian reasoning. The proposed multi-agent system is composed of several heterogeneous agents pursuing a common goal, but in different ways. Each agent can choose an action from three possibilities and interact with other agents randomly. Bayesian learning allows each agent to learn using the information obtained from past iterations and to adapt its behavior according to the characteristics of the evaluated problem. The model has been evaluated in comparison with nine other regression models on eleven different regression databases extracted from the UCI site. The comparison between the models has been carried out with RMSE (Root Mean Squared Error) and MAE (Median Absolute Error) metrics. The results show that the proposed model obtains good results, globally among the three best models and effectively improves on the ESCBR base model.

1 Introduction

Our previous work with case based reasoning called ESCBR (Ensemble Stacking Case Based Reasoning) is a model based on the case based reasoning and a stacking ensemble method to predict values in regression problems. This model modifies the basic case based reasoning cycle by introducing learning steps, real-time evaluation and several algorithms to find neighbors and generate solutions (Soto-Forero et al., 2024).

Some models can be improved with multi-agent systems because these are decentralized and consist of multiple entities, called agents, that have the ability to perform specific actions and react to the environment according to the partial information they obtain. They can also collaborate with each other and cooperate to achieve a specific goal.

From the information they perceive and exchange, agents can learn autonomously and achieve their goals efficiently (Kamali et al., 2023). Multi-agent systems can be executed in parallel given the independence of the agents. They are robust to problems that present uncertainty (Didden et al., 2023).

Bayesian reasoning is a hypothesis about the functioning of the human brain and its relationship with the environment. The principle is that past experiences allow inferring future states, which allows action and decision-making (Hipólito and Kirchhoff, 2023), and also allows inferring information and learning from incomplete information (Zhang et al., 2023).

This paper proposes a model that integrates an ensemble case-based reasoning model with multi-agent systems and cognitive Bayesian reasoning. The objective is to improve the processes of neighbor search and solution generation in ESCBR by making use of inter-agent information exchange and reinforcement learning, thus obtaining emergent effects in the global behavior of the algorithm, which can lead to improvement in the predictions of the ESCBR algorithm (Soto-Forero et al., 2024).

[a] https://orcid.org/0000000307534673

[b] https://orcid.org/0000000281034098

[c] https://orcid.org/0000000276714574

The contributions of this paper are:

- Improvement of exploration and generation phases of the ESCBR algorithm with the multi-agent system integration.
- A multi-agent system with cognitive agents and independent stochastic behavior according to three predefined actions.
- Addition of a Bayesian learning process in the search for similar problems and generation of solutions.

2 Related Works

ESCBR is a model using CBR as a base to obtain competitive results for ten popular databases in regression problems (Soto-Forero et al., 2024). The algorithm runs independent processes to obtain an estimated answer to a given problem. Fundamentally, the process relies on the concept of analogy between problems and analogy between solutions to both generate and evaluate the proposed solutions.

Multi-agent systems have the ability to evolve and improve using the reinforcement learning strategy, as seen in the work of (Li et al., 2023b), Using a multi-agent evolutionary system with a graph neural network, machining parameters are optimized to boost quality and production efficiency. The method improves accuracy between 13% and 25% over other optimization algorithms. Multi-agent systems can be also developed by imitating natural process as (Kamali et al., 2023), whose objective is to solve multi-objective optimization in dynamic environments using an artificial immune–system metaphor, the method enables agents to react to environmental changes and generate diverse, adaptive solutions. As shown in prior work, multi-agent systems can adapt, generate innovative solutions, and even learn collectively. Also in the work of (Cinkusz and Chudziak, 2025) authors show how cognitive multi-agent systems can increase precision and intelligence, in this case applied to a problem of manage software projects. For example, (Rezaei et al., 2022) presents a biased inferential social-learning model based on Bayesian inference that combines local and global observations for improved decision-making. The algorithm's behavior is demonstrated through multiple simulations. These works highlight the potential of multi-agent algorithms to explore, generate, and refine solutions over time through interaction, cooperation, emergent behavior, and feedback. Their flexibility allows integration with other paradigms to enhance internal reasoning and solution-space exploration. In this work, we introduce a Bayesian reasoning approach to further improve overall solution quality.

The idea that human reasoning is fundamentally Bayesian has been widely explored in cognitive sciences, as it unifies multiple information sources and uses past experience to infer future states. Experimental evidence shows that human mental representations align with Bayesian prediction. Bayesian reasoning also supports efficient perception, flexibility, and top-down integration of prior knowledge (Hipólito and Kirchhoff, 2023). In (Kim et al., 2019), the authors model how people interpret data using a Bayesian cognitive framework in which beliefs are iteratively updated. They compare human judgments across three studies with Bayesian predictions, showing that the model retains notable predictive power even under uncertainty and seemingly rational behavior. Bayesian inference has also been used in multi-agent systems to facilitate learning and complement the cognitive process. In Rezaei *et al.* (Rezaei et al., 2022) a Bayesian inference model is proposed in which agents estimate the world's state to make better decisions. Agents begin with initial beliefs and, through social interaction, consensus, and new information, converge toward the truth. Simulations show rapid adaptation and recovery from false observations, demonstrating that Bayesian inference combined with social cooperation improves decision-making in complex, dynamic, and uncertain environments. Ristic *et al.*(Ristic et al., 2020) presents a Bayesian review for decision-making under uncertainty, introducing quantitative autonomous models, rules for combining incomplete information, and related decision tasks. It presents Bayesian probability as a strong framework for modeling both known and unknown factors, demonstrated through a classification problem. The work of Zhang *et al.* (Zhang et al., 2023) proposes a Bayesian learning algorithm to find optimal set-function values in both low and high dimensions.

In the work of Cicirello *et al.* (Cicirello and Giunta, 2022), The goal is to optimize a generic engineering system by quantifying uncertainty in interval variables using Bayesian Optimization (BO). Posterior probabilities are obtained by updating priors through deterministic simulations. This approach greatly reduces the number of simulations needed for accurate results by balancing exploration and exploitation through interval-based Bayesian optimization. Also, BO is applied in the work of Petit *et al.* (Petit et al., 2020) focused on learning hyperparameters for a robotic arm, improving task efficiency using past experiences. A model using Bayesian inference with agent-based models is explained in Dyer *et al.*(Dyer et al., 2024), where a Bayesian black-box model is compared with a neural network black-box model for parameter inference in economic simulation problems. Presscott *et al.* (Prescott et al., 2024), uses Bayesian inference to estimate parameters of complex stochastic models with intractable likelihoods. Likelihood-free Bayesian inference is combined with an adaptive multi-fidelity algorithm to estimate probability distributions in stochastic simulations. Bayesian inference makes it possible to work with data with a high level of uncertainty, which is one of the reasons why Nikpour and Aamodt (Nikpour and Aamodt, 2021) combine Bayesian inference with case-based reasoning for an oil well drilling problem, a real-world, uncertain domain. These studies show that Bayesian reasoning balances exploration and exploitation while achieving high performance, accuracy, and adaptability from limited, uncertain data. Integrating it with multi-agent systems further enhances inference, learning, and collaborative correction.

3 Proposed Model

The ESCBR-MAS (Ensemble Stacking Case-Based Reasoning with Multi-Agent Systems) algorithm is based on the generic CBR paradigm and incorporates various neighbor search (Retrieve) and solution generation algorithms (Reuse and Revise) . These are integrated using a variation of the stacking model in two iterative stages, executed by agents that independently operate the algorithms defined in the revision and reuse containers. The agents have a local memory that permits them to adjust and develop their search for neighbors, generating solutions during each iteration. Each individual agent has a corresponding behavior that enables the exchange of information with other agents. The decision process and stages of the agents within the broader system are shown in Figure 1.

The overall model is composed of two cycles that aim to refine the search for similar problems (Stage 0) and the generation of solutions (Stage 1). Each cycle is executed according to the number of iterations defined as parameters. The first cycle consists of the stages retrieve, reuse, revise, renovate, and retain. The second cycle consists of the stages retrieve, reuse, revise, reconfigure, and exchange, as shown in Figure 1.

ESCBR-MAS is the two stages of ESCBR (Ensemble Stacking Case-Based Reasoning) stacking integrated with a multi-agent system (Soto-Forero et al., 2024). Each agent conducts a problem neighbor search algorithm in the first stage and generates a solution in the second stage by referencing the list of solutions obtained in the first stage. In each iteration, agents can select from three programmed behaviors: searching for problems neighbors, generating solutions, or exchanging information with another agent. Asynchronous execution is employed for these behaviors, while neighbor list creation and selection occur synchronously to unify the global algorithm.

The retrieve, reuse, revise, and retain stages remain as in the conventional CBR model. The proposed algorithm adds three new stages: reconfigure, where agents update local parameters to improve the next solution; exchange, where agents share information to enhance search and generation; and renovate, which updates global algorithm parameters. Figure 2 shows the full flow, starting with the creation of n agents. During initialization, agents randomly select retrieve and reuse algorithms and initialize Bayesian vectors, working on the same dataset. Agents execute the retrieve algorithm in parallel, extract solutions from similar problems, and generate new solutions. Solutions are evaluated with a minimization objective function, where the lowest value is best. In subsequent iterations, agents update Bayesian vectors based on objective results and randomly choose one of three actions: change both retrieve and reuse algorithms, change only the reuse algorithm, or exchange information with another agent. This randomness allows agents to refine candidate solutions.

The complete variables and parameters for the proposed algorithm are shown in Table 1. The multi-agent system is composed of a variable number of agents, all homogeneous but with independent and stochastic internal cognitive processes.

3.1 Algorithms

In the first stacking stage, each agent selects an algorithm to find problems similar to the new one. Options include KNN, weighted KNN, K-Means, GMM, and FuzzyC-Means. KNN identifies nearby instances in the feature space, with weighted KNN allowing dimension-specific weighting (Saxena et al., 2024). K-Means clusters points by minimizing distances to cluster centers, updating centers as the mean of assigned points (Li et al., 2023a). GMM models data as a weighted sum of K Gaussian components

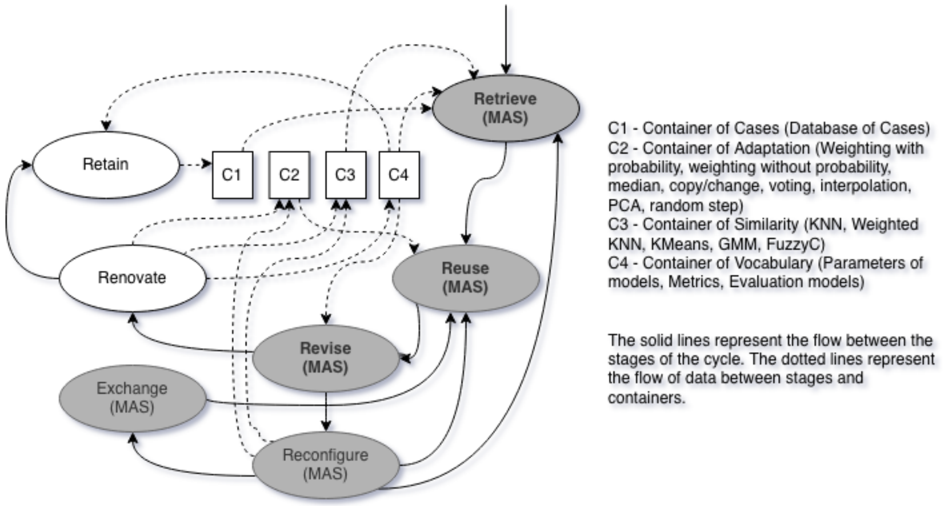

Figure 1: Two Cycles of the proposed ESCBR-MAS

Table 1: Variables and parameters of the proposed model (Type: p - parameter, v - variable, f - function)

ID	Type	Description	Domain
it	p	Number of iterations	$\mathbb{N}, it > 0$
np	p	Number of agents	$\mathbb{N}, np > 2$
nl	p	Maximum number of local neighbors	$\mathbb{N}, nl > 0$
ng	p	Number of global neighbors	$\mathbb{N}, ng > 2$
n	v	Dimension of problem space	$\mathbb{N}, n > 0$
m	v	Dimension of solution space	$\mathbb{N}, m > 0$
p	v	Problem description	$\mathbb{R}^n$
s	v	Solution description	$\mathbb{R}^m$
p_w	v	New problem description	$\mathbb{R}^n$
s_w	v	New solution description	$\mathbb{R}^m$
$d(x_1, x_2)$	f	Distance function between x_1 and x_2	$\mathbb{R}$
$rn(x, y)$	f	Random value with Normal distribution x mean, y standard deviation	$\mathbb{R}_+$
$rnp(x)$	f	Discrete random value, x discrete vector of probabilities	$\mathbb{Z}$
$f_s(p^n, s^m)$	f	Solutions evaluation	$\mathbb{R}$

for regression, classification, or clustering (Rakesh et al., 2023). FuzzyC-Means extends clustering by using fuzzy set theory to express membership uncertainty. (Jiao et al., 2022).

In the second stacking stage, agents select from weighting with probability, weighting without probability, median, copy/change, voting, interpolation, PCA, or random step. Weighting with probability copies solution information based on higher probabilities, while weighting without probability copies randomly. Median takes the median for each dimension. Copy/change combines parts of two random solutions. Voting selects the most frequent values. Interpolation uses values from a random interpolation function. PCA maps problem descriptions to solution space. Random step modifies a single dimension of a solution by a small random amount.

3.2 Agent Structure

The structure of the agents is similar for all, but each agent performs an individual cognitive process that allows it to obtain an independent behavior that is different from all the others. Figure 3 shows the actions and variables necessary to perform the whole process.

There are three possible actions: Retrieve and reuse, Reuse, and Exchange, and eight variables: neighbors number (how many similar problems to search), nearest neighbor list (agents for information exchange), retrieve algorithm (algorithm to find similar problems), reuse algorithm (algorithm to generate a candidate solution), solution (generated solution description), solution evaluation (quality via the objective function, Eqs. 1–2), Bayesian vector for retrieve algorithms, and Bayesian vector for reuse algorithms

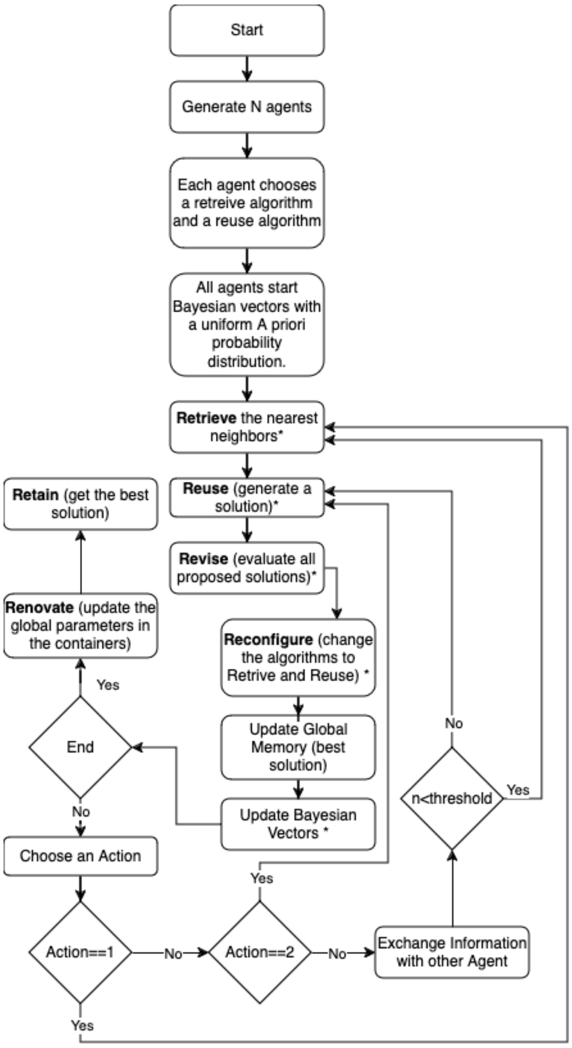

Figure 2: Flow of Stacking CBR (* is a task performed by each agent)

(probabilities for each algorithm).

$$min\ (f_s(p_w^n, s_w^m)) = min\left(\sum_{i=1}^{ng} \frac{d(s_w^m, s_i^t)}{d(p_w^n, p_i^n)^2}\right) \quad (1)$$

$$s_i^t = s_i^m + rn(0, d(p_w^n, p_i^n)) \quad (2)$$

3.3 Agent Learning

Each agent has a Bayesian probability vector for retrieve (C3) and reuse (C2) phases, initially uniform. These vectors are updated each iteration using the agent's best results as the likelihood. Algorithm choices are then guided by these updated probabilities, enabling the agent to learn from experience and favor algorithms most likely to yield optimal

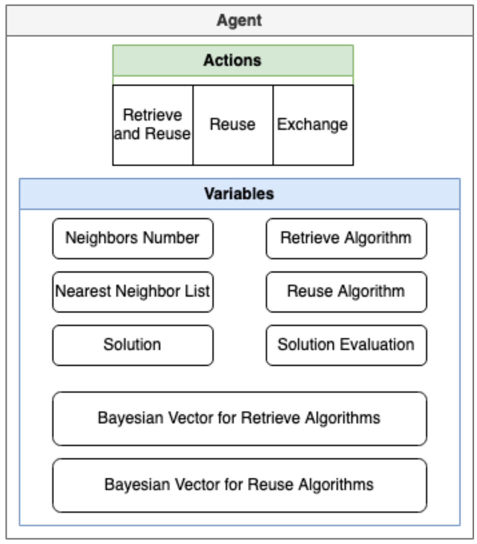

Figure 3: Agent Internal Structure

solutions.

Learning uses Bayesian reasoning, with retrieve and reuse vectors initially uniform (Figure 4). In each iteration, algorithms contributing to the best solution earn one point per agent. These points form a normalized likelihood vector $P(A)$, used in the Bayesian update (Eq. 3), with $P(B)$ as the global normalization term.

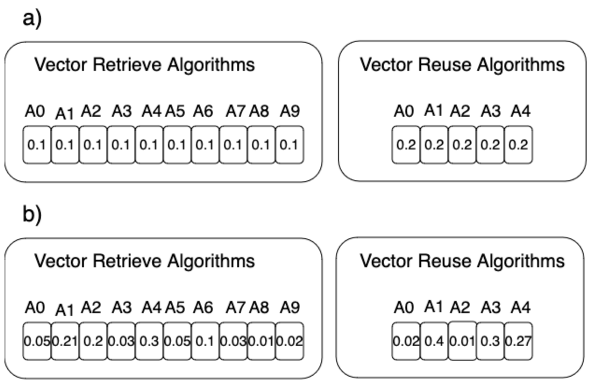

Figure 4: Example of the vector Bayesian evolutions for an agent. a) Initialization of probabilities $P(B|A)$ vectors for Retrieve and Reuse (A priori distribution), b) Probabilities after some iteration $P(A|B)$ vectors for Retrieve and Reuse (A posteriori distribution)

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \quad (3)$$

The selection of a retrieval algorithm a_{rt} is made through a random value taken from the discrete retrieve distribution vector obtained from Bayesian reasoning as shown in Equation 4.

$$a_{rt} = rnp(P_{rt}(A|B)) \quad (4)$$

The selection of a reuse algorithm a_{rs} is made through a random value taken from the discrete reuse distribution vector obtained from Bayesian reasoning as shown Equation 5.

$$a_{rs} = rnp(P_{rs}(A|B)) \quad (5)$$

The learning process is performed individually for each agent, but contains a collaborative aspect as the agents communicate the information they have found, the local optimizations they have achieved, and the algorithms and parameters.

3.4 Agent Exchanges

In each iteration, agents can randomly select a neighbor from their nearest neighbor list to exchange information, allowing high-performing parameters to propagate. Randomly chosen information—neighbors number, neighbor list, retrieve/reuse algorithms, or Bayesian vectors—is replaced with the neighbor's copy, providing feedback and improving solutions.

4 Results and Discussion

In order to compare the performance prediction and behavior of the proposed algorithm ESCBR-SMA (A11), eleven regression databases with different characteristics has been selected. The databases and their characteristics are shown in Table 2. The list of algorithms that have been compared are : Linear regression (A1), K-Nearest Neighbor (A2), Decision Tree (A3), Random Forest (A4), Multi-Layer Perceptron (A5), Polynomial Regression (A6), Ridge Regression (A7), Lasso Regression (A8), Gradient Boosting (A9), Ensemble Case-Based Regression (A10). The values for the parameters of algorithm are: $it = 100$, $np = 50$, $nl = 10$ and $ng = 10$.

The average ranking for all the databases considering the MAE (Median Absolute Error) metric, are shown in Table 3.

The results show that the algorithm is competitive in most of the selected datasets, effectively improving the ESCBR model proposed above and obtaining the best RMSE and MAE results in the DS4, DS5 and DS11 datasets. In comparison with the previous ESCBR model and according to RMSE the proposed model improves the results by 3.6%; with MAE, there is an improvement of 8.3% on average over all datasets.

Analyzing the characteristics of all datasets, it is possible to determine that the proposed model performs better with asymmetric datasets ($skewness > 0.5$) and flat distribution of probability or high probability of outliers with thresholds ($kurtosis < -0.5$ or $kurtosis > 1$) on average over all the data. The data quantity, descriptive dimension, dimension of solutions and domain type are insufficient parameters to effectively gauge successful outcomes. In some of the datasets evaluated (DS6, DS8), the results obtained are not highly satisfactory because when analyzing the data recorded in these datasets, there are cases where the records contain the same description, but the expected results are different. Despite this, the model is robust because it tries to obtain an approximate solution that unifies the results in contradictory cases. But, in the end, it may cause distant or extreme values in the metric used.

Globally, according to the average rank the algorithm is placed in third position, close to other ensemble algorithms. Compared to ESCBR, an improvement in results (RMSE: 22.31%, MAE: 14.33%) and variance reduction has been achieved, which shows that multi-agent systems and Bayesian stochastic reasoning contribute to the learning and convergence of the algorithm toward solutions closer to the real solution.

The global dispersion, median and outliers for six representative datasets are shown in Figure 5, which shows that the proposed algorithm in some cases generates more outliers than others algorithms, but the variance is very low and the convergence is close to the real value, better than most of the compared algorithms.

In addition to the results, the proposed algorithm presents several advantages with respect to the algorithms with which it has been compared. In addition to the advantages already obtained with the previous algorithm, there are: dynamic learning and adjustment capabilities as well as cooperative solution construction, which facilitates the search in the solution space.

5 Conclusion

The proposed ESCBR with MAS tries to use the advantages of multi-agent systems to improve the

Table 2: Description of evaluated datasets. (* After encoding String data)

ID	**DataSet**	**Features**	**Instances**	**Output Dimension**	**Input Domain**	**Output Domain**
DS1	Yatch Hydrodynamics	6	308	1	$\mathbb{R}$	$\mathbb{R}$
DS2	Electrical Grid Stability	12	10000	1	$\mathbb{R}$	$\mathbb{R}$
DS3	Real State Valuation	6	414	1	$\mathbb{R}_+$	$\mathbb{R}_+$
DS4	Wine Quality (Red)	11	1598	1	$\mathbb{R}_+$	$\mathbb{N}$
DS5	Wine Quality (White)	11	4897	1	$\mathbb{R}_+$	$\mathbb{N}$
DS6	Concrete Compressive Strength	8	1030	1	$\mathbb{R}_+$	$\mathbb{R}_+$
DS7	Energy Efficiency (ENB)	8	768	2	$\mathbb{R}_+$	$\mathbb{R}_+$
DS8	Gas Turbine CO, NOx Emission (GT - 2015)	9	7384	2	$\mathbb{R}_+$	$\mathbb{R}_+$
DS9	Student Performance Portuguese	30	649	3	$\mathbb{N}*$	$\mathbb{N}$
DS10	Student Performance Math	30	395	3	$\mathbb{N}*$	$\mathbb{N}$
DS11	Generated Student Performance	5	1000	1	$\mathbb{R}_+$	$\mathbb{R}_+$

Table 3: Comparison of the best MAE (Median Absolute Error) for multiple datasets with machine learning regression algorithms

Dataset	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11
DS1	6.776	2.385	0.231	0.207	3.632	6.778	6.307	5.186	0.162	1.193	1.1
DS2	0.015	0.017	0.012	0.008	0.012	0.015	0.015	0.030	0.007	0.011	0.010
DS3	5.092	4.320	4.1	3.632	4.435	5.092	5.20	5.132	3.504	3.9	3.771
DS4	0.413	0.495	0.18	0.325	0.451	0.413	0.412	0.544	0.387	0.154	0.135
DS5	0.509	0.548	0.285	0.374	0.550	0.509	0.509	0.633	0.456	0.113	0.085
DS6	6.989	5.709	3.134	2.839	4.306	6.989	6.989	6.986	3.084	5.439	5.072
DS7	1.393	1.372	0.217	0.218	2.523	1.393	1.529	2.346	0.243	1.008	1.006
DS8	0.549	0.297	0.365	0.289	0.742	0.549	0.549	0.540	0.309	0.861	0.794
DS9	1.496	1.788	2.080	1.612	2.005	1.496	1.496	1.714	1.538	1.721	1.556
DS10	2.344	2.534	2.910	2.331	2.543	2.344	2.344	2.481	2.258	2.602	2.371
DS11	0.387	0.35	0.46	0.338	0.384	0.387	0.387	0.453	0.327	0.35	0.347
Avg. Rank	7.22	7.05	5.5	2.55	7.95	7.32	7.36	8.90	2.45	5.51	4.72

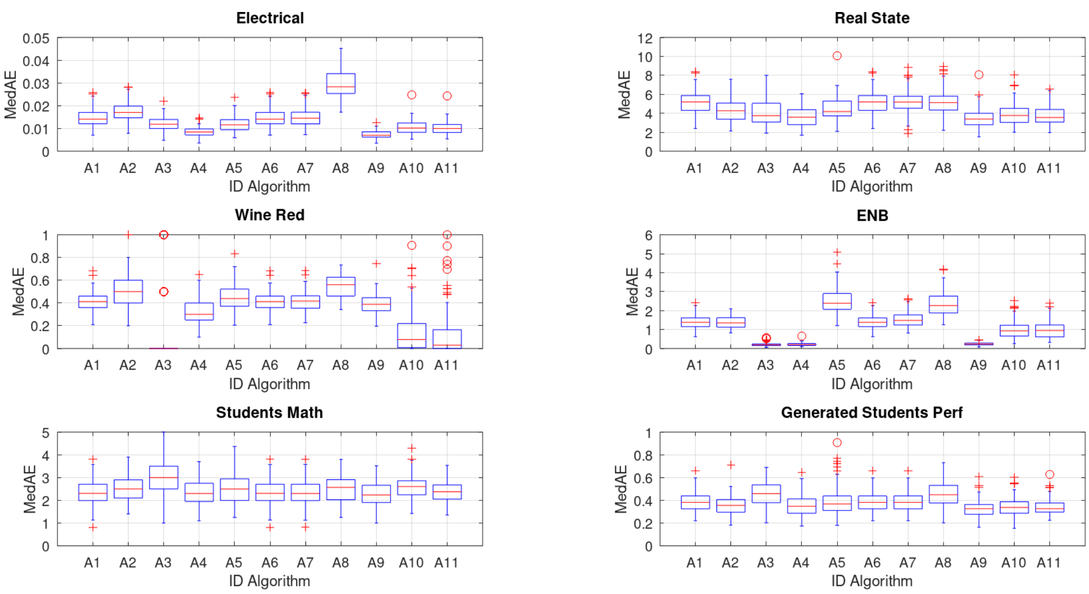

Figure 5: Results of MAE (Median Absolute Error) for eleven algorithms and six representative databases

quality of the proposed solutions as well as to improve the global learning process in order to increase the performance of the algorithm at each new execution even on different data, it also makes use of Bayesian reasoning, which allows obtaining good approximations and optimal learning with little data.

This work has demonstrated the model's capability to find solutions near to global optimum for the majority of the analyzed dataset, even with diverse, asymmetric and different-sized datasets. Indeed, the results show an improvement of basic ESCBR. This may be due to the combination of ESCBR potential and features of multi-agent systems such as feedback effects, emergent effects and the implemented cognitive Bayesian reasoning behavior.

In future work, a set of rules can be implemented together with the search algorithms to generate searches that explore the data more qualitatively. Also, new solution generation algorithms can be integrated and even tested with heterogeneous agents that perform specific functions, which can improve the performance and speed of the model. The ultimate goal of this model is its use in predicting student outcomes in both simulated and real datasets to improve the adaptation and revision processes in an autonomous intelligent tutoring system. Therefore, the proposed model behaves correctly, since it obtains good results in database DS11 (Generated Student Performance), and it is the only one of all the databases evaluated that corresponds to the structure and the objectives to achieve.

REFERENCES

Cicirello, A. and Giunta, F. (2022). Machine learning based optimization for interval uncertainty propagation. *Mechanical Systems and Signal Processing*, 170:108619.

Cinkusz, K. and Chudziak, J. A. (2025). Agile software management with cognitive multi-agent systems. In *Proceedings of the 17th International Conference on Agents and Artificial Intelligence - Volume 1: ICAART*, pages 385–392. INSTICC, SciTePress.

Didden, J. B., Dang, Q.-V., and Adan, I. J. (2023). Decentralized learning multi-agent system for online machine shop scheduling problem. *Journal of Manufacturing Systems*, 67:338–360.

Dyer, J., Cannon, P., Farmer, J. D., and Schmon, S. M. (2024). Black-box bayesian inference for agent-based models. *Journal of Economic Dynamics and Control*, 161:104827.

Hipólito, I. and Kirchhoff, M. (2023). Breaking boundaries: The bayesian brain hypothesis for perception and prediction. *Consciousness and Cognition*, 111:103510.

Jiao, L., Yang, H., ga Liu, Z., and Pan, Q. (2022). Interpretable fuzzy clustering using unsupervised fuzzy decision trees. *Information Sciences*, 611:540–563.

Kamali, S. R., Banirostam, T., Motameni, H., and Teshnehlab, M. (2023). An immune inspired multi-agent system for dynamic multi-objective optimization. *Knowledge-Based Systems*, 262:110242.

Kim, Y.-S., Walls, L. A., Krafft, P., and Hullman, J. (2019). A bayesian cognition approach to improve data visualization. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, page 1–14, New York, NY, USA. Association for Computing Machinery.

Li, J., Li, J., Wang, D., Mao, C., Guan, Z., Liu, Z., Du, M., Qi, Y., Wang, L., Liu, W., and Tang, P. (2023a). Hierarchical and partitioned planning strategy for closed-loop devices in low-voltage distribution network based on improved kmeans partition method. *Energy Reports*, 9:477–485. 2022 The 3rd International Conference on Power and Electrical Engineering.

Li, W., He, S., Mao, X., Li, B., Qiu, C., Yu, J., Peng, F., and Tan, X. (2023b). Multi-agent evolution reinforcement learning method for machining parameters optimization based on bootstrap aggregating graph attention network simulated environment. *Journal of Manufacturing Systems*, 67:424–438.

Nikpour, H. and Aamodt, A. (2021). Inference and reasoning in a bayesian knowledge-intensive cbr system. *Progress in Artificial Intelligence*, 10(1):49–63.

Petit, M., Dellandrea, E., and Chen, L. (2020). Bayesian optimization for developmental robotics with meta-learning by parameters bounds reduction. In *2020 Joint IEEE 10th International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, pages 1–8.

Prescott, T. P., Warne, D. J., and Baker, R. E. (2024). Efficient multifidelity likelihood-free bayesian inference with adaptive computational resource allocation. *Journal of Computational Physics*, 496:112577.

Rakesh, S., Hegde, N. P., Venu Gopalachari, M., Jayaram, D., Madhu, B., Hameed, M. A., Vankdothu, R., and Suresh Kumar, L. (2023). Moving object detection using modified gmm based background subtraction. *Measurement: Sensors*, 30:100898.

Rezaei, Z., Setayeshi, S., and Mahdipour, E. (2022). A biased inferential naivety learning model for a network of agents. *Cognitive Systems Research*, 76:1–12.

Ristic, B., Gilliam, C., Byrne, M., and Benavoli, A. (2020). A tutorial on uncertainty modeling for machine reasoning. *Information Fusion*, 55:30–44.

Saxena, N., Kumar, R., Rao, Y. K. S. S., Mondloe, D. S., Dhapekar, N. K., Sharma, A., and Yadav, A. S. (2024). Hybrid knn-svm machine learning approach for solar power forecasting. *Environmental Challenges*, 14:100838.

Soto-Forero, D., Henriet, J., and Betbeder, M.-L. (2024). Ensemble stacking case-based reasoning for regression. In *Proceedings of the 2024 ICCBR International Conference on Case-Based Reasoning*, New York, NY, USA. Springer.

Zhang, P., Wang, L., Fei, Z., Wei, L., Fei, M., and Menhas, M. I. (2023). A novel human learning optimization algorithm with bayesian inference learning. *Knowledge-Based Systems*, 271:110564.