Enhancing 3D Euler Number Calculation: Octo-Voxel Formulas and Optimized Regression Models

C. E. Muñoz-Chavez

***Abstract*—Metaheuristics have become widely popular in several disciplines, attracting considerable attention from both the scientific and industrial sectors. In recent years, there has been a significant increase in the advancement of novel metaheuristic algorithms, which are frequently derived from biological processes, human behavior, physical principles, or other natural occurrences. These high-level problem-solving frameworks are designed to provide heuristics that may effectively find, produce, or choose solutions for complicated optimization issues. They are particularly useful when time restrictions or the intrinsic complexity of the problem make it unfeasible to use exact approaches. Metaheuristics are particularly useful for tackling NP-hard problems, which are computationally impractical to solve optimally.**

This article explores the application of metaheuristics in determining the optimal parameters for Lasso-Lars and ARD regression, which are two linear regression models that involve variable selection. Our inquiry showcases the capacity of metaheuristics to optimize regression models, emphasizing their practicality and efficiency in this specific scenario. The objective of the study is to determine the necessity and effectiveness of employing metaheuristics in optimizing tasks of this nature, as well as their performance in parameter selection for linear regression models.

***Index Terms*—Optimization, Metaheuristic, Swarn intelligence, Evolutionary algorithms**

I. Introduction

The Euler characteristic, plays a crucial role in the characterization of images and objects [1]. This topological invariant has widespread applications, from discerning low bone density in vertebral tomographic images to quantifying the morphological traits and network representation of soil pore structures [2]. Furthermore, it aids in investigating the intricate relationship between elastic modulus, bone mineral density (BMD) [3], and trabecular microstructure in three dimensions [4].

The Euler number is a vital topological characteristic that is beneficial for the description of objects or images. It is a robust and adaptable descriptor that is readily integrated into algorithms and systems that are dedicated to the comprehension, interpretation, and identification of complex visual data due to its unique characteristics. In essence, the Euler number provides a practical method for expressing and capturing the fundamental topological properties of objects or images in computer structures [5].

Equation 1 expresses the Euler number, which is defined as the difference between the number of connected components O and the number of perforations H in a digital binary image, in accordance with mathematical principles. This equation is underlined to emphasize its inherent topological nature, which captures the essential structural characteristics of visual data.

$$e = o - h \tag{1}$$

The potential applications of 3D analysis are examined in this study after a thorough examination of the 3D case to ascertain its Euler number. The Euler formula is a shape descriptor that is a valuable tool for describing the geometry of 3D objects; it is an essential topological invariant. In addition to its primary function, its efficacy is extended to the estimation of structural connectivity. This formula effectively resolves a diverse array of specific issues by employing the Euler number, a critical component, to achieve remarkable outcomes.

The Euler formula is difficult to compute due to the use of volumetric representations. This technique is even more essential due to the prevalence of subtle characteristics in real-world objects, such as tunnels and caverns. An abundance of academic articles have been developed to address this issue, offering approaches that are explicitly designed to derive the Euler formula and are particularly suitable for the intricacies of three-dimensional objects [6].

In equation 2, the total number of cavities (or bubbles) within the image is denoted by 'c', while the total number of openings (or passageways) is denoted by 'h'. The overall count of objects is represented by 'o'. The Euler number is a fundamental topological measure that encapsulates the complex spatial features of the three-dimensional visual data.

$$e = o - h + c \tag{2}$$

Several crucial concepts need to be introduced at this point to facilitate a complete understanding of the content presented in this paper.

A "volume element," additionally referred to as a voxel, is a three-dimensional pixel used to represent a point in three dimensions. It serves as the base unit of a three-dimensional grid, extending the idea of a two-dimensional image's pixel, which represents a point in a two-dimensional grid. More

specifically, a resolution cell with three Cartesian coordinates within a three-dimensional grid is called a voxel.

Sánchez et al. [6] introduced concepts that will help simplify the known Euler-Poincaré formula, which includes both the octo-voxel and the tetra-voxel. In their paper, they elucidate that an octo-voxel comprises a compact arrangement of eight interconnected voxels within a cubic structure.

Sánchez et al. [6] define an octo-voxel as a cubic volume measuring 2×2×2, housing precisely eight neighboring voxels. Fig. 1.

As commonly understood, a 2×2×2 voxel pattern comprises eight voxels, each designated as a background voxel or a foreground voxel. Consequently, theoretically, there exist 256 distinct configurations for a 2×2×2 voxel pattern. To determine the specific configuration of a voxel pattern, it is imperative to evaluate it from all possible orientations [7].

The truth table used in the study is displayed in Table I and was taken from the work of Yao et al. [7]. This table was crucial for figuring out the Euler number for images that were randomly generated using a specialized technique.

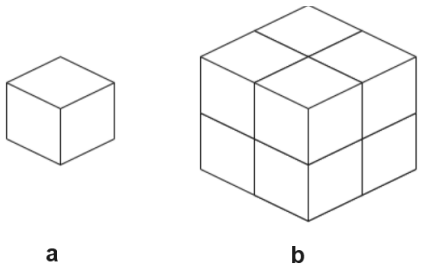

Fig. 1: a) This is the visual depiction of a voxel. b) It illustrates the visual representation of a 2×2×2 octo-voxel.

II. Proposed Approach

Table I illustrates the method utilized only 49 of the 256 potential configurations for a 2×2×2 voxel pattern showed non-zero Euler number increases. Thirty patterns in this sample of configurations that affected a 3D image's Euler number had increments of -1, seventeen patterns had increments of 1, and two patterns had increments of -2.

The development of an algorithm for calculating the Euler number from three-dimensional images required the use of this algorithm, which was essential in establishing the study methodology. In order to calculate the Euler number exactly, the process involves accessing and obtaining each octo-voxel in the image one by one and looking at every possible combination.

Similarly to a kernel, a stride of 1 was required to collect every possible combination of octo-voxels. On the other hand, padding was added in parallel to enclose the 3D image and provide error-free image reading.

A dataset was employed for this experimentation, denoted as Dataset A, serving as the training set. This data set comprises 890 synthetic binary 3D images, each 16×16×16 voxels. These images were randomly generated as part of the training data.

TABLE I: Indexes of the Euler number increments of voxels patterns.

Index	!E	Index	!E	Index	!E	Index	!E
00000010	1	00001001	-1	00001011	-1	00011000	-1
00011001	-1	00011010	-1	00011011	-1	00100001	-1
00100011	-1	00100100	-1	00100101	-1	00100110	-1
00100111	-1	00101000	-1	00101001	-2	00101010	-1
00101011	-2	00101100	-1	00101101	-1	00101110	-1
00101111	-1	00111000	-1	00111001	-1	00111010	-1
00111011	-1	10000001	-1	10000011	-1	10001001	-1
00111011	-1	10000001	-1	10000011	-1	10001001	-1
10001011	-1	10010100	1	10010101	1	10010110	1
10010111	1	10011100	1	10011101	1	10011110	1
10011111	1	10100001	-1	10100011	-1	10101001	-1
10101011	-1	10110100	1	10110101	1	10110110	1
10110111	1	10111100	1	10111101	1	10111110	1
10111111	1	Others	0				

$$f(x,y,z) = \begin{cases} 1, if\ \upsilon(0,1) \geq 0.5 \\ 0, otherwise \end{cases} \tag{3}$$

where $\upsilon(0,1)$ is a uniform random number in the range of (0, 1).

$$\beta(i,j,k) \in \{0\}^{(y+2)\times(x+2)\times(z+2)} \tag{4}$$

The expression $\beta(i,j,k)$ denotes a larger cube constructed using the dimensions x, y, and z. This augmented cube serves as the padding necessary to accommodate the placement of $f(x,y,z)$ within its confines.

$$f(x,y,z)\ is\ centered\ within\ \beta(i,j,k) \tag{5}$$

This additional padding introduces a higher computational load as it increases the values of x, y, and z by 2 (18×18×18). Nevertheless, this approach ensures comprehensive analysis and retrieval of all voxels when employing this algorithm.

The determination of the index involves the conversion of the voxel values into a binary string denoted $abcdefgh$. As depicted in Fig. 2, consider a voxel pattern where the values of voxels a, b, c, d, e, f, g, and h are 0, 0, 0, 0, 0, 0, 1, and 0, respectively. This configuration yields a binary string of 00000010, and the corresponding voxel pattern's index is expressed as the decimal number 2.

With reference to Table I, this pattern's Euler number increment is 1. As a result, 1 is the current Euler number. Nevertheless, the resulting Euler number drops to 0 if the binary representation 00001001, which corresponds to -1 in Table 1, is taken into consideration.

Fig. 2 illustrates the necessity of obtaining combinations of octo-voxels from the image to accurately analyze and interpret the Euler number increments.

To extract octo-voxel combinations, we trained a variety of regression models. Based on their prevalence in the current state of the art, it was explicitly determined to employ two models: ARD regression [8] and LassoLars regression [9]. We evaluated each model for both performance and time efficiency by comparing the prediction error to the actual data. These models were chosen for their proved efficacy in high-dimensional data scenarios and their prominence in current machine learning applications.

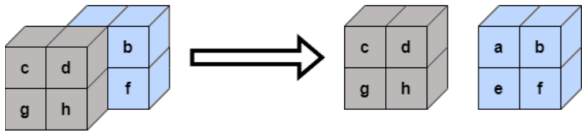

Fig. 2: Accessing order of voxel in a voxel pattern.

A. Automatic Relevance Determination (ARD) Regression

ARD regression is a Bayesian linear regression model that uses automatic relevance determination to enhance feature selection. By placing a prior distribution over the model parameters, ARD can effectively identify and prioritize the most relevant features. This capability is particularly beneficial when dealing with high-dimensional datasets, such as those encountered in voxel-based analyses.

Advantages:

- **Feature Selection:** ARD automatically identifies important features by shrinking the coefficients of irrelevant ones towards zero.
- **Robustness:** It maintains model performance even in the presence of many irrelevant features.
- **Interpretability:** The probabilistic framework provides insights into the significance of each feature.

Disadvantages:

- **Complexity:** ARD is computationally more intensive compared to simpler regression models.
- **Sensitivity to Priors:** The model's performance can be influenced by the choice of prior distributions.

B. Least Angle Regression with Lasso (LassoLars)

LassoLars regression integrates the strengths of Lasso (Least Absolute Shrinkage and Selection Operator) and Least Angle Regression (LARS). Lasso adds an L1 penalty to the regression objective, promoting sparsity by driving some coefficients to zero. LARS is an efficient algorithm for high-dimensional data, incrementally including features based on their correlations with the residuals.

Advantages:

- **Sparsity:** LassoLars effectively selects variables and regularizes the model, leading to simpler and more interpretable solutions.
- **Efficiency:** The LARS algorithm is computationally efficient, making it suitable for large datasets.
- **Simplicity:** Produces straightforward models that are easy to interpret.

Disadvantages:

- **Bias Introduction:** The L1 penalty can introduce bias into the coefficient estimates.
- **Parameter Sensitivity:** Model performance is sensitive to the choice of the regularization parameter.

C. Performance and Time Efficiency Evaluation

Both ARD and LassoLars regression models were rigorously tested to evaluate their performance and time efficiency. The evaluation was conducted by comparing the prediction error against actual data, with the following criteria:

Prediction Error: This metric was used to measure the accuracy of the models. A lower prediction error indicates higher model accuracy.

Computational Time: The time required to train the models and generate predictions was recorded to assess their efficiency. This is crucial for practical applications, especially with large datasets or in real-time processing scenarios.

The rigorous testing ensured that both models could balance low prediction error with acceptable computational times, making them suitable for extracting meaningful patterns from octo-voxel data.

D. Metaheuristic Approaches

To further evaluate the necessity and efficiency of using regression models for this application, we employed four metaheuristic algorithms: Artificial Bee Colony (ABC) [10], Differential Evolution (DE) [11], Harris Hawks Optimization (HHO) [12], and Particle Swarm Optimization (PSO) [13], and . These algorithms were chosen for their effectiveness in optimization tasks across various domains.

Metaheuristics:

- **Differential Evolution (DE):** DE is a population-based optimization algorithm that uses differential mutation and recombination to evolve the population towards the global optimum.
- **Harris Hawks Optimization (HHO):** HHO is inspired by the cooperative behavior and chasing strategy of Harris hawks, and it effectively balances exploration and exploitation during the optimization process.
- **Particle Swarm Optimization (PSO):** PSO simulates the social behavior of birds flocking or fish schooling to explore the search space and converge to the best solution.
- **Artificial Bee Colony (ABC):** ABC is inspired by the foraging behavior of honey bees, using employed and onlooker bees to explore and exploit the search space.

The performance of these metaheuristics was compared to determine whether it is necessary to use more than one metaheuristic approach or if a single one is sufficient for optimizing the regression models. This comparative analysis helped in understanding the robustness and efficiency of the regression models enhanced by these metaheuristic techniques.

E. Results

The ARD and LassoLars regression models were found to be effective in identifying octo-voxel combinations, as evidenced by the experiments. The ARD model exhibited extraordinary feature selection capabilities, which enabled it to maintain its robustness in the presence of irrelevant features. In contrast, LassoLars generated sparse, interpretable models and improved computational efficiency. The metaheuristic comparison demonstrated that all four approaches enhanced the optimization method; however, the significance of utilizing multiple metaheuristics is reliant on the application's complexity and specific requirements. The decision between

these models and metaheuristics may be determined by the specifications for computational efficiency, interpretability, and optimization robustness..

In this study, we split a data set into two parts, which we referred to as Data Sets A and B, and employed them as training sets. This set comprises 12,000 synthetic 3D binary images, with sizes ranging from 3×3×3 to 64×64×64 voxels.

The six subsets of each of the data sets A and B correspond to distinct resolutions: 3×3×3, 4×4×4, 8×8×8, 16×16×16, 32×32×32, and 64×64×64 dollars. We produced 1,000 images at each of these resolutions, yielding a total of 6,000 images for Data Set A and an additional 6,000 images for Data Set B.

In order to ensure a fair distribution of training samples across both sets, we partitioned the resolutions into sections. The decision was made to maintain this structure to facilitate the programming and interpretation of the results, as well as the diversity of the training and test data, despite the fact that the regression models trained on each data set could have extracted the same coefficients without evident differences.

Knowing how the database was divided, we focused on the search for the optimal parameters of the regression models mentioned above. The search was based on a comparison of time and the parameters of each metaheuristic. These parameters vary depending on the specific metaheuristic being used. In this study, we considered the following parameters for each metaheuristic.

- For Differential Evolution (DE): the number of individuals (30, 60, and 90), a mutation factor of 0.7, and a crossover probability of 0.8.
- For Artificial Bee Colony (ABC): the number of bees (30, 60, and 90).
- For Harris Hawks Optimization (HHO): the number of hawks (30, 60, and 90).
- For Particle Swarm Optimization (PSO): the number of particles (30, 60, and 90), inertia of 0.729, cognitive weight of 1.494, and social weight of 1.494.

For the regression models, adjustments were made to the hyperparameters as follows: LassoLars was tuned for α with bounds between 1.0 and 1×10^{-6} and for the number of iterations between 100 to 1000, while ARD used α_1, α_2, λ_1, λ_2 with bounds between 1.0 to 1×10^{-8} and for the number of iterations between 100 to 1000.

The evaluation metric for both regression models was Mean Squared Error (MSE), and the dataset consisting of 1000 64x64x64 images was the only dataset utilized for comparison. The LassoLars regression results in Table II indicate that parameter search was unnecessary, as the formula length remained consistent with that of Euler.

The results of the ARD regression were more intricate and diverse, but they were ultimately consistent. The same formula was obtained in all trials, despite the variability and multiple MSE outcomes. This consistency implies that the metaheuristic models did not necessitate parameter adjustments for these models (see Table III).

III. Conclusion and Future Work

Regression models serve as powerful tools for deriving complex outcomes that may be challenging to achieve with simpler techniques. In this study, we extracted Euler numbers using the concept of octo-voxels, considering all combinations except the one where every voxel in the array is zero. We applied this method to a range of images for algorithm training, ensuring that the input data was sufficient for accurate Euler number calculations in 3D images, regardless of their resolution.

Future work will focus on identifying the most important combinations of octo-voxels and disregarding the less significant ones. The primary objective is to identify the critical data parameters required for a precise computation of the Euler number. Additionally, we plan to explore alternative classifiers and evaluate their performance by experimenting with various hyperparameters. We will generate more images of varying sizes to enhance the training process and provide a richer data set.

Our findings show that metaheuristics are not needed for this type of problem due to its nature and the good results of the regression models. However, metaheuristics could be beneficial in future projects, such as those involving convolutional neural networks (CNNs), where optimizing hyperparameters will be significantly more complex than in the current regression techniques with fewer parameters. Using metaheuristics in such scenarios could lead to substantial improvements in model performance.

For more detail, go to the linked site, which includes programming details along with additional outcomes.This repository : https://github.com/DreamnovaCesar/Project_2_Semester

Acknowledgment

César Eduardo Muñoz Chavez wants to thank CONACYT for the support of this research.

References

[1] J.H. Sossa-Azuela, Á.A. Carreón-Torres, R.Santiago-Montero, E.Bribiesca-Correa, and A.Petrilli-Barceló, "Efficient Computation of the Euler Number of a 2-D Binary Image," in *Advances in Computational Intelligence*

[2] Vogel, H.J., Roth, K.: Quantitative morphology and network representation of soil pore structure. Adv. Water Resour. 24, 233–242 (2001)

[3] W.Roque, A.Souza, and D.Barbieri, "The Euler-Poincaré characteristic applied to identify low bone density from vertebral tomographic images," *Revista Brasileira de Reumatologia*, vol. 49, pp. 140–145, Apr. 2009, doi: 10.1590/S0482-50042009000200006.

[4] T. Uchiyama, T. Tanizawa, H. Muramatsu, N. Endo, H. E. Takahashi and T. Hara. Three-Dimensional Microstructural Analysis of Human Trabecular Bone in Relation to Its Mechanical Properties. Bone. 25(4) (1999) 487-491.

[5] H.Sossa and H.Sánchez, "Computing the number of bubbles and tunnels of a 3-D binary object," in *Pattern Recognition Applications and Methods - 5th International Conference, ICPRAM 2016, Revised Selected Papers*, M. De Marsico, G. S. di Baja, and A. Fred, Eds., vol. 10163 LNCS, Lecture Notes in Computer Science, Springer Verlag, 2017, pp. 194–211.

[6] H.Sánchez-Cruz, H.Sossa-Azuela, U.-D. Braumann, and E.Bribiesca, "The Euler-Poincaré Formula Through Contact Surfaces of Voxelized Objects," *Journal of Applied Research and Technology*, vol. 11, no. 1, pp. 65–78, 2013, doi: 10.1016/S1665-6423(13)71515-3.

TABLE II: Table comparing LassoLars regression outcomes using different metaheuristics

ABC						
Number of bees	**Alpha**	**Iterations**	**Time**	**Formula length**	**Iteration found**	**MSE**
30	1e-08	512	25.6	84	1	4.439e-15
60	1e-08	612	68.9	84	1	4.439e-15
90	1e-08	528	95.8	84	1	4.439e-15
DE						
Number of individuals	**Alpha**	**Iterations**	**Time**	**Formula length**	**Iteration found**	**MSE**
30	1e-08	442	32.2	84	1	4.439e-15
60	1e-08	523	61.7	84	1	4.439e-15
90	1e-08	710	91.3	84	1	4.439e-15
HHO						
Number of hawks	**Alpha**	**Iterations**	**Time**	**Formula length**	**Iteration found**	**MSE**
30	1e-08	100	11	84	1	4.439e-15
60	1e-08	100	23.4	84	1	4.439e-15
90	1e-08	100	89	84	1	4.439e-15
PSO						
Number of particles	**Alpha**	**Iterations**	**Time**	**Formula length**	**Iteration found**	**MSE**
30	1e-08	320	29.3	84	1	4.439e-15
60	1e-08	492	65.9	84	1	4.439e-15
90	1e-08	940	98.5	84	1	4.439e-15

TABLE III: Table comparing ARD regression outcomes using different metaheuristics

ABC								
Number of bees	**Alpha 1**	**Alpha 2**	**lambda 1**	**lambda 2**	**Iteration found**	**Time**	**Formula length**	**MSE**
30	0.475	0.017	0.201	0.846	9	117	194	4.82e-16
60	9.31e-01	1e-06	1e-06	5.84e-01	6	252.2	194	7.86e-16
90	5.69e-01	1e-06	6.02e-01	9.16e-01	6	451.6	194	6.39e-16
DE								
Number of individuals	**Alpha 1**	**Alpha 2**	**lambda 1**	**lambda 2**	**Iteration found**	**Time**	**Formula length**	**MSE**
30	1e-06	1e-06	2.00e-01	8.41e-01	9	139.4	194	9.36e-16
60	1e-06	8.00e-03	8.08e-01	7.95e-01	9	271.4	194	1.26e-15
90	1e-06	1e-06	1.07e-01	2.20e-01	4	342.9	194	2.69e-16
HHO								
Number of hawks	**Alpha 1**	**Alpha 2**	**lambda 1**	**lambda 2**	**Iteration found**	**Time**	**Formula length**	**MSE**
30	1.27e-05	1e-06	1.51e-05	4.78e-02	10	100.6	194	6.13e-16
60	0.0009	0.0001	0.0001	0.0006	10	213.5	194	3.20e-16
90	0.0008	0.001	0.001	0.001	1	336.1	194	3.09e-16
PSO								
Number of particles	**Alpha 1**	**Alpha 2**	**lambda 1**	**lambda 2**	**Iteration found**	**Time**	**Formula length**	**MSE**
30	8.67e-01	1e-06	1e-06	9.12e-01	8	103.3	194	7.92e-16
60	6.7e-013	1e-06	3.86e-01	9.52e-01	8	192.7	194	1.74e-16
90	4.81e-01	1e-06	5.42e-01	3.84e-03	4	776.8	194	2.85e-16

[7] B.Yao, D.Han, S.Kang, Y.Chao, and L.He, “A Novel Method for Improving the Voxel-Pattern-Based Euler Number Computing Algorithm of 3D Binary Images,” in *Image Analysis and Processing. ICIAP 2022 Workshops*, P. L. Mazzeo, E. Frontoni, S. Sclaroff, and C. Distante, Eds. Cham: Springer International Publishing, 2022, pp. 84–94.
[8] D. J. C. MacKay, ‘Bayesian Non-Linear Modeling for the Prediction Competition’, in Maximum Entropy and Bayesian Methods: Santa Barbara, California, U.S.A., 1993, G. R. Heidbreder, Ed. Dordrecht: Springer Netherlands, 1996, pp. 221–234.
[9] R. Tibshirani, ‘Regression Shrinkage and Selection Via the Lasso’, Journal of the Royal Statistical Society: Series B (Methodological), vol. 58, no. 1, pp. 267–288, 1996.
[10] D. Karaboga and B. Basturk, ‘Artificial Bee Colony (ABC) Optimization Algorithm for Solving Constrained Optimization Problems’, in Foundations of Fuzzy Logic and Soft Computing, 2007, pp. 789–798.
[11] R. Storn and K. Price, ‘Differential Evolution: A Simple and Efficient Adaptive Scheme for Global Optimization Over Continuous Spaces’, Journal of Global Optimization, vol. 23, 01 1995.
[12] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, ‘Harris hawks optimization: Algorithm and applications’, Future Generation Computer Systems, vol. 97, pp. 849–872, 2019.
[13] J. Kennedy and R. Eberhart, ‘Particle swarm optimization’, in Proceedings of ICNN’95 - International Conference on Neural Networks, 1995, vol. 4, pp. 1942–1948 vol.4.

César Eduardo Muñoz-Chavez was born in 1994. He received a degree in Mechatronics Engineering from the UTA (Universidad Tecnologica de Aguascalientes). His research interests include Artificial neuronal network and deep learning.