Towards Interpretable Alzheimer's Disease Detection: Comparing Genetic Programming Paradigms on Handwriting Data

Emanuele Nardone[1][0009−0005−8718−5435], Leonardo Vanneschi[2][0000−0003−4732−3328], Francesco Fontanella[3][0000−0002−3242−0179], and Tiziana D'Alessandro[3][0009−0005−4340−7227]

[1] Department of Physics and Mathematics, Center for Photonics Sciences, University of Eastern Finland, 80100 Joensuu, Finland
{emanuele.nardone}@uef.fi

[2] NOVA Information Management School (NOVA IMS), Universidade Nova de Lisboa, Campus de Campolide, Lisboa 1070-312 – Portugal
{lvanneschi}@novaims.unl.pt

[3] Department of Electrical and Information Engineering (DIEI), University of Cassino and Southern Lazio, Via G. Di Biasio 43, 03043 Cassino (FR), Italy
{fontanella, tiziana.dalessandro}@unicas.it

Abstract. Handwriting analysis is a promising non-invasive approach for the early detection of Alzheimer's disease (AD). However, most current analytical pipelines rely on classifiers that are difficult to interpret, thereby limiting their suitability for clinical implementation. This study investigates Symbolic Regression (SR) as an interpretable alternative and systematically compares three paradigms on the DARWIN benchmark (174 participants, 25 handwriting tasks): standard tree-based Genetic Programming (GP), Geometric Semantic Genetic Programming (GSGP), and six variants of the Semantic Learning algorithm based on Inflate and deflate Mutation (SLIM). Two feature representations are evaluated: 450 raw kinematic descriptors and 2,150 handcrafted statistical descriptors. Using a fixed hold-out partition and 20 seeds per method, GSGP achieves the highest mean accuracy among the GP-based methods (0.78 ± 0.06 on kinematic features) and exceeds the retrained conventional Machine Learning (ML) baselines evaluated under the same split. SLIM addition-family variants provide the best accuracy–complexity trade-off, producing models approximately $25\times$ smaller than GSGP and training $2.4\times$ faster. We also show that, under the specific fixed-threshold sigmoid pipeline adopted here, SLIM multiplication-family variants become degenerate because their non-negative aggregated outputs cannot fall below the zero logit required for negative predictions. Ablation results suggest that $p_{\text{inflate}} = 0.5$ is a useful default for the SLIM addition family. Raw kinematic features yield stronger performance than the higher-dimensional statistical representation under the present protocol.

Keywords: Alzheimer's disease · Handwriting analysis · Symbolic regression · Genetic programming · Interpretable machine learning.

1 Introduction

AD represents the most prevalent form of dementia. According to the latest WHO dementia fact sheet, 57 million people had dementia worldwide in 2021, and AD may contribute to 60–70% of cases [21]. AD is a progressive neurodegenerative disorder that impairs memory, language, and executive function. Pharmacological interventions demonstrate the greatest efficacy when initiated early in the disease [11]. Consequently, there is a critical need for non-invasive, cost-effective, and user-friendly screening tools suitable for large-scale implementation. Handwriting integrates cognitive, perceptual, and fine motor processes [15]. Neurological deterioration in AD is reflected in quantifiable changes in handwriting kinematics, such as reduced velocity, increased pressure variability, prolonged in-air time, and diminished stroke fluency, often preceding clinical diagnosis [20]. These findings have motivated research on digitized handwriting as a biomarker for AD detection [9,2]. The DARWIN (Diagnosis AlzheimeR WIth handwritiNg) dataset [1] has become a benchmark resource in this domain. It comprises kinematic and dynamic data from 25 handwriting tasks performed by both individuals with AD and healthy controls. Most ML approaches for AD detection in handwriting use ensemble methods, support vector machines, or deep neural networks [2,5]. Although these models often achieve high accuracy, their decision-making processes are difficult to interpret. In clinical practice, stating the rationale for a diagnosis is necessary to build clinician trust and meet regulatory standards [17]. This challenge has driven efforts to develop more interpretable models. SR addresses this gap by identifying mathematical expressions that accurately fit the data while remaining interpretable for human analysis [12]. GP is the primary algorithm for SR, evolving populations of syntax trees through operators inspired by biological evolution [10]. Standard tree-based GP algorithms exhibit well-documented limitations, including bloat, difficulty with high-dimensional inputs, and sensitivity to random initialization [14]. GSGP addresses some of these challenges by operating directly in the semantic space of programs, ensuring that offspring possess geometric properties that lead to smoother convergence [13]. The SLIM has recently been introduced as a semantic-aware GP variant. SLIM constructs models as weighted combinations of sub-trees and incorporates an inflation and deflation mechanism to regulate model complexity [18]. Despite these advances, SR remains underexplored in clinical classification, and no prior work has jointly compared GP, GSGP, and SLIM in this context. The interaction between SLIM constraints and standard classification pipelines, as well as the choice between kinematic and statistical representations, remains unclear. The present study addresses these gaps through an empirical analysis of the DARWIN dataset, contrasting SR with conventional ML pipelines that dominate handwriting-based AD detection [2,5,6] and typically trade interpretability for predictive performance. To ensure a protocol-matched comparison, the conventional ML classifiers considered here were retrained using the same fixed-partition protocol as the GP-based methods. A recurring finding is that SLIM multiplication-family variants become degenerate under the adopted fixed-threshold sigmoid pipeline because of their

non-negative outputs, a pipeline-dependent limitation that is analyzed in detail below.

The main contributions of this work are fourfold:

(i) a protocol-matched empirical comparison of GP, GSGP, six SLIM variants, and conventional ML baselines for handwriting-based AD classification;
(ii) an analysis of how the fixed-threshold sigmoid pipeline interacts with non-negative SLIM multiplication outputs;
(iii) ablation studies on SLIM inflation probability and GP initialization depth;
(iv) a controlled comparison of two handwriting-derived feature representations under a reproducible fixed-partition protocol.

2 Materials and Methods

2.1 Participants and Data Collection

The data employed in this study were collected from $N = 174$ participants, including 89 patients diagnosed with AD (class P) and 85 healthy controls (class H). Each participant completed 25 handwriting tasks, ordered by increasing cognitive difficulty and designed to assess combinations of motor, visuospatial, language, and memory functions. The tasks were organized into four categories: graphic tasks (6), which included basic strokes, point-to-point connections, and drawing or retracing geometric figures of varying size and complexity; copy and reverse-copy tasks (14), involving letters, words, and numbers, including items written in reverse order with varying length and spatial layout; a memory task (1), which required participants to write words recalled from memory; and dictation tasks (4), in which participants wrote short phrases or numbers under auditory dictation, thereby engaging working memory. A complete description of the protocol is available in [3]. Data were collected using a Wacom Bamboo Folio tablet equipped with a sensorized pen. Participants wrote on standard paper sheets secured to the tablet to maintain a natural writing experience. The device recorded pen position (x, y), pressure (z), and timestamps at a sampling rate of 200 Hz. In addition to capturing ink traces on the paper, the system recorded pen movements up to approximately 3 cm above the writing surface. As a result, the handwriting dataset comprises two complementary trajectory types: on-paper trajectories, corresponding to actual ink traces, and in-air trajectories, corresponding to pen-up movements between successive strokes.

2.2 Feature Representations

Two feature representations were derived from the same raw handwriting signals and compared throughout this study.

DARWIN kinematic features The DARWIN (Diagnosis AlzheimeR WIth handwritiNg) dataset [1] provides a publicly available benchmark for the automatic

detection of AD based on handwriting data. Eighteen kinematic and dynamic features were extracted for each of the 25 tasks. These features include air time, paper time, total time, mean speed and acceleration (calculated separately for on-paper and in-air segments), mean and variance of pressure, number of pen-down events, maximum horizontal and vertical extension, displacement index, and mean jerk (on-paper and in-air). This process yields a total of $18 \times 25 = 450$ features per subject, capturing temporal characteristics, pressure statistics, and stroke-segmentation properties of the handwriting signal.

Hand-crafted statistical features (Hand-Stat) A second, richer feature set was constructed from the same raw trajectories following the extraction protocol described in [4]. The features were designed to capture both static properties, such as stroke geometry, and dynamic properties, such as velocity and acceleration profiles, of the handwriting signal. To capture distinct handwriting phases, features were extracted separately from on-paper and in-air samples. On-paper features capture execution dynamics, while in-air features reflect planning-related pen-up movements. For each task, the on-paper and in-air feature vectors were concatenated to form a unified representation, resulting in 86 features per task. These features include stroke-level means and standard deviations for duration, vertical and horizontal start positions, vertical and horizontal sizes, peak vertical velocity, peak vertical acceleration, straightness error, slant, loop surface, absolute and normalized jerk, road length, and average absolute velocity. Concatenating features across all 25 tasks produced $86 \times 25 = 2{,}150$ task-derived features per subject. The dataset also contains four demographic variables (age, gender, years of education, and work status) that have been shown to influence handwriting [4]. These variables were excluded from the feature matrix to ensure that the comparison between representations reflected only handwriting-derived information. This high-dimensional setting ($p = 2{,}150$, $N = 174$) places the data in a $p \gg N$ regime that challenges learning algorithms.

2.3 Algorithms

In this study, we compared three families of SR algorithms, each adapted for binary classification using sigmoid activation.

Standard Genetic Programming. Tree-based GP evolves mathematical expressions that are represented as syntax trees. The initial population of trees was generated using the Ramped Half-and-Half method with an initial depth of 6. A maximum depth of 15 was permitted during evolution. The function set included four binary arithmetic operators: addition, subtraction, multiplication, and protected division. The terminal set consisted of the input features and five constants $(-1, 2, 3, 4, 5)$. This fixed constant set follows the default configuration of the `slim-gsgp` library. Ephemeral random constants (ERCs), which are commonly used in SR applications, were not employed. Reproduction was performed using subtree crossover with a probability of $p_{\text{xo}} = 0.8$ and subtree mutation with a probability of $p_m = 0.2$.

Geometric Semantic Genetic Programming. GSGP operates directly on the semantics of programs, guaranteeing that offspring lie on the segment (for crossover) or within a ball (for mutation) connecting the parents in the semantic space [13]. Geometric semantic mutation was applied with probability $p_m = 1.0$, meaning that every individual was mutated in each generation. Crossover was disabled ($p_{\text{xo}} = 0.0$).

Semantic Learning algorithm based on Inflate and Deflate Mutation GSGP. SLIM is a recent semantic-aware GP variant that constructs models as linear combinations of sub-trees. Model complexity is balanced using an inflation and deflation mechanism [18]; the non-bloating behavior of these operators and the specific variants studied here have been characterized in recent work [19,8]. Six SLIM variants were evaluated, derived from two operator families and three deflation strategies:

- **Addition family** (+): SLIM+ABS, SLIM+SIG1, SLIM+SIG2.
- **Multiplication family** (×): SLIM*ABS, SLIM*SIG1, SLIM*SIG2.

In this context, ABS denotes absolute-value deflation, while SIG1 and SIG2 refer to two distinct sigmoid-based deflation strategies. SLIM was configured with a maximum tree depth of 15 and a default inflation probability of $p_{\text{inflate}} = 0.2$.

Conventional machine learning baselines. To contextualize the symbolic-regression results, five conventional classifiers were included as protocol-matched baselines: support vector machines (SVM), Gaussian naive Bayes (GNB), logistic regression (LR), linear discriminant analysis (LDA), and k-nearest neighbors (KNN). All five models were trained and evaluated on the same fixed 80/20 hold-out partition used for GP, GSGP, and SLIM, using the same feature matrices and preprocessing choices for each representation. These baselines provide a direct reference for assessing the predictive performance of GP-based symbolic-regression methods under the experimental protocol adopted in this study. To ensure reproducibility, the ML baselines were implemented in `scikit-learn` using an identical train–test split. Test set information was excluded from both training and model selection. Each baseline was evaluated across 20 runs with unique random seeds, and performance metrics are reported as mean and standard deviation, consistent with the GP-based protocol.

Structural constraint of the multiplication family. In SLIM* variants, the model output is computed as the product of activations from each block:

$$f_{\times}(\mathbf{x}) = \prod_{i=1}^{B} a(g_i(\mathbf{x}))\,, \tag{1}$$

Here, g_i denotes the evolved sub-trees, and $a(\cdot)$ represents the deflation activation function applied to each block. For all three deflation strategies, the activation function a is non-negative: $|\cdot|$ (ABS) maps to $[0, +\infty)$, while $\sigma(\cdot)$ (SIG1, SIG2) maps to $(0, 1)$. As a result, $f_{\times}(\mathbf{x}) \geq 0$ for all inputs $\mathbf{x}$, independent of the specific learned block functions. In contrast, SLIM+ aggregates block activations

through summation, and the inclusion of an unconstrained additive offset enables the combined output to span the entire real line, thereby preserving the range necessary for binary discrimination.

2.4 Fitness Function and Classification

All three algorithm families minimized the *binary cross-entropy* (BCE) loss as the fitness function. The real-valued output of each evolved program was transformed into a class probability using a sigmoid activation function:

$$\hat{y} = \frac{1}{1 + e^{-f(\mathbf{x})}} \tag{2}$$

where $f(\mathbf{x})$ is the program output for input $\mathbf{x}$. A threshold of 0.5 was applied to generate binary predictions. The elite individual was evaluated on both the training and test partitions at each generation. The 0.5 decision threshold corresponds, after inversion of eq. (2), to a program output of zero. Because $f_{\times}(\mathbf{x}) \geq 0$ for all inputs (eq. (1)), the sigmoid function satisfies $\sigma(f_{\times}(\mathbf{x})) \geq \sigma(0) = 0.5$ for all cases. Consequently, with the fixed 0.5 threshold used in this study, SLIM* variants classify every instance as positive, regardless of the evolved program. This constraint renders them structurally incapable of producing true negatives. This limitation arises specifically from the combination of non-negative outputs and a fixed decision boundary. Introducing a learnable bias term (i.e., $f_{\times}(\mathbf{x}) - b$), calibrating the threshold on a held-out validation set (for example, using Youden's J statistic), or appending a signed affine output head after the product aggregation would likely restore discriminative capacity.

2.5 Experimental Setup

The dataset was split into a fixed 80%/20% train/test partition shared across all methods. No feature standardization or normalization was applied before training; therefore, all methods operated on the raw feature values. Although its interaction with the fixed sigmoid mapping requires careful consideration (see section 4), this approach successfully preserves feature scale. Unless otherwise specified, all evolutionary algorithms used a population size of 500, 2000 generations, and elitism with one elite individual retained across generations; algorithm-specific settings are reported in table 1. Each GP-based configuration and each ML baseline were evaluated over 20 independent runs with distinct random seeds. This ensures a uniform evaluation protocol across all methods, including those with stochastic and deterministic components, and allows all reported metrics to be computed over the same number of repetitions.

Experimental conditions. Four experimental conditions were evaluated within each experiment set:

1. **Experiment 1 (DARWIN features):** All three GP-based algorithm families were applied to the complete 450-feature DARWIN representation ($8 \times 20 = 160$ runs per set).

Table 1: Algorithm-specific hyperparameters.

Parameter	GP	GSGP	SLIM
Population size	500	500	500
Generations	2000	2000	2000
Max depth	15	—	15
Init depth	6	—	—
p_{xo} (crossover)	0.8	0.0	0.0
p_m (mutation)	0.2	1.0	1.0
p_{inflate}	—	—	0.2
Mutation step (ms)	—	*[0.1]*	*[0.1]*
Crossover operator	subtree	—	—
Mutation operator	subtree	geometric	inflate/deflate
Elites	1	1	1
Fitness function	Binary Cross-Entropy		

2. **Experiment 2 (Hand-Stat features):** The same algorithmic comparison using the 2,150-feature Hand-Stat representation (86 features per task $\times$ 25 tasks).
3. **Experiment 3 (p_{inflate} ablation):** SLIM sensitivity to the inflation probability, with $p_{\text{inflate}} \in \{0.2, 0.5, 0.8\}$; GP and GSGP served as baselines.
4. **Experiment 4 (Initialisation depth ablation):** GP-only study of initial tree depth $d_{\text{init}} \in \{4, 6, 8, 10\}$.

Evaluation metrics and implementation. Classification performance was evaluated on the held-out test set using accuracy, specificity, and F1-score. Model complexity was quantified by tree size (the number of nodes in the final elite) and training time. For each configuration, the mean and standard deviation across independent runs were reported. Evolutionary dynamics were tracked using per-generation elite fitness (mean BCE per sample). All experiments were implemented using the `slim-gsgp` Python library [16], which provides unified implementations of GP-based algorithms. Random seeds were set at the algorithm level to ensure reproducibility.

3 Results

All results are reported as mean values with standard deviations across 20 independent runs, each with a different random seed for each method.

3.1 Experiment 1: Algorithm Comparison on DARWIN Features

The classification performance of all algorithms on the 450 DARWIN kinematic features is summarised in Table 2, including the five ML baselines. Accuracy, specificity, F1-score, and tree size are averaged over 20 seeded runs. GSGP achieved the highest mean accuracy (0.78 ± 0.06), outperforming both GP

Table 2: Classification performance on DARWIN features.

Algorithm	Accuracy	Specificity	F1-Score	Tree Size
GP	0.64 ±0.10	0.33 ±0.22	0.72 ±0.06	215 ±93
GSGP	**0.78** ±0.06	**0.63** ±0.12	**0.80** ±0.06	3181 ±517
SLIM+ABS	0.64 ±0.11	0.31 ±0.15	0.73 ±0.09	114 ±19
SLIM+SIG1	0.60 ±0.11	0.21 ±0.14	0.71 ±0.08	**111** ±21
SLIM+SIG2	0.67 ±0.08	0.37 ±0.13	0.74 ±0.06	127 ±6
SLIM*ABS	0.50 ±0.07	0.00 ±0.00	0.67 ±0.06	126 ±7
SLIM*SIG1	0.50 ±0.07	0.00 ±0.00	0.67 ±0.06	123 ±8
SLIM*SIG2	0.50 ±0.07	0.00 ±0.00	0.67 ±0.06	134 ±7
SVM	0.70 ±0.07	0.67 ±0.09	0.71 ±0.08	—
GNB	0.65 ±0.08	0.66 ±0.10	0.65 ±0.09	—
LR	0.70 ±0.07	0.65 ±0.08	0.70 ±0.07	—
LDA	0.67 ±0.06	0.64 ±0.08	0.68 ±0.07	—
KNN	0.63 ±0.07	0.59 ±0.11	0.64 ±0.09	—

(0.64 ± 0.10) and all SLIM variants. Within the SLIM variants, a clear distinction was observed between the addition and multiplication operator families. The addition variants achieved meaningful classification performance (SLIM+SIG2: 0.67 ± 0.08; SLIM+ABS: 0.64 ± 0.11), whereas the multiplication variants exhibited near-chance accuracy (~ 0.50), predicting all instances as positive.

The distribution of classification metrics across runs is shown in fig. 1. The GSGP distributions are both higher and tighter than those of GP and SLIM, with notably stronger specificity. SLIM+SIG2 is the highest-performing SLIM variant, achieving GP-level accuracy while using smaller trees. Among the protocol-matched conventional baselines, SVM and logistic regression achieved the strongest ML performance on the DARWIN feature representation, whereas GSGP achieved the highest mean accuracy across all evaluated methods on this fixed split. The fitness convergence over generations is illustrated in fig. 2. GSGP converges fastest and reaches the lowest mean training fitness, followed by SLIM and GP. GP exhibits the widest variance band, reflecting greater sensitivity to random initialization.

Model complexity and training time. GSGP produced the largest trees (3181 ± 517 nodes) but trained in only 284 ± 6 s, an order of magnitude faster than GP (2104 ± 530 s). All SLIM variants produced compact models (111–134 nodes) and trained in 76–118 s (fig. 3).

3.2 Experiment 2: Hand-Crafted Statistical Features

The same algorithmic comparison was repeated on the 2150-dimensional HandStat features; results are reported in table 3, including the five conventional

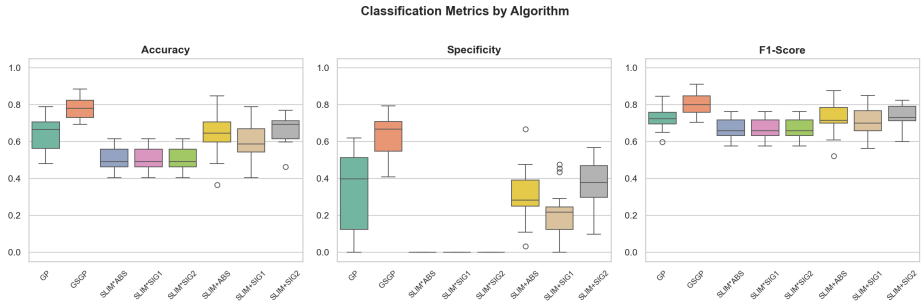


Fig. 1: Classification metrics on DARWIN features (boxes: IQR; whiskers: 1.5×IQR; points: outliers).

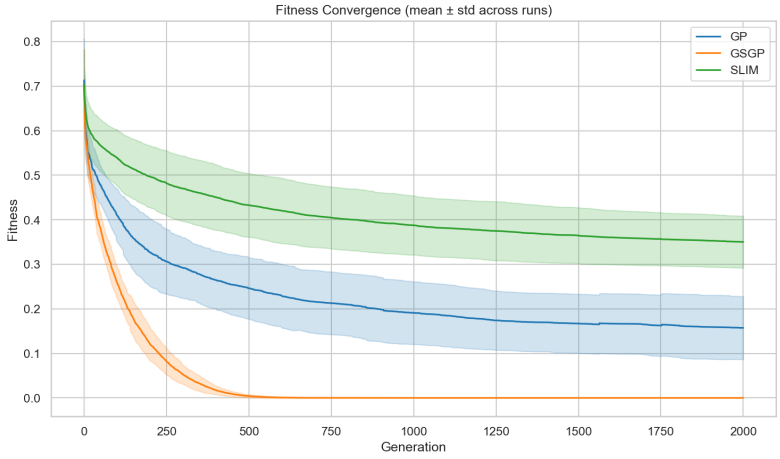


Fig. 2: Elite fitness convergence on DARWIN features over 2000 generations.

baselines retrained on this representation under the same fixed-partition protocol. All algorithms performed worse than on the DARWIN features. GSGP again achieved the highest accuracy (0.60 ± 0.06) but fell far below its DARWIN performance, GP dropped to near-chance (0.51 ± 0.06), and the SLIM $\times$ variants again collapsed to trivial classifiers.

The performance degradation across all algorithms suggests that the raw DARWIN kinematic features retain discriminative information that is partially lost through stroke-level statistical aggregation. The higher dimensionality of the Hand-Stat representation ($p = 2{,}150$ vs $p = 450$) relative to the sample size ($N = 174$) may further exacerbate overfitting and hinder evolutionary search. GSGP's specificity halved (from 0.63 to 0.46), and GP's specificity dropped to near-zero (0.07), indicating that both algorithms struggled to correctly identify healthy controls under this feature representation. The protocol-matched conventional baselines also exhibit a decline under the Hand-Stat representation,

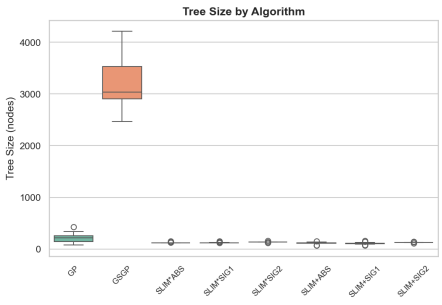
Table 3: Classification performance on Hand-Stat features.

Algorithm	Accuracy	Specificity	F1-Score	Tree Size
GP	0.51 ±0.06	0.07 ±0.09	0.66 ±0.06	119 ±70
GSGP	**0.60** ±0.06	**0.46** ±0.10	0.64 ±0.07	2825 ±1920
SLIM+ABS	0.55 ±0.08	0.17 ±0.11	**0.68** ±0.06	118 ±10
SLIM+SIG1	0.53 ±0.07	0.08 ±0.07	**0.68** ±0.05	**113** ±3
SLIM+SIG2	0.55 ±0.07	0.17 ±0.08	0.67 ±0.07	123 ±3
SLIM*ABS	0.50 ±0.06	0.00 ±0.00	0.67 ±0.05	122 ±5
SLIM*SIG1	0.50 ±0.06	0.00 ±0.00	0.66 ±0.05	120 ±3
SLIM*SIG2	0.50 ±0.06	0.00 ±0.00	0.66 ±0.05	133 ±0
SVM	0.67 ±0.07	0.63 ±0.09	0.68 ±0.08	—
GNB	0.62 ±0.08	0.62 ±0.10	0.63 ±0.09	—
LR	0.67 ±0.07	0.61 ±0.08	0.67 ±0.07	—
LDA	0.64 ±0.06	0.60 ±0.08	0.65 ±0.07	—
KNN	0.60 ±0.07	0.55 ±0.11	0.61 ±0.09	—

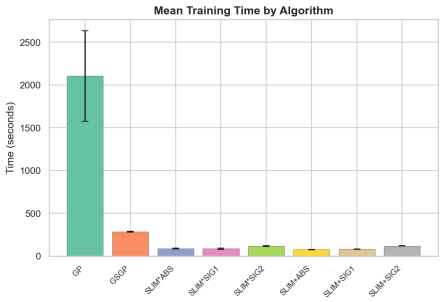
broadly retaining the relative pattern observed on the DARWIN features. The metric distributions and fitness convergence for this experiment are shown in fig. 4.

3.3 Experiment 3: Inflation Probability Ablation

The effect of the inflation probability p_{inflate} on SLIM accuracy, disaggregated by operator family, is reported in table 4. Each cell aggregates three SLIM variants, each evaluated over 20 seeded runs ($n = 60$); the GP (0.64 ± 0.10) and GSGP (0.78 ± 0.06) baselines are provided for reference. Increasing p_{inflate} from 0.2 to 0.5 improved mean accuracy for the + family from 0.64 to 0.70, while the × family remained near-chance regardless of the inflation setting.

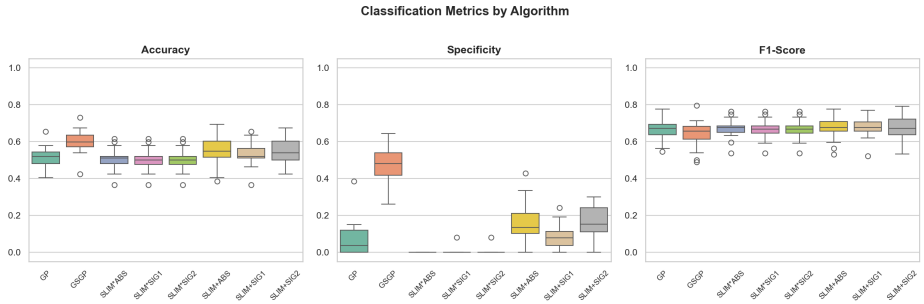


(a) Tree size in nodes (boxes: IQR; whiskers: 1.5×IQR).

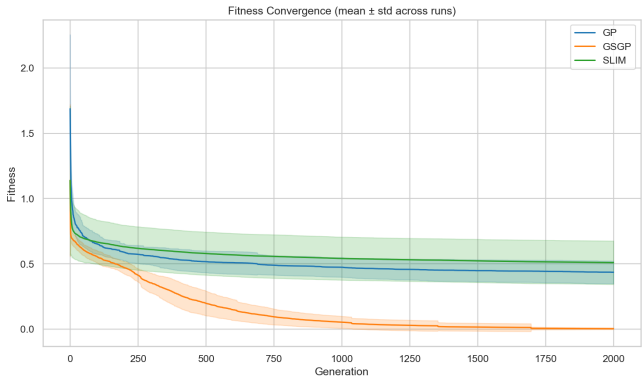


(b) Mean training time in seconds.

Fig. 3: Model complexity on DARWIN features.



(a) Classification metrics across 20 runs (boxes: IQR; whiskers: 1.5×IQR).



(b) Elite fitness convergence over 2000 generations.

Fig. 4: Results on Hand-Stat features (Experiment 2).

The best individual SLIM configuration was SLIM+SIG2 with $p_{\text{inflate}} = 0.5$ or 0.8, achieving a mean accuracy of 0.69 ± 0.08. Overall, $p_{\text{inflate}} = 0.5$ provided the best balance between exploration and parsimony for the + family, while the $\times$ family remained structurally limited under the fixed-threshold pipeline used here. The corresponding convergence behavior is shown in fig. 5.

3.4 Experiment 4: Initialization Depth Ablation

The effect of initial tree depth on GP performance is reported in table 5. Both training and test fitness are expressed as mean per-sample BCE over 20 seeded runs. The default setting ($d_{\text{init}} = 6$) achieved the lowest training fitness (0.16 ± 0.07), while the shallowest setting ($d_{\text{init}} = 4$) produced comparable test fitness with smaller trees. Deeper initializations degraded training fitness and therefore offer little practical advantage in this application.

Table 4: Effect of p_{inflate} on SLIM test accuracy (Experiment 3).

p_{inflate}	**SLIM (+ family)**	**SLIM (× family)**	**Overall**
0.2	0.64 ±0.10	0.50 ±0.07	0.57 ±0.11
0.5	**0.70** ±0.08	0.51 ±0.08	**0.60** ±0.13
0.8	0.68 ±0.07	0.51 ±0.08	0.60 ±0.12

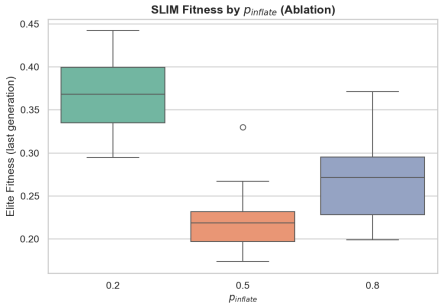
4 Discussion

GSGP delivered the strongest overall performance, reaching 0.78 ± 0.06 accuracy on DARWIN features and approaching the range typically reported for non-interpretable ML classifiers. Its advantage over standard GP is consistent with the smoother search dynamics induced by geometric semantic operators, whereas GP remained more sensitive to initialization and exhibited higher variance across runs. The main negative result is equally informative: the SLIM multiplication family is structurally unsuitable for fixed-threshold binary classification, as its non-negative outputs cannot cross the sigmoid decision boundary. In contrast, addition-family variants—particularly SLIM+SIG2—provided the most favorable accuracy–complexity trade-off, and the ablation study identified $p_{\text{inflate}} = 0.5$ as a robust default setting. Several considerations are important when interpreting these results. Although all methods used 20 seeds on the same partition, variability arises from repeated runs on a single split; consequently, the results should be regarded as descriptive rather than statistically validated. Moreover, the absence of feature standardization may have allowed the fixed sigmoid mapping to interact with heterogeneous feature scales, potentially affecting both calibration and search dynamics. Finally, while SR can, in principle, produce interpretable expressions, practical interpretability depends strongly on model size. The compact SLIM addition-family models are more amenable to inspection than the larger GSGP expressions, indicating that explicit expression-level analysis is required for clinical use. Importantly, the observed degeneracy of the multiplication family is not intrinsic to SLIM, but arises from its interac-

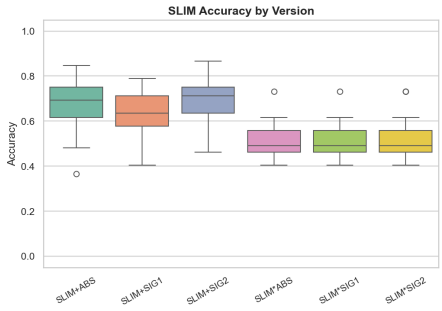
Table 5: GP performance by initial tree depth. All fitness values are **mean per-sample BCE** (lower is better).

d_{init}	**Train Fitness**	**Test Fitness**	**Gap Δ**	**Tree Size (nodes)**
4	0.18 ±0.08	**0.16** ±0.09	−0.02	**184** ±90
6	**0.16** ±0.07	0.17 ±0.08	+0.01	383 ±687
8	0.21 ±0.11	0.17 ±0.06	−0.04	227 ±123
10	0.27 ±0.09	0.19 ±0.03	−0.08	233 ±127

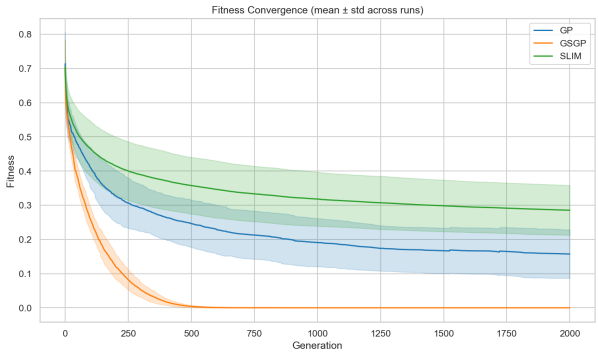
Note: Gap Δ = Test Fitness − Train Fitness (Negative values indicate better test than train performance.)



(a) Elite fitness by p_{inflate} variant



(b) Test accuracy distribution by SLIM variant at each p_{inflate}



(c) Fitness convergence (baseline GP, GSGP, and SLIM)

Fig. 5: Inflation probability ablation (Experiment 3).

tion with a fixed-threshold classification pipeline, highlighting the need for joint design of model structure and decision calibration.

5 Conclusion

This study presents a reproducible comparison of GP, GSGP, and SLIM for AD detection using digitized handwriting. Under a fixed hold-out protocol, GSGP demonstrated the highest predictive performance on DARWIN kinematic features. In contrast, SLIM addition-family variants achieved the most advantageous balance between accuracy and model complexity, resulting in smaller and faster models. A key methodological observation is that SLIM multiplication-family variants become degenerate under fixed-threshold sigmoid classification because their non-negative outputs cannot cross the decision boundary. This phenomenon results from the interaction between model structure and the classification pipeline, rather than from an inherent limitation of SLIM. These results require cautious interpretation, as all metrics are averaged over 20 seeds

on a single hold-out partition and no feature standardization was applied, which may influence both calibration and search dynamics. Overall, the findings suggest that SR approaches are promising for interpretable handwriting-based AD screening, provided that model design and decision calibration are addressed together. Future research should prioritize cross-validation, calibration strategies for outputs, and the explicit analysis of compact symbolic expressions [7].

References

1. Cilia, N.D., De Gregorio, G., De Stefano, C., et al.: Diagnosing Alzheimer's disease from on-line handwriting: A novel dataset and performance benchmarking. Engineering Applications of Artificial Intelligence **111**, 104822 (2022). https://doi.org/10.1016/j.engappai.2022.104822
2. Cilia, N.D., De Stefano, C., Fontanella, F., et al.: An experimental protocol to support cognitive impairment diagnosis by using handwriting analysis. Procedia Computer Science **141**, 466–471 (2018). https://doi.org/10.1016/j.procs.2018.10.141
3. Cilia, N.D., De Stefano, C., Fontanella, F., et al.: An experimental protocol to support cognitive impairment diagnosis by using handwriting analysis. In: Proceedings of the 8th International Conference on Knowledge Discovery and Information Retrieval (KDIR 2018). Procedia Computer Science, vol. 141, pp. 466–471. Elsevier (2018). https://doi.org/10.1016/j.procs.2018.10.141
4. D'Alessandro, T., Carmona-Duarte, C., De Stefano, C., et al.: A machine learning approach to analyze the effects of Alzheimer's disease on handwriting through lognormal features. In: Graphonomics in Human Body Movement. Bridging Research and Practice from Motor Control to Handwriting Analysis and Recognition (IGS 2023). Lecture Notes in Computer Science, vol. 14285, pp. 103–121. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-45461-5_8
5. De Stefano, C., Fontanella, F., Impedovo, D., et al.: Handwriting analysis to support neurodegenerative diseases diagnosis: A review. Pattern Recognition Letters **121**, 37–45 (2019). https://doi.org/10.1016/j.patrec.2018.05.013
6. Erdogmus, P., Ekiz, S.: Alzheimer disease diagnosis from handwriting with DARWIN dataset. In: Proceedings of the International Conference on Advanced Technologies, Computer Engineering and Science (ICATCES). pp. 1–5 (2022)
7. Farinati, D., Pietropolli, G., Vanneschi, L.: Exploring the impact of data scale on mutation step size in SLIM-GSGP. In: Xue, B., Manzoni, L., Bakurov, I. (eds.) Genetic Programming. EuroGP 2025. Lecture Notes in Computer Science, vol. 15609, pp. 35–51. Springer, Cham (2025). https://doi.org/10.1007/978-3-031-89991-1_3
8. Farinati, D., Pietropolli, G., Vanneschi, L.: A study on the dynamics and effectiveness of the deflate geometric semantic mutation. IEEE Transactions on Evolutionary Computation (2025). https://doi.org/10.1109/TEVC.2025.3611226, early Access
9. Impedovo, D., Pirlo, G.: Dynamic handwriting analysis for the assessment of neurodegenerative diseases: A pattern recognition perspective. IEEE Reviews in Biomedical Engineering **12**, 209–220 (2019). https://doi.org/10.1109/RBME.2018.2840679
10. Koza, J.R.: Genetic Programming: On the Programming of Computers by Means of Natural Selection. MIT Press, Cambridge, MA (1992)

11. Livingston, G., Huntley, J., Sommerlad, A., et al.: Dementia prevention, intervention, and care: 2020 report of the Lancet commission. The Lancet **396**(10248), 413–446 (2020). https://doi.org/10.1016/S0140-6736(20)30367-6
12. Makke, N., Chawla, S.: Interpretable scientific discovery with symbolic regression: A review. Artificial Intelligence Review **57**(1), 2 (2024). https://doi.org/10.1007/s10462-023-10622-0
13. Moraglio, A., Krawiec, K., Johnson, C.G.: Geometric semantic genetic programming. In: Coello Coello, C.A., Cutello, V., Deb, K., Forrest, S., Nicosia, G., Pavone, M. (eds.) Parallel Problem Solving from Nature – PPSN XII. Lecture Notes in Computer Science, vol. 7491, pp. 21–31. Springer, Berlin, Heidelberg (2012). https://doi.org/10.1007/978-3-642-32937-1_3
14. Poli, R., Langdon, W.B., McPhee, N.F.: A field guide to genetic programming. Lulu Enterprises (2008), freely available at http://www.gp-field-guide.org.uk
15. Rosenblum, S., Weiss, P.L., Parush, S.: Product and process evaluation of handwriting difficulties. Educational Psychology Review **15**(1), 41–81 (2003). https://doi.org/10.1023/A:1021371425220
16. Rosenfeld, L., Farinati, D., Rasteiro, D., et al.: Slim_gsgp: A Python library for non-bloating GSGP. In: Genetic and Evolutionary Computation Conference (GECCO '25). p. 9. ACM, New York, NY, USA (2025). https://doi.org/10.1145/3712256.3726398
17. Rudin, C.: Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nature Machine Intelligence **1**(5), 206–215 (2019). https://doi.org/10.1038/s42256-019-0048-x
18. Vanneschi, L.: SLIM_GSGP: The non-bloating geometric semantic genetic programming. In: Giacobini, M., Xue, B., Manzoni, L. (eds.) Genetic Programming: 27th European Conference, EuroGP 2024, Held as Part of EvoStar 2024, Aberystwyth, UK, April 3–5, 2024, Proceedings. Lecture Notes in Computer Science, vol. 14631, pp. 125–141. Springer, Cham (2024). https://doi.org/10.1007/978-3-031-56957-9_8
19. Vanneschi, L., Farinati, D., Rasteiro, D., Rosenfeld, L., Pietropolli, G., Silva, S.: Exploring non-bloating geometric semantic genetic programming. In: Winkler, S.M., Banzhaf, W., Hu, T., Lalejini, A. (eds.) Genetic Programming Theory and Practice XXI, pp. 237–258. Genetic and Evolutionary Computation, Springer, Singapore (2025). https://doi.org/10.1007/978-981-96-0077-9_12
20. Werner, P., Rosenblum, S., Bar-On, G., et al.: Handwriting process variables discriminating mild Alzheimer's disease and mild cognitive impairment. The Journals of Gerontology: Series B **61**(4), P228–P236 (2006). https://doi.org/10.1093/geronb/61.4.P228
21. World Health Organization: Dementia in refugees and migrants: epidemiology, public health implications and global responses. World Health Organization, Geneva (2025), https://www.who.int/publications/i/item/9789240102224