AIMS Medical Science

AIMS Medical Science, 13(3): 159–186.
DOI: 10.3934/medsci.2026012
Received: 16 November 2025
Revised: 21 June 2026
Accepted: 14 July 2026
Published: 27 July 2026

http://www.aimspress.com/journal/medicalScience

Research article

Improving substructural analysis with chemical weighted RMSE in symbolic regression

Mohd Isrul Esa*, Nor Samsiah Sani and Salwani Abdullah

Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, Bangi, Malaysia

* **Correspondence:** Email: p102377@siswa.ukm.edu.my.

Abstract: Substructural Analysis (SSA) is a ligand-based virtual screening method that ranks compounds by combining fragment weights, where each weight reflects the contribution of a molecular substructure to the separation of active and inactive compounds. However, conventional SSA weighting schemes are often static and may be less adaptive to heterogeneous and highly imbalanced bioactivity datasets. In this study, we proposed a hybrid SSA framework that used Genetic Algorithms (GA) to optimize fragment-level weights and Genetic Programming (GP) to evolve symbolic weighting equations guided by a Chemical Weighted Root Mean Square Error (CW-RMSE) fitness function. Enrichment factor at the top 1% of the ranked list (EF@1%) was used as the main early-recognition metric because it measured active-compound retrieval relative to random selection in the most practically relevant part of a screening list. The framework was evaluated using 15 ChEMBL-derived activity classes represented by molecular fingerprints. SSA-GP achieved higher median EF@1% than GA-based SSA in 14 of 15 classes, with statistically significant improvements in 13 of these classes using the Mann-Whitney U test ($p < 0.05$). Notable median EF@1% gains were observed for RNN (16.98 to 69.34), MMP1 (36.67 to 83.33), 5HT1A (3.13 to 46.25), and FXA (15.83 to 57.91), while AT1 was the exception in which GA performed better. These results indicated that SSA-GP, guided by the CW-RMSE fitness function, could improve early active-compound prioritization while retaining the interpretability advantage of symbolic SSA models. Further validation against non-SSA machine-learning baselines and expert chemical interpretation of the generated equations is recommended.

Keywords: virtual screening; substructural analysis; genetic programming; genetic algorithm; chemoinformatics; drug discovery

Abbreviations: ACT: active-compound count; AP: average precision; AUC: area under the receiver operating characteristic curve; CW-RMSE: chemical weighted root mean square error; EF: enrichment factor; GA: genetic algorithm; GP: genetic programming; INACT: inactive-compound count; LBVS: ligand-based virtual screening; MACCS: Molecular ACCess System; N: number of observations; NACT: number of active compounds; NINACT: number of inactive compounds; RMSE: root mean square error; SSA: substructural analysis; TOT: total-compound count

1. Introduction

Drug discovery is a long and costly process in which early screening determines how efficiently promising chemical matter can be prioritized for experimental testing. In silico virtual screening has therefore become an important component of modern discovery pipelines because it can rank large chemical libraries before more expensive in vitro assays are performed [1–8]. Within this setting, ligand-based virtual screening (LBVS) is particularly useful when the three-dimensional structure of the biological target is unavailable or incomplete [9,10].

High-throughput screening (HTS) remains an important experimental approach for identifying hit compounds from large chemical libraries, but it is often associated with substantial cost, experimental workload, and false-positive results [1–3]. Virtual screening (VS) complements HTS by computationally ranking compounds before laboratory testing, thereby reducing the number of candidates that need to be evaluated experimentally [4,5]. Within ligand-based virtual screening, Substructural Analysis (SSA) is particularly useful because it represents compounds through chemically interpretable molecular fragments or fingerprint bits.

SSA is one of the earliest machine-learning approaches used in cheminformatics and remains relevant because it represents compounds through molecular fragments or fingerprint bits that are chemically interpretable [11–15]. In SSA, each compound is encoded as a binary vector, and each fragment is assigned a weight derived from its occurrence among active and inactive compounds. The compound score is then calculated by combining the weights of the fragments present in the compound, after which compounds are ranked for screening.

Work has shown that Genetic Algorithms (GA) can improve SSA by directly optimizing fragment weights instead of relying only on fixed statistical weighting schemes [15,16]. Nevertheless, GA-based SSA still operates within a fixed scoring structure and may not fully capture nonlinear relationships among fragments, especially in heterogeneous activity classes and highly imbalanced screening datasets where active compounds form a small minority of the library.

In this study, we address this limitation by introducing a Chemical Weighted Root Mean Square Error (CW-RMSE) fitness function within a Genetic Programming (GP) symbolic-regression framework. GA-derived fragment weights are used as chemically informed inputs, while GP evolves symbolic equations that can model nonlinear and context-dependent relationships between fragment statistics and activity. Our aim is not to replace all virtual-screening methods, but to improve the SSA weighting mechanism while preserving its fragment-level interpretability.

The major contributions of this study are: (i) A hybrid GA-GP SSA framework for evolving fragment-weighting equations; (ii) the CW-RMSE fitness function to guide GP toward chemically informative and imbalance-aware weighting behavior; and (iii) an empirical evaluation on 15 ChEMBL-derived activity classes using EF@1% as the primary early-recognition metric, supported by AUC, Average Precision, Recall@1%, and non-parametric statistical testing.

2. Materials and methods

2.1. Substructural analysis workflow

The SSA workflow used in this study is illustrated in Figure 1. The fragments of interest were not manually selected; they corresponded to the 166 molecular fingerprint bits generated for every compound. A value of 1 indicated that a fragment was present in a compound, whereas 0 indicated an absence. For each activity class, fragment statistics were first computed from the training set. Fragment weights were then learned by GA or generated through GP equations. The score for a test compound was obtained by summing the weights associated with the fragments present in that compound, and the test compounds were ranked in descending order of their scores. Active retrieval was then evaluated at the top of the ranked list.

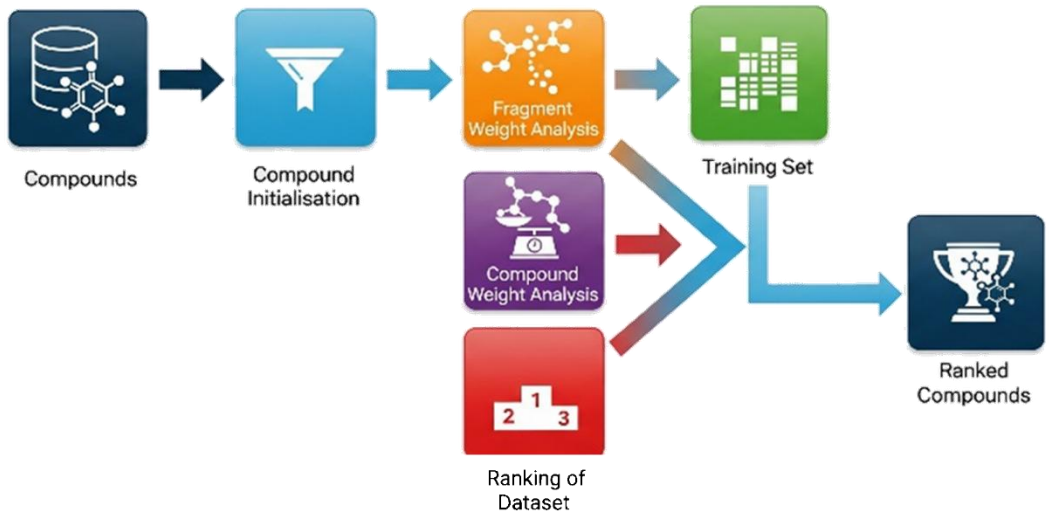

Figure 1. Overview of the SSA workflow used in this study.

2.2. Evolutionary optimization framework

Evolutionary algorithms (EAs) are population-based optimization methods inspired by variation, selection, and survival. In this study, GA and GP were used for different but complementary tasks. GA represents each candidate solution as a vector of fragment weights and searches for weight combinations that retrieve more active compounds near the top of the ranked list. GP represents each candidate solution as a symbolic expression tree and evolves mathematical equations that transform fragment statistics into weights. Thus, GA provides direct numerical weight optimization, whereas GP provides an interpretable symbolic model capable of capturing nonlinear relationships.

The proposed hybrid framework uses GA-derived weights as chemically informed guidance for GP. In this framework, CW-RMSE guides the GP search toward symbolic equations that assign greater emphasis to errors associated with chemically informative fragments during evolution.

2.3. GA in SSA

The GA baseline followed the SSA weighting approach developed by Holliday et al. [15]. Each chromosome encoded a candidate vector of fragment weights. For a molecule, the weighted score was calculated from the fingerprint bits that were present in that molecule. The molecules were then ranked, and the chromosome fitness was determined by the number of active compounds retrieved in the top-ranked subset. This GA model served as the SSA reference method against which the SSA-GP framework was evaluated. GA-based strategies have also been applied in molecular discovery and generative molecular design, supporting the relevance of GA for exploring chemical search spaces [17].

Selection, crossover, and mutation were then applied iteratively to improve the fragment-weight vectors. The final GA-derived weights were also used to construct the chemical weighting component of CW-RMSE for the GP experiments. Developments in evolutionary computation have introduced alternative selection and adaptation mechanisms that may further improve GA performance [18,19].

2.3.1. GA fitness function

The GA fitness function, adapted from Holliday et al. [15], is a single-objective, rank-based function. It counts the number of active molecules retrieved within a selected early part of the ranked list. EF@1% was selected because early recognition is central to virtual screening, where only a small portion of a ranked list is normally advanced for experimental testing. The use of the 1% cut-off also enables direct comparison with the SSA-GA reference protocol used as the baseline in this study.

The fitness function is mathematically represented as follows (Eq 1):

$$Fitness = \sum_{i=k\ in}^{n} Data.Active\ [i] \tag{1}$$

where: i is the index of a molecule in the ranked list; k is the threshold index for the top X per cent of molecules; n is the total number of molecules in the dataset; and Data. Active [i] represents the activity status (active or inactive) of the molecule at position i.

The key features of the fitness function are as follows:

1. Single-Objective Focus: This function simplifies the evaluation by focusing only on the number of active molecules in the top-ranked subset. This maximized the retrieval of active compounds during virtual screening.
2. Rank-Based Evaluation: Ranking molecules according to their weighted scores ensures that the most promising candidates (those with the highest scores) are prioritized.
3. Threshold Selection: Parameter k sets the cut-off for the top X percentage of molecules, enabling the function to focus on the most relevant part of the dataset.

We chose this fitness function because it directly measures how well a chromosome identifies the active molecules. By optimizing this metric, the GA can repeatedly improve the weights of molecular fragments, thereby improving the model's ability to distinguish between active and inactive compounds.

2.3.2. GA parameterization

The GA settings in Table 1 were retained primarily to reproduce a comparable SSA-GA baseline based on the reference methodology, and not to claim that roulette-wheel selection or the selected

operators that represent the most recent GA design. More recent selection and adaptation mechanisms may further improve GA performance, but their inclusion would change the baseline and is therefore reported as future work [17–19].

The parameters for the GA were based on the methods described in a previous study by Holliday et al. [15]. These parameters were carefully chosen through systematic experiments and proven effective in cheminformatics. The details are listed in Table 1.

Table 1. GA parameter setting.

Parameter	Chosen value	Description/Justification
Parent selection	Roulette wheel	Ensures proportional selection based on fitness, emphasizing chromosomes with higher fitness.
Crossover procedure	One-Point	Simple yet effective, enabling recombination of parent solutions to generate diversity.
Crossover rate	0.95	A high rate ensures sufficient mixing of genetic material for better exploration of solutions.
Mutation rate	0.01	A low mutation rate was used to introduce limited diversity while reducing disruptive changes to fragment-weight solutions.
Population size	200	Balances diversity with computational efficiency, ensuring a manageable solution space.
Number of iterations	500	A fixed iteration budget was used to balance search depth, comparability, and computational cost.
Mutation constraints	Constrained	Ensures fragment weights maintain logical consistency with prior calculations.

2.3.3. GA pseudocode

As shown in the pseudocode in Figure 2, to calculate the fitness score of a chromosome, we first applied the chromosome (random fragment weights) to the molecular dataset. We then added the weighted presence of each fragment to obtain the total score for each molecule. Next, we ranked the molecules from highest to lowest scores. Finally, the number of active molecules in the top 1% of the ranked list was determined.

The fitness score for each chromosome, which quantifies its ability to predict molecular activity, was calculated using a three-step process. First, a score was computed for each molecule in the dataset. This score was determined by the dot product of the chromosome fragment weight vector and the corresponding binary fragment presence vector of a molecule. Subsequently, all molecules were ranked in descending order based on their scores. The final fitness score was defined as the total number of known active molecules in the top 1% of the ranked list. A higher fitness score indicated stronger top-1% active-compound enrichment under this GA objective but did not imply better performance across all screening metrics.

Algorithm 1: Main GA Program for Generating Molecule Weights

Input: A dataset of molecules with associated fragment data.
Output: An optimized weight set for each fragment.

```
Initialize a population of chromosomes, each containing randomly generated fragment weights.
For each generation, perform Steps 3–14.
    For each chromosome in the population, perform Steps 4–12.
        For each molecule in the dataset, perform Steps 5–10.
            Initialize the molecule score to 0.
            For each fragment in the molecule, perform Steps 7–8.
                Multiply the fragment-presence value by the chromosome's
                weight for that fragment.
                Add the resulting value to the molecule score
            End the fragment loop.
        End the molecule loop.
        Sort the molecules in descending order of their scores.
        Calculate fitness as the number of active molecules within the
        top X% of the ranked list.
    End the chromosome loop.
    Apply selection, crossover, and mutation according to chromosome
    fitness to generate a new population.
End the generation loop.
Return the best fragment-weight set from the final generation.
```

Figure 2. Pseudocode of GA.

2.4. GP methodology

GP is a highly flexible framework that uses mathematical functions and operators to represent symbolic expressions [20]. We chose GP as our modeling technique because it can automatically create understandable mathematical models while effectively capturing complex nonlinear relationships in molecular datasets [21]. Unlike traditional optimization and machine learning methods, which often rely on fixed features or unclear models, GP dynamically develops symbolic expressions, providing more flexibility in identifying important structure-activity relationships (SAR) [22]. This adaptability enables the GP to evolve tree-like expressions, enabling it to effectively capture complex patterns and relationships in drug discovery datasets.

GP is especially useful for handling the diverse and high-dimensional nature of molecular datasets, where it is difficult to define clear mathematical relationships between molecular fragments and biological activity [23]. The nonlinear nature of GP models enables them to effectively handle complex chemical interactions and variations in molecular activity data [24]. This makes GP especially beneficial compared with traditional regression techniques or neural networks, which often require extensive feature engineering or are difficult to interpret [25]. In addition, the ability of the GP to evolve the model structure and its parameters simultaneously enables it to explore many possible solutions, making it suitable for problems where the functional form of the solution is not known in advance [26].

The GP method begins by randomly creating a population of symbolic expressions, each representing a possible solution [27]. Through repeated generations, GP uses genetic operations,

including selection, crossover, and mutation to improve the population gradually. Population diversity and inheritance are important considerations in GP because they influence the exploration of symbolic-regression search spaces [28]. These operations ensure that symbolic expressions that are better at modeling molecular activity are more likely to be improved further [23]. The evolutionary process was continued until the predefined stopping criterion or maximum generation limit was reached, and the best-performing symbolic expression was retained.

Using GP, the approach ensures that the developed models not only predict well but also provide useful insights into molecular interactions. A GP's ability to create expressions that humans can read further improves its usefulness in drug discovery, where clarity and understandability are important for decision-making [22]. These strengths made GP a good choice for this study, ensuring that the resulting models help with prediction accuracy and a better understanding of the structure-activity relationships.

2.4.1. GP symbolic regression

GP goes beyond GA by evolving symbolic mathematical expressions instead of fixed weights [27]. This change enables the GP to find nonlinear relationships between molecular features that depend on context, overcoming the limitations of the linear weighting of traditional SSA [22]. Guided by the CW-RMSE fitness function, GP improves the models to reduce chemically relevant errors while increasing the prediction accuracy. By focusing on the evolutionary process of biological relevance, GP aligns with activity patterns that are often difficult to observe in datasets with uneven amounts of active and inactive compounds. The key innovations of GP in Cheminformatics are as follows:

1. Tree-structured Expression Representation: GP represents solutions as tree structures that combine variables (such as fragment frequencies) and mathematical operations [28].
2. Adaptive Model Evolution: GP changes the complexity of its equations based on variations in the dataset, such as different amounts of active and inactive compounds. This may reduce sensitivity to class imbalance, although overfitting remains possible without fully nested validation [29].
3. Interpretable Symbolic Regression: Unlike "black-box" machine learning models, GP creates equations that humans can read, linking molecular substructures to the activity. These equations provide useful information regarding the possible fragment-weighting patterns associated with activity-class separation and the design of new molecules [30].

The ability of GP to produce understandable models while handling complex, nonlinear data makes it a valuable tool in cheminformatics, especially for understanding structure–activity relationships and solving unique challenges in drug discovery research [31,32].

2.5. Synergy between GA and GP

The combination of GA and GP creates a systematic approach that improves the power and understanding of SSA in virtual screening. By combining their strengths, this collaboration effectively reduces the weaknesses of traditional methods while balancing the computational efficiency and model complexity.

In this framework, the GA acts as the base by optimizing the fixed fragment weights through repeated selection, crossover, and mutation. GA identifies molecular fragments strongly linked to biological activity, particularly focusing on less common active compounds in unbalanced datasets. For example, GA might identify sulfonamide groups and aromatic rings as key fragments in protease inhibitor datasets. These optimized weights provide a chemically relevant starting point for further improvements.

Building on GA's results, GP acts as an advanced modeling layer, developing symbolic mathematical expressions that show nonlinear relationships between context-dependent fragments. Unlike fixed weights, GP-derived equations capture complex interactions, explaining how molecular fragments work together to affect biological activities.

The interaction between GA and GP lies in their complementary strengths. GA's optimization narrows the search by highlighting chemically important fragments, enabling GP to focus its computational power on modeling the interactions of these key features. This step-by-step method reduces the computational load while improving the accuracy and interpretability of the resulting models. GA effectively performs the "rough tuning" of weights, while GP does the "fine-tuning" by finding complex relationships and creating equations that match real biological data.

In addition, this framework effectively addresses the common problem of imbalanced datasets often found in virtual screenings. The CW-RMSE metric guides the GP search toward chemically weighted errors associated with informative fragments. Subsequently, GP continued this focus and developed equations that are sensitive to underrepresented active classes. Together, GA and GP reduce the bias toward inactive compounds, leading to models that are less biased and chemically relevant.

In addition to improving computational efficiency, the combination of GA and GP greatly increases the understandability and accuracy of cheminformatics models. GA provides a basic understanding of important molecular fragments, whereas GP builds on this by creating easy-to-read equations that link these fragments to the activity. By using GA for efficient weight optimization and GP for creating dynamic equations, this framework accelerates virtual screening and provides a flexible and clear method for identifying new drugs.

2.6. Experimental setup

In this study, we developed and evaluated a SSA-GP framework using CW-RMSE as the internal fitness function for improving SSA fragment weighting in virtual screening. For the experimental design, we compared the proposed SSA-GP model with the SSA-GA baseline using the same activity classes, molecular representation and train-test split, so that differences in performance could be attributed to the weighting and fitness mechanisms.

2.6.1. Dataset

The molecular data were obtained from ChEMBL, a curated public database of compounds and bioactivity measurements used widely in cheminformatics and drug discovery research [33,34]. The dataset was constructed using *Homo sapiens* targets with ChEMBL confidence score 9. A compound was labeled active for a given activity class when the curated

activity value met the threshold pChEMBL or curated potency value ≥5.0. Compounds that did not meet this class-specific active definition were used as the background inactive set for that one-vs-rest screening task. These inactive/background compounds are therefore not computational decoys and should not be interpreted as experimentally confirmed inactive against all possible targets. This definition provides a clearer basis for interpreting the reported screening results and their practical scope within the present one-vs-rest evaluation setting.

Fifteen activity classes were selected: COX, 5HT, 5HT1A, 5HT3, ACHE, AT1, D2, FXA, HIVP, MMP1, PDE4, PKC, RNN, SUBP, and THRM (Table 2). The selection was guided by three criteria: (i) Established relevance in drug discovery studies; (ii) sufficient curated active compounds to support model training and testing; and (iii) alignment with the target classes used in the SSA-GA reference study, enabling a controlled methodological comparison [15].

A stratified 90%/10% train-test split was used for each activity class. The 90% training portion was selected to retain enough active examples for GP evolution in low-hit classes such as AT1, COX, 5HT3, PDE4, and PKC, while the 10% held-out test set still contained approximately 10,000 compounds per class for final evaluation. Stratification preserved the active/inactive ratio in both subsets. The final results reported in the Results section were calculated on the held-out test sets, and ten independent runs were used to account for stochastic variation.

Table 2. Summary of the train and test dataset.

Class activity	Number of active compounds	Train dataset active compounds	Train dataset inactive compounds	Test dataset active compounds	Test dataset inactive compounds
COX	139	118	89,882	21	9979
5HT	2448	2217	87,783	231	9769
5HT1A	1479	1319	88,681	160	9840
5HT3	213	194	89,806	19	9981
ACHE	724	665	89,335	59	9941
AT1	106	100	89,900	6	9994
D2	1858	1662	88,338	196	9804
FXA	1502	1363	88,637	139	9861
HIVP	2157	1951	88,049	206	9794
MMP1	395	350	89,650	45	9955
PDE4	254	227	89,773	27	9973
PKC	211	192	89,808	19	9981
RNN	982	876	89,124	106	9894
SUBP	847	765	89,235	82	9918
THRM	838	752	89,248	86	9914

2.6.2. GP fitness function

In GP, the fitness function evaluates the quality of a symbolic expression tree and guides the evolutionary search toward better solutions [35]. In this study, GP was used to generate fragment-weighting equations for SSA; therefore, the fitness function must reflect prediction error and the chemical relevance of the fragments contributing to that error.

Root Mean Square Error (RMSE) provides a conventional measure of prediction error, but by itself it treats all errors uniformly. For highly imbalanced virtual-screening data, uniform error treatment may under-emphasize rare but informative active-associated fragments. CW-RMSE addresses this by scaling the squared error using chemical weights derived from the GA fragment-weighting process.

The CW-RMSE fitness function therefore encourages GP to evolve symbolic equations that minimize prediction error while assigning greater importance to chemically informative fragments. This design is intended to improve early enrichment without removing the interpretability of SSA, because the generated equations remain expressed in terms of fragment-level statistics.

We incorporated CW-RMSE as the fitness function within the GP framework. CW-RMSE extends conventional RMSE by applying molecule- or fragment-derived chemical weights so that errors linked to more informative fragments contribute more strongly to the GP fitness value.

The CW-RMSE is calculated as follows (Eq 2):

$$\mathrm{CW-RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} w_i \cdot (y_i - \widehat{y_i})^2} \tag{2}$$

where: i indexes the molecule or data point being evaluated; n is the total number of data points used in the CW-RMSE calculation; w is the chemical weight associated with the i-th data point, derived from the normalized GA fragment-weight contribution, and $y_i, \widehat{y_i}$ represent the observed and predicted activity values, respectively.

In this formulation, (n) denotes the number of observations, whereas (w_i) represents the chemical weighting term assigned to the (i)-th observation. The weighting term does not replace the observed activity label; rather, it adjusts the contribution of each error term so that the GP search is more sensitive to chemically informative fragments and less dominated by the majority inactive or background class.

2.6.3. Chemical weight

Normalization is applied to the GA-derived fragment weights W_i so that all weights are placed on a comparable scale before being used in CW-RMSE. This prevents fragments with extreme raw weights from dominating the symbolic-regression process while preserving their relative contribution to active-compound enrichment.

One common normalization method is the Z-score, where each weight is normalized using the following formula:

$$[w_i' = \frac{w_i - \mu}{\sigma}] \tag{3}$$

In this study, Eq 3 is used to transform the raw GA-derived fragment weights into normalized chemical weights. For a compound, the relevant normalized weights are then aggregated over the fragments present in its fingerprint to obtain the chemical weighting term used in CW-RMSE.

Table 3 summarizes the normalized GA-derived chemical-weight signal for each activity class. These values indicate the direction and relative magnitude of the class-level weighting information supplied to the GP fitness function. Positive values suggest a stronger positive weighting signal, whereas negative values indicate a weaker or penalized weighting signal within the corresponding activity class.

Table 3. List of normalized weights from GA.

Activity class	Normalised weight
COX	−0.05478
5HT	0.013102
5HT1A	−0.06035
5HT3	−0.08216
ACHE	0.034704
AT1	0.032084
D2	0.017418
FXA	0.028491
HIVP	0.041556
MMP1	0.019712
PDE4	−0.02224
PKC	−0.19322
RNN	−0.02436
SUBP	0.004225
THRM	0.041539

2.6.4. GP parameterization

Before GP execution, the terminal and function sets were defined. The terminal set consisted of ACT, INACT, TOT, N, NACT, and NINACT, which represent fragment-level counts or totals derived from the training data. Specifically, ACT denotes the number of active training compounds containing a given fragment, INACT denotes the number of inactive/background training compounds containing the fragment, TOT represents the total number of training compounds containing the fragment, N denotes the total number of training compounds in the activity class, and NACT and NINACT refer to the total numbers of active and inactive/background compounds in the training set, respectively.

The GP parameterization stage was treated as a sensitivity analysis on two structurally contrasting activity classes, COX and RNN. The tested ranges included population size, maximum iteration, tree construction method, parent selection, crossover rate, and mutation rate, as summarized in Table 4. Because the parameter study was conducted as a sensitivity analysis rather than a fully nested cross-validation protocol, potential parameter-selection bias remains a limitation of this study.

Table 4: Performance of genetic programming parameter settings based on number of active compounds retrieved in the top 1% for RNN and COX classes.

Parameter	Value	RNN	COX
Population size	100	6	11
	200	14	15
	300	10	6
	400	12	16
	500	5	0
Iteration	50	5	16
	100	14	13
	200	8	9
	300	12	5
Tree construction	Grow	14	16
	Full	12	9
	Ramped half-and-half	6	8
Parent selection	Roulette wheel	14	16
	Tournament	11	13
	Random	6	9
Crossover rate	0.6	14	13
	0.7	9	16
	0.8	11	9
	0.9	8	5
Mutation rate	0.1	5	9
	0.2	11	16
	0.3	14	11
	0.4	8	5
Number of nodes	20	7	10
	25	9	13
	30	12	15
	35	14	16
	40	11	14
Depth of tree	5	6	9
	10	9	12
	15	12	14
	20	8	6

COX and RNN were retained because they represent contrasting structural-diversity profiles in the reference SSA literature. The selected settings were subsequently fixed for the reported comparison so that SSA-GA and SSA-GP could be evaluated under consistent conditions. No guarantee of convergence was claimed; instead, repeated runs and the stability of enrichment behavior were used to assess robustness.

For each parameter value, five independent GP runs were performed, while the remaining parameters were kept at their baseline settings. The number of active compounds retrieved from the top 1% of the ranked list was recorded, and the best-performing value was selected as a

practical empirical setting for the subsequent experiments. This procedure was intended for parameter selection rather than as a formal proof of convergence or global optimality.

Table 4 reports the sensitivity of GP performance to the tested parameter values. Parameters that produced comparatively higher early active retrieval across the two representative activity classes were selected for the final configuration. This tuning procedure was used to identify a practical empirical setting for subsequent experiments rather than to establish formal convergence or global optimality.

The selected values are summarized in Table 5 (COX for the homogeneous class) and Table 6 (RNN for the heterogeneous class).

Table 5: Optimal genetic programming parameters for the COX activity class.

Parameter	Selected parameter
Population size	400
Maximum iteration	50
Tree construction method	Grow
Parent selection method	Roulette wheel
Crossover rate	0.7
Mutation rate	0.2

Table 6: Optimal genetic programming parameters for the RNN activity class.

Parameter	Selected parameter
Population size	200
Maximum iteration	100
Tree construction method	Grow
Parent selection method	Roulette wheel
Crossover rate	0.6
Mutation rate	0.3

2.6.5. GP pseudocode

Figure 3 presents a pseudocode illustration of the procedural details of the Genetic Programming (GP) algorithm, providing a step-by-step overview of the evolutionary process used in this study.

The algorithm first used GA to search for fragment-weight vectors that improve early active retrieval. The best GA-derived weights were then normalized and used as chemical weighting information for GP.

In the second phase, GP evolved symbolic equations using CW-RMSE as the fitness function. The GP process continued until the predefined stopping criterion was reached. Because evolutionary algorithms did not guarantee convergence to a global optimum, the results are reported over multiple independent runs rather than as a single deterministic solution.

Algorithm 2: Combined GA and GP procedure

Input: A dataset of chemical compounds with molecular features and activity labels.
Output: An optimized symbolic model for predicting molecular activity.

Start the GA stage.
    Initialize the population of fragment weights.
        Create a population with randomly generated fragment weights
    Repeat the following steps.
        For each chromosome in the population, perform Steps 6–12.
            Calculate fitness as follows:
                For each molecule, perform Step 8.
                    Compute the molecule score as the sum of the weighted fragment-presence values.
                End the molecule loop.
                Sort the molecules by score and count the active molecules within the top 1%.
                Assign fitness based on the number of active molecules retrieved
        End the chromosome loop.
        Apply selection, crossover, and mutation to generate new fragment weights.
    Continue until the predefined stopping criterion is met.
    Retain the best set of fragment weights.
End the GA stage.
Start the GP stage.
    Initialize the population of symbolic expressions.
        Create a population with randomly generated symbolic expressions
    Repeat the following steps.
        For each expression in the population, perform Steps 22–25.
            Evaluate the expression using CW-RMSE as follows:
                Apply the GA-derived weights to calculate CW-RMSE for the expression.
                Assign fitness based on the CW-RMSE value.
        End the expression loop.
        Apply selection, crossover, and mutation to generate new expressions.
    Continue until the predefined stopping criterion is met.
    Select and retain the best symbolic expression.
End the GP stage.

Figure 3. Combined GA and GP procedure.

2.6.6. Evaluation of virtual screening quality

Virtual-screening performance was evaluated primarily using enrichment factor (EF), a rank-based metric that compares the proportion of active compounds retrieved in a selected top fraction of the ranked list with the proportion expected from random selection [36]. EF@1% was selected as the main metric because practical virtual screening often focuses on the earliest part of a ranked list, where only a small number of compounds may be selected for follow-up testing.

The EF was calculated after screening the top 1% of the ranked library using the following formula (Eq 4):

$$\mathrm{EF} = \frac{(Nexperimental/Nexptotal)}{(Ntheoretical/Ntotal)} \quad (4)$$

where Nexperimental is the number of active compounds found within the selected top-ranked subset; Nexptotal is the total number of compounds in the selected top-ranked subset; Ntheoretical is the total number of active compounds in the full test set; and Ntotal is the total number of compounds in the full test set.

To complement EF@1%, three additional metrics were used. The Area Under the ROC Curve (AUC) measured overall discrimination between active and inactive/background compounds across thresholds. Average Precision (AP) summarized the precision-recall curve and was informative for imbalanced datasets. Recall@1% reported the percentage of all active test compounds retrieved within the top 1% of the ranked list. These metrics are discussed together with EF results in the Results and discussion section.

The EFs ranged from a minimum of 1 (random sorting, no significant enrichment) to values up to 100 (considerable enrichment compared to random screening). An EF above 10 indicated that the virtual screening method significantly reduced the time, cost, and experimental effort required, making it a valuable tool for prioritizing compounds for testing.

3. Results and discussion

3.1. *Substructural analysis for the GA result*

In this study, we introduced a novel approach for identifying chemical compounds with potential activity against specific biological targets. We employed a GA to assign weights to 166 common molecular fragments. These weights were aggregated to calculate the total score for each compound, enabling the ranking of compounds from highest to lowest rank. The number of active compounds in the top 1% of the ranked list was recorded.

Table 7 summarizes the EF results for SSA-GA across 15 activity classes. The performance of SSA-GA varied substantially, with several activity classes showing strong enrichment. AT1 achieved the highest median EF of 83.33 (SD = 11.50), followed by THRM (median = 60.36, SD = 9.31) and PKC (median = 53.60, SD = 6.48). These results suggest that the GA-based method may be effective for prioritizing active compounds in activity classes where fragment-weight patterns are more readily captured by direct GA optimization.

Table 7. Enrichment factors for fifteen (15) activity classes in the top 1% active compounds retrieval for SSA-GA.

Activity class	Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Run 8	Run 9	Run 10	Median	Standard deviation
COX	9.52	14.29	28.57	33.33	19.05	19.05	9.52	4.76	4.76	23.81	16.67	9.34
5HT	9.00	12.38	6.75	8.16	9.00	7.88	9.29	9.85	9.00	8.44	9.00	1.40
5HT1A	3.75	1.88	3.75	5.00	1.25	2.50	5.00	2.50	1.25	5.00	3.13	1.44
5HT3	42.11	42.11	10.53	52.63	15.79	21.05	15.79	36.84	15.79	36.84	28.95	13.97
ACHE	18.64	8.47	16.95	13.56	8.47	13.56	13.56	10.17	11.86	8.47	12.71	3.40
AT1	76.67	83.33	83.33	100.00	83.33	83.33	76.67	76.67	95.83	95.83	83.33	11.50
D2	9.18	14.29	11.22	5.61	9.18	12.76	7.65	15.31	7.14	15.31	10.20	3.35
FXA	24.46	12.95	20.14	18.71	15.11	16.55	12.23	12.23	10.07	24.46	15.83	4.87
HIVP	25.73	21.84	15.53	23.79	21.36	18.45	16.02	17.96	16.02	13.59	18.21	3.77
MMP1	51.11	48.89	24.44	37.78	28.89	48.89	20.00	35.56	40.00	31.11	36.67	10.20
PDE4	22.22	18.52	25.93	25.93	25.93	29.63	25.93	18.52	14.81	44.44	25.93	7.73
PKC	51.05	56.15	71.47	56.15	51.05	51.05	56.15	45.95	56.15	51.05	53.60	6.48
RNN	21.70	21.70	18.87	21.70	15.09	24.53	15.09	10.38	14.15	10.38	16.98	4.78
SUBP	32.93	24.39	21.95	40.24	23.17	18.29	40.24	12.20	30.49	25.61	25.00	8.61
THRM	58.81	58.81	61.91	65.00	55.71	61.91	61.91	61.91	30.95	52.62	60.36	9.31

Table 8. Enrichment factors for fifteen (15) activity classes in the top 1% active compounds retrieval for SSA-GP.

Activity class	Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Run 8	Run 9	Run 10	Median	Standard deviation
COX	23.81	61.9	28.57	47.62	66.67	61.9	66.67	23.81	71.43	38.1	54.76	18.08
5HT	9.09	5.63	17.75	9.96	18.18	2.60	17.32	9.09	14.72	15.58	12.34	5.19
5HT1A	46.25	39.38	35	31.25	60.00	50.63	52.5	46.25	41.25	55.63	46.25	8.70
5HT3	100.00	94.74	57.89	94.74	100.00	78.95	63.16	84.21	89.47	52.63	86.84	16.85
ACHE	25.42	18.64	20.34	23.73	28.81	27.12	27.12	25.42	25.42	16.95	25.42	3.75
AT1	66.67	66.67	66.67	66.67	50.00	83.33	100	66.67	83.33	50.00	66.67	14.53
D2	25.51	32.14	38.78	36.73	49.49	47.45	50.00	25.51	34.18	36.73	36.73	8.54
FXA	43.88	67.63	58.99	58.27	53.24	39.57	67.63	43.88	64.03	57.55	57.91	9.59
HIVP	35.92	38.35	36.41	24.76	25.24	32.52	31.07	35.92	32.04	25.24	32.28	4.85
MMP1	84.44	77.78	95.56	66.67	91.11	86.67	80.00	84.44	82.22	51.11	83.33	12.13
PDE4	74.07	62.96	51.85	81.48	66.67	81.48	88.89	74.07	55.56	74.07	74.07	11.21
PKC	73.68	68.42	52.63	68.42	100.00	52.63	63.16	73.68	84.21	52.63	68.42	14.40
RNN	47.17	57.55	77.36	83.02	69.81	66.04	90.57	47.17	69.81	68.87	69.34	13.45
SUBP	82.93	67.07	58.54	60.98	79.27	97.56	69.51	82.93	64.63	64.63	68.29	11.75
THRM	100.00	91.86	84.88	96.51	82.56	55.81	50.00	63.95	67.44	60.47	75.00	17.06

Conversely, several classes, such as 5HT1A (median = 3.13, SD = 1.44) and 5HT (median = 9.00, SD = 1.40), recorded lower enrichment, indicating a weaker performance in identifying actives in more challenging or heterogeneous spaces. Variability in performance was also notable in some classes; for example, 5HT3 and COX showed high standard deviations (13.97 and 9.34, respectively), suggesting inconsistent rankings across the ten runs.

These findings highlight that although SSA-GA can achieve high enrichment in certain contexts, its performance is less stable and more sensitive to class characteristics than the GP-based approach.

3.2. Substructural analysis using GP results

Table 8 presents the EF outcomes for SSA-GP, using CW-RMSE as the fitness function. Compared with SSA-GA, SSA-GP generally achieved a stronger and more consistent performance across the 15 activity classes. Notably, the 5HT3 class recorded the highest median EF of 86.84 (SD = 16.85), followed closely by THRM (median = 75.00, SD = 17.06), MMP1 (median = 83.33, SD = 12.13), and PDE4 (median = 74.07, SD = 11.21), indicating the model's capacity to effectively rank active compounds in diverse target spaces.

The method also performed well in previously underperforming classes. For instance, 5HT1A, which showed limited enrichment under SSA-GA (median = 3.13), yielded a much higher median EF of 46.25 (SD = 8.70) with SSA-GP than with SSA-GA. Similarly, D2 and FXA achieved higher median EFs of 36.73 and 57.91, respectively, reflecting notable gains in compound prioritization.

Despite these improvements, some variability remained, particularly in COX (SD = 18.08) and 5HT3 (SD = 16.85), suggesting that enrichment performance can fluctuate across runs depending on the target class. Nonetheless, the SSA-GP results indicate that using CW-RMSE as the internal fitness function can improve enrichment performance and model stability across a broader range of biological targets compared with the SSA-GA baseline.

3.3. Comparison of EF using SSA-GA and SSA-GP

Figure 4 shows a bar graph comparing the median enrichment factors of SSA-GA and SSA-GP across the activity classes.

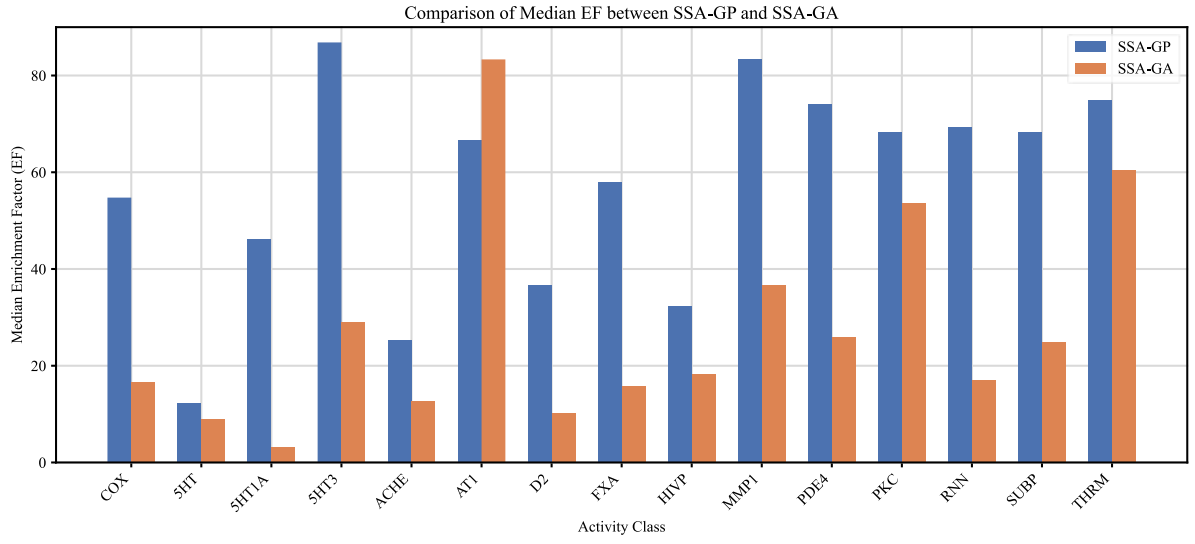

Figure 4. Bar chart comparison of median EF@1% between SSA-GA and SSA-GP.

The graph in Figure 4 compares the median EF values of SSA-GA and SSA-GP across 15 activity classes. SSA-GP achieved higher median EF@1% than SSA-GA in most activity

classes, with AT1 as the main exception. Improvements for SSA-GP were observed in PDE4 and PKC, where the performance gap between the two methods was more pronounced.

THRM and MMP1 exhibited considerable gains under SSA-GP, further highlighting the advantage of the proposed method in improving early active-compound retrieval. Similar improvements were also observed for COX, 5HT1A, D2, and FXA, indicating that SSA-GP generally provided stronger enrichment performance across activity classes. In contrast, 5HT showed a smaller difference between the two methods, suggesting more comparable performance for this class. These results indicate that SSA-GP improved median EF values for most activity classes, although the magnitude of improvement varied across targets.

The chart indicates that SSA-GP delivered higher median enrichment across most activity classes, supporting its potential for early active-compound prioritization.

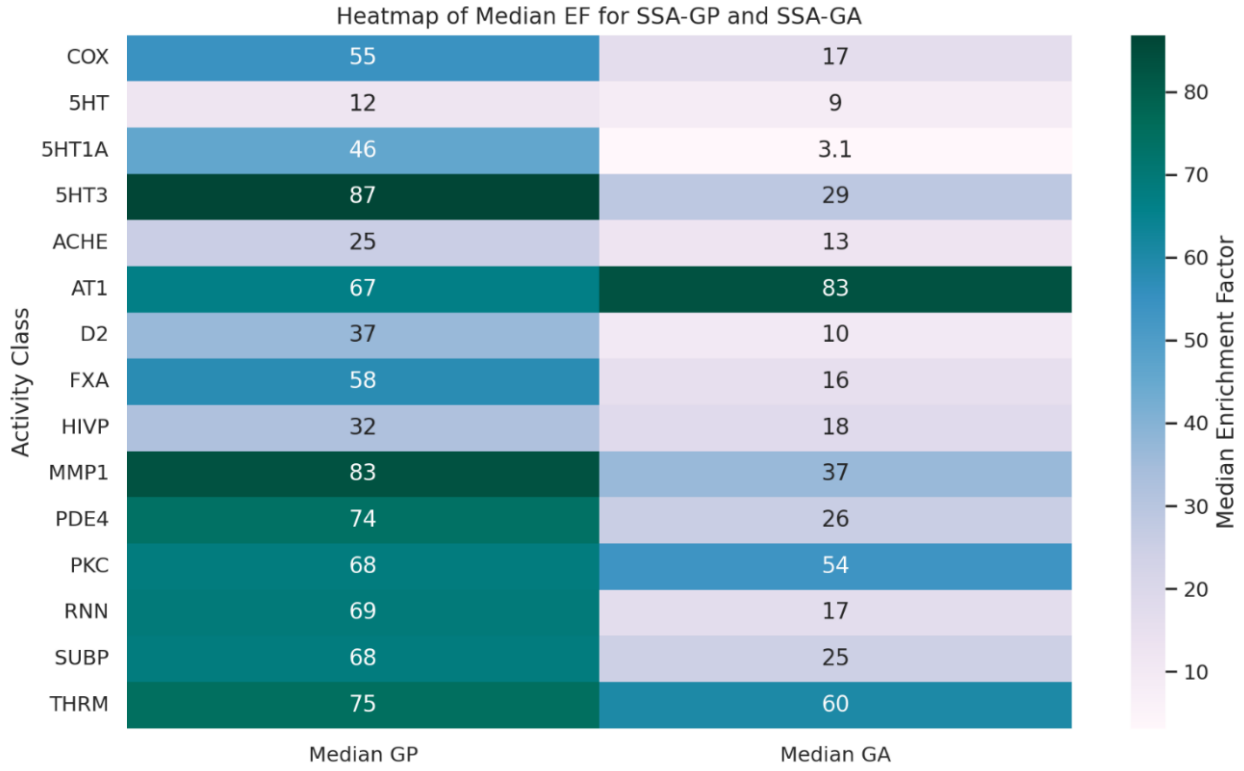

Figure 5. Heatmap of median EF@1% values for SSA-GP and SSA-GA.

The heatmap in Figure 5 shows the median EF@1% values for SSA-GP and SSA-GA across the activity classes. Higher median values were observed for SSA-GP in most classes, particularly 5HT3, RNN, MMP1, and PDE4. The statistical significance of these differences is reported separately in Table 9.

Significant positive differences were also evident in PKC, FXA, and SUBP, further reinforcing the class-dependent improvement achieved by SSA-GP in optimizing enrichment across biological targets. In contrast, certain classes, such as ACHE, 5HT, and HIVP, exhibited smaller or non-significant differences, suggesting comparable performances between GA and GP in these cases. Notably, AT1 was the only class in which a negative median difference was observed, indicating a slightly better performance of GA.

The findings suggest that SSA-GP generally improve enrichment outcomes; however, its effectiveness varies across activity classes, and SSA-GA may remain appropriate where the advantage of SSA-GP is limited.

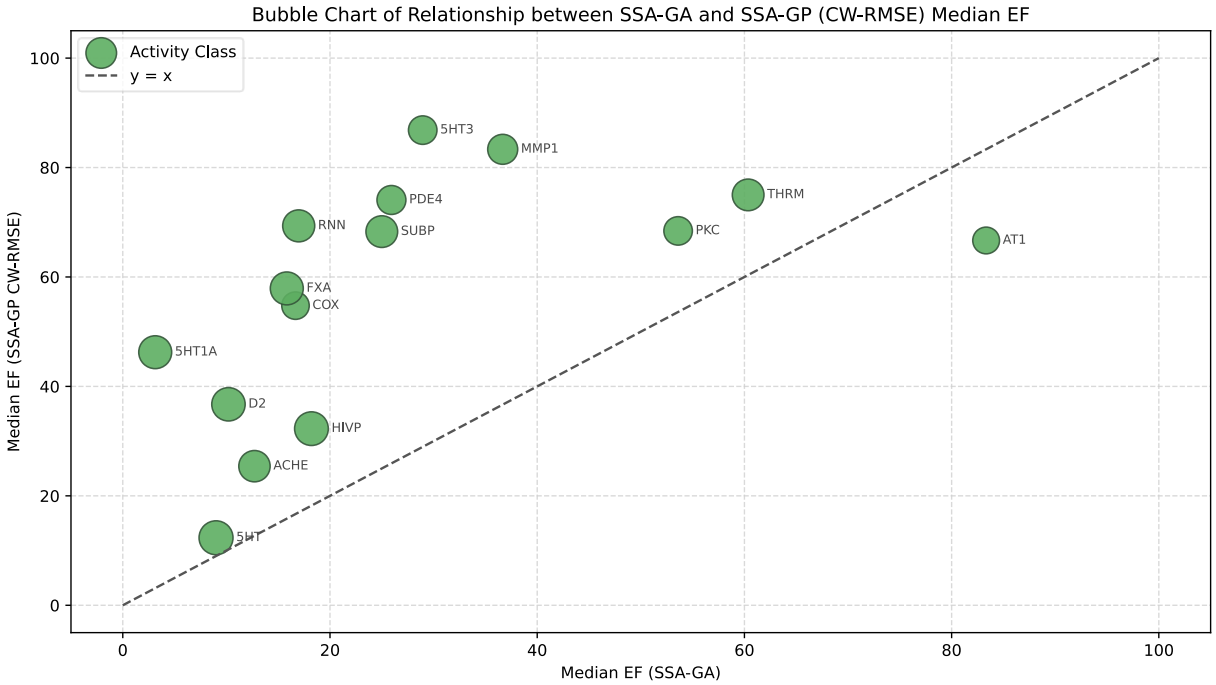

Figure 6. Bubble chart comparing median EF@1% values between SSA-GA and SSA-GP.

The bubble chart in Figure 6 illustrates the relationship between the median EF values of SSA-GA and SSA-GP across the activity classes. Most bubbles appear above the diagonal line, indicating that SSA-GP yielded higher median EF values than SSA-GA for most classes. Larger bubbles represent classes with greater performance differences, reinforcing the advantage of SSA-GP for early active-compound prioritisation. However, the position of AT1 below the diagonal line indicates that SSA-GA performed better for this class.

While the chart supports the general advantage of SSA-GP, smaller bubbles indicate activity classes where the performance difference between SSA-GP and SSA-GA was limited or not statistically significant. This suggests that SSA-GA may remain competitive for certain activity classes, particularly where the benefit of SSA-GP is less pronounced.

3.4. Mann-whitney U test

To statistically evaluate the results, the Mann–Whitney U test was used to determine whether there was a significant difference in EF@1% performance between SSA-GA and SSA-GP across the independent runs. This non-parametric test was selected because the EF@1% distributions may not follow a normal distribution, which is common in biological and chemical screening data. Unlike parametric tests such as the t-test, the Mann–Whitney U test does not require the assumptions of normality or equal variance, making it suitable for comparing the performance distributions of the two SSA-based methods.

Using this test, we determined whether the differences in median EF@1% between SSA-GA and SSA-GP were statistically significant rather than attributable to random variation across independent runs. The analysis provided evidence on the extent to which SSA-GP improved early active-compound retrieval across activity classes. The statistical results supported the stronger ranking capability of SSA-GP in most classes, thereby strengthening the reliability of the proposed approach for fragment-based virtual screening and structure–activity relationship analysis.

Table 9. Mann-whitney U test result comparing SSA-GA and SSA-GP.

Activity class	Median EF (GP)	Median EF (GA)	U statistic	P-value	Conclusion
COX	54.76	16.67	93.50	0.001103	Significant
5HT	12.34	9.00	73.00	0.088372	Not significant
5HT1A	46.25	3.13	100.00	0.000175	Significant
5HT3	86.84	28.95	99.50	0.000202	Significant
ACHE	25.42	12.71	98.00	0.000305	Significant
AT1	66.67	83.33	19.50	0.020202	Significant
D2	36.73	10.20	100.00	0.000179	Significant
FXA	57.91	15.83	100.00	0.000179	Significant
HIVP	32.28	18.21	97.00	0.000433	Significant
MMP1	83.33	36.67	99.50	0.000209	Significant
PDE4	74.07	25.93	100.00	0.000167	Significant
PKC	68.42	53.60	82.00	0.016181	Significant
RNN	69.34	16.98	100.00	0.000175	Significant
SUBP	68.29	25.00	100.00	0.000180	Significant
THRM	75.00	60.36	78.00	0.036847	Significant

Table 9 compares the EF@1% performance of SSA-GP and SSA-GA across 15 activity classes. SSA-GP achieved higher median EF@1% than SSA-GA in 14 of the 15 classes. Among these 14 classes, 13 showed statistically significant improvements ($p < 0.05$), whereas 5HT showed a higher median under SSA-GP but the difference was not statistically significant. The largest positive gains were observed in D2, FXA, MMP1, PDE4, RNN, SUBP, and THRM. AT1 was the only class in which SSA-GA achieved a higher median EF@1% than SSA-GP.

For example, the median EF@1% increased from 16.98 under SSA-GA to 69.34 under SSA-GP in RNN, and from 36.67 to 83.33 in MMP1. AT1 was the only class in which SSA-GA achieved a higher median EF@1% than SSA-GP (83.33 vs. 66.67). This exception may reflect the very small number of active compounds in the AT1 test set, as well as a fragment pattern that may be more effectively captured by direct GA-based weighting than by the evolved symbolic equations.

The findings indicate that SSA-GP offers an advantage for most activity classes, although its superiority is not universal. This cautious interpretation is important because enrichment performance in highly imbalanced classes can be sensitive to small changes in the number of active compounds retrieved at the top of the ranked list.

3.5. Complementary evaluation metrics

Using the definitions introduced in the evaluation section, SSA-GA and SSA-GP were also compared using AUC, Average Precision (AP), and Recall@1%. These complementary metrics were included because EF@1% focuses on early recognition, whereas AUC and AP provide broader information on ranking and discrimination across the test set.

Table 10. Comparison of SSA-GP and SSA-GA across AUC, average precision (AP), and Recall@1% for 15 activity classes.

Activity class	AUC SSA-GP	AP SSA-GP	Recall@1% SSA-GP	AUC SSA-GA	AP SSA-GA	Recall@1% SSA-GA
5HT	0.6554	0.1738	5.67	0.4144	0.1783	2.38
5HT1A	0.7387	0.2873	8.44	0.5325	0.1151	0.00
5HT3	0.9102	0.4302	8.00	0.4746	0.0223	4.00
AT1	0.7292	0.0882	0.00	0.7993	0.0295	0.00
FXA	0.7689	0.3871	4.96	0.8041	0.2713	1.95
HIVP	0.6583	0.2121	6.76	0.5744	0.1988	1.76
PDE4	0.7436	0.2185	7.67	0.6394	0.0692	5.41
SUBP	0.7625	0.2402	5.89	0.5253	0.0693	1.23
THRM	0.7948	0.2667	6.74	0.6666	0.0939	4.29
ACHE	0.7012	0.1024	2.45	0.6465	0.0818	1.22
D2	0.7428	0.2415	6.73	0.584	0.1705	4.19
RNN	0.707	0.1738	7.77	0.7229	0.1609	3.88
COX	0.7391	0.1625	4.74	0.8779	0.2834	4.65
PKC	0.8253	0.2698	8.74	0.8068	0.249	3.46
MMP1	0.8051	0.1532	7.69	0.6666	0.0939	4.29

Table 10 presents the performance of SSA-GP and SSA-GA across 15 activity classes using three metrics: AUC, AP, and Recall@1%. These metrics reflect overall ranking quality, early active retrieval, and class discrimination.

SSA-GP achieved higher AUC values than SSA-GA in 11 of the 15 activity classes, with substantial gains in 5HT3 (0.91 vs. 0.47), 5HT1A (0.74 vs. 0.53), and SUBP (0.76 vs. 0.53). However, SSA-GA produced higher AUC values for AT1, FXA, RNN, and COX, indicating that the overall discrimination advantage of SSA-GP was not uniform across all activity classes.

In terms of AP, SSA-GP produced higher values than SSA-GA in most cases, particularly for 5HT3 (0.43 vs. 0.02), 5HT1A (0.29 vs. 0.12), and SUBP (0.24 vs. 0.07), suggesting improved early ranking of active compounds for these activity classes.

The Recall@1% values further support these results. SSA-GP retrieved more actives within the top 1% of the ranked list for nearly all classes than SSA-GA. For instance, for 5HT1A, SSA-GP achieved 8.44% recall, whereas SSA-GA retrieved none. Similar improvements were observed for MMP1 (7.69% vs. 4.29%) and PKC (8.74% vs. 3.46%).

Taken together, the EF@1%, AUC, AP, and Recall@1% results indicate that SSA-GP generally improves early active-compound prioritization, although the magnitude of improvement varies by activity class.

3.6. Interpretation of GP-generated mathematical equations

During GP symbolic regression, one representative equation was selected for each activity class based on its EF@1% performance and stability over the independent runs. The equations in Table 11 should be interpreted as fragment-weighting functions for SSA rather than direct biochemical mechanistic models.

Table 11. Representative GP-generated fragment-weighting equations for SSA-GP.

Activity class	GP mathematical equation
COX	$-cos(log(ACT * N))/(log(NACT) - ACT + sin(TOT))$
5HT	$(2 * (cos(INACT\text{^}(1/2)) + (sin(log(NACT)/ACT) * log(NACT))/2))/log(NACT)$
5HT1A	$(INACT * sin(TOT + log(INACT)))/(NACT - N)\text{^}(1/2)$
5HT3	$N/(cos(N) - INACT * NINACT)$
ACHE	$-(TOT * (ACT - INACT))/(INACT\text{^}(1/2) * (ACT - INACT + N + N/ACT))$
AT1	$sin(log((ACT - TOT)/log(TOT)))$
D2	$(log(NACT) * (INACT - log(NINACT)))/(NACT * sin(N))$
FXA	$log(NINACT)/(ACT * log(ACT)) - sin(cos(log(INACT)))$
HIVP	$-sin(log(N - TOT) - sin(N)/INACT\text{^}(1/2))$
MMP1	$(log(NACT) * (ACT - cos(ACT)))/(INACT + cos(ACT))\text{^}2$
PDE4	$(log(cos(INACT) - ACT + cos(TOT/ACT))/(sin(sin(TOT)) * (NINACT - NACT)\text{^}(1/2))$
PKC	$sin(ACT * sin(NACT))/(INACT\text{^}2 * (N/NACT)\text{^}(1/2))$
RNN	$(log(ACT)/2 + log(N - NINACT))/INACT$
SUBP	$sin(log(INACT/log(TOT)))$
THRM	$(ACT + cos(ACT) + INACT * cos(NACT))/(tan(cos(NACT)) + cos(INACT) + INACT\text{^}2)$

The terminal set consisted of ACT, INACT, TOT, N, NACT, and NINACT. ACT denotes the number of active training compounds containing a given fragment, INACT denotes the number of inactive/background training compounds containing the fragment, TOT denotes the total number of training compounds containing the fragment, N denotes the total number of training compounds in the activity class, NACT denotes the total number of active compounds in the training set, and NINACT denotes the total number of inactive/background compounds in the training set.

Equation 5 illustrates the COX-specific fragment-weighting function evolved by GP. The equation combines fragment-level statistics, including ACT, N, NACT, and TOT, to transform fragment occurrence information into an SSA weighting expression. The logarithmic and trigonometric terms should be interpreted as computational transformations selected by symbolic regression rather than direct biochemical mechanisms. Thus, the equation is interpretable at the level of its terminal symbols and mathematical operations, but chemical interpretation of individual terms requires further assessment by domain experts.

$$-cos(log(ACT * N))/(log(NACT) - ACT + sin(TOT)) \quad (5)$$

The mathematical expression generated through GP plays a critical role in fragment-based screening and SAR analysis. This enables the evaluation of compounds based on a combination of their relative activity, structural complexity, and representation within the dataset. To capture the nonlinear patterns often observed in molecular activity prediction, the model incorporates mathematical transformations such as logarithmic and trigonometric functions.

The logarithmic and trigonometric terms should be interpreted as nonlinear transformations of fragment statistics rather than direct biochemical mechanisms. Similar symbolic expressions were developed for other activity classes, each tailored to reflect the unique molecular features and distribution patterns specific to the datasets. For instance, in the RNN activity class, the following equation is derived (Eq 6):

$$(log(ACT)/2 + log(N - NINACT))/INACT \quad (6)$$

This formula emphasizes the balance between active and inactive compounds and uses logarithmic transformations to manage different dataset distributions. The first term, log(ACT)/2, adjusts the contribution of the active compounds, preventing the domination of highly active fragments. The second term, log(N−NINACT)/INACT, captures the proportion of compounds that are not completely inactive or too common in the dataset, ensuring a more balanced weighting of molecular contributions. Logarithmic scaling in both terms maintains model stability when handling datasets with significant differences in activity levels, enhancing adaptability to various molecular datasets.

The adaptability of these mathematical formulas enables predictive models to function well across chemical spaces while remaining understandable. This is especially important in fragment-based screening, where the relationships between small molecular fragments and biological activity are often complex and context-dependent. By improving these equations through symbolic regression in a GP framework, the models can capture the basic SAR principles without requiring pre-set ideas about molecular interactions. The fact that these equations are understandable enables researchers to gain meaningful chemical insights and suggest possible fragment-weighting patterns that may guide further chemical interpretation and support future lead optimization in drug discovery. Therefore, these mathematical models do more than just predict; they act as analytical tools that connect computational modeling with experimental testing, aiding in the rational design of bioactive compounds.

4. Conclusions

In this study, we investigated SSA-GP, in which GP was guided by the CW-RMSE fitness function, and compared it with SSA-GA. The results show that SSA-GP achieved higher median EF@1% than SSA-GA in most of the 15 ChEMBL-derived activity classes, with statistically significant improvements in 13 classes. AT1 was the only exception, where SSA-GA achieved a higher median EF@1%, indicating that the proposed approach is beneficial for many, but not all, target classes.

The results suggest that SSA-GP, guided by the CW-RMSE fitness function, can improve early active-compound prioritization and support structure–activity relationship analysis in cheminformatics. However, this study is limited to SSA-based comparisons and EF@1%-centred evaluation; therefore, further validation is required to assess its robustness and

applicability across molecular datasets. Future work should include nested training-only parameter selection, EF@5% and EF@10% assessment, and evaluation against modern non-SSA baselines, such as Random Forest, support vector machines, deep learning, and graph-based molecular models, together with expert validation of the generated symbolic equations. In addition, researchers should examine the scalability of SSA-GP to larger and more chemically diverse datasets, the reliability of CW-RMSE across varied chemical environments, and its integration with hybrid modeling approaches such as deep learning, ensemble learning, Bayesian optimization, or reinforcement learning. These extensions may further improve fragment weighting, prediction accuracy, and chemical relevance while preserving the interpretability of the SSA-GP framework. In summary, SSA-GP offers an interpretable and evidence-based approach for scalable molecular screening and lead prioritization.

Author contributions

Mohd Isrul Esa: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing – Original Draft; Nor Samsiah Sani: Data Curation, Writing – Review & Editing, Supervision; Salwani Abdullah: Supervision, Project administration.

Use of AI tools declaration

The authors declare that OpenAI ChatGPT was used during revision to assist with language refinement, clarity improvement, restructuring of revised passages, and preparation of responses to reviewers. Grammarly was used as an assistive language-editing tool. These tools were used in revised passages throughout the Introduction, Materials and methods, Results and discussion, and Conclusions. The AI tools were not used to generate, fabricate, or modify research data, perform statistical analysis, create original research results, generate figures or images, or draw scientific conclusions. All outputs, technical content, analyses, interpretations, references, and conclusions were independently reviewed, verified, edited, and approved by the authors. The authors take full responsibility for the final content of the manuscript.

Conflict of interest

The authors declare no conflict of interest.

References

1. Alves VM, Yasgar A, Wellnitz J, et al. (2023) Lies and liabilities: Computational assessment of high-throughput screening hits to identify artifact compounds. *J Med Chem* 66: 12828–12839. https://doi.org/10.1021/acs.jmedchem.3c00482
2. Chung HH, Kao CY, Wang TSA, et al. (2021) Reaction tracking and high-throughput screening of active compounds in combinatorial chemistry by tandem mass spectrometry molecular networking. *Anal Chem* 93: 2456–2463. https://doi.org/10.1021/acs.analchem.0c04481
3. Feng T, Basu P, Sun W, et al. (2019) Optimal design for high-throughput screening via false discovery rate control. *Stat Med* 38: 2816–2827. https://doi.org/10.1002/sim.8144

4. Muegge I, Bentzien J, Ge Y (2024) Perspectives on current approaches to virtual screening in drug discovery. *Expert Opin Drug Discov* 19: 1173–1183. https://doi.org/10.1080/17460441.2024.2390511
5. Warr WA, Nicklaus MC, Nicolaou CA, et al. (2022) Exploration of ultralarge compound collections for drug discovery. *J Chem Inf Model* 62: 2021–2034. https://doi.org/10.1021/acs.jcim.2c00224
6. Shen C, Hu Y, Wang Z, et al. (2021) Beware of the generic machine learning-based scoring functions in structure-based virtual screening. *Brief Bioinform* 22: bbaa070. https://doi.org/10.1093/bib/bbaa070
7. Choi H, Kang H, Chung KC, et al. (2019) Development and application of a comprehensive machine learning program for predicting molecular biochemical and pharmacological properties. *Phys Chem Chem Phys* 21: 5189–5199. https://doi.org/10.1039/C8CP07002D
8. Niazi SK, Mariam Z (2023) Recent advances in machine-learning-based chemoinformatics: A comprehensive review. *Int J Mol Sci* 24: 11488. https://doi.org/10.3390/ijms241411488
9. Ross GA, Morris GM, Biggin PC (2013) One size does not fit all: The limits of structure-based models in drug discovery. *J Chem Theory Comput* 9: 4266–4274. https://doi.org/10.1021/ct4004228
10. Chauhan SS, Jamal T, Singh A, et al. (2023) Structure-based virtual screening, In: *Cheminformatics, QSAR and Machine Learning Applications for Novel Drug Development*, Elsevier, 239–262. https://doi.org/10.1016/B978-0-443-18638-7.00016-5
11. Cramer RD, Redi G, Berkoff CE (1974) Substructural analysis. A novel approach to the problem of drug design. *J Med Chem* 17: 533–535. https://doi.org/10.1021/jm00251a014
12. Machado-Alba JE, Jiménez-Morales AL, Moran-Yela YC, et al. (2020) Adverse drug reactions associated with the use of biological agents. *PLoS One* 15: e0240276. https://doi.org/10.1371/journal.pone.0240276
13. Villar HO, Hansen MR, Kho R (2007) Substructural analysis in drug discovery. *Curr Comput Aided-Drug Des* 3: 59–67. https://doi.org/10.2174/157340907780058745
14. Abdo A, Salim N (2011) New fragment weighting scheme for the Bayesian inference network in ligand-based virtual screening. *J Chem Inf Model* 51: 25–32. https://doi.org/10.1021/ci100232h
15. Holliday JD, Sani N, Willett P (2015) Calculation of substructural analysis weights using a genetic algorithm. *J Chem Inf Model* 55: 214–221. https://doi.org/10.1021/ci500540s
16. Holliday JD, Sani N, Willett P (2018) Ligand–based virtual screening using a genetic algorithm with data fusion. *Match Commun Math Co* 80: 623–638.
17. Wang M, Wu Z, Wang J, et al. (2024) Genetic algorithm-based receptor ligand: A genetic algorithm-guided generative model to boost the novelty and drug-likeness of molecules in a sampling chemical space. *J Chem Inf Model* 64: 1213–1228. https://doi.org/10.1021/acs.jcim.3c01964
18. Wang Z, Sobey A (2020) A comparative review between Genetic Algorithm use in composite optimization and the state-of-the-art in evolutionary computation. *Compos Struct* 233: 111739. https://doi.org/10.1016/j.compstruct.2019.111739
19. Greenstein BL, Elsey DC, Hutchison GR (2023) Determining best practices for using genetic algorithms in molecular discovery. *J Chem Phys* 159: 091501. https://doi.org/10.1063/5.0158053

20. Chiang TC, Chang CH, Yu TL (2024) A novel symbolic regressor enhancer using genetic programming. 2024 IEEE Congress on Evolutionary Computation (CEC); 1–8. https://doi.org/10.1109/CEC60901.2024.10612124
21. Mei Y, Chen Q, Lensen A, et al. (2023) Explainable artificial intelligence by genetic programming: A survey. *IEEE T Evolut Comput* 27: 621–641. https://doi.org/10.1109/TEVC.2022.3225509
22. Fleck P, Werth B, Affenzeller M (2024) Population dynamics in genetic programming for dynamic symbolic regression. *Appl Sci* 14: 596. https://doi.org/10.3390/app14020596
23. Tran B, Xue B, Zhang M (2016) Genetic programming for feature construction and selection in classification on high-dimensional data. *Memet Comput* 8: 3–15. https://doi.org/10.1007/s12293-015-0173-y
24. Darwish SM, Shendi TA, Younes A (2019) Quantum-inspired genetic programming model with application to predict toxicity degree for chemical compounds. *Expert Syst* 36: 1–15. https://doi.org/10.1111/exsy.12415
25. Deighan DS, Field SE, Capano CD, et al. (2021) Genetic-algorithm-optimized neural networks for gravitational wave classification. *Neural Comput Appl* 33: 13859–13883. https://doi.org/10.1007/s00521-021-06024-4
26. Stanovov V, Akhmedova S, Semenkin E (2022) The automatic design of parameter adaptation techniques for differential evolution with genetic programming. *Knowl-Based Syst* 239: 108070. https://doi.org/10.1016/j.knosys.2021.108070
27. Fan Q, Bi Y, Xue B, et al. (2024) A genetic programming-based method for image classification with small training data. *Knowl-Based Syst* 283: 111188. https://doi.org/10.1016/j.knosys.2023.111188
28. Burlacu B, Yang K, Affenzeller M (2024) Population diversity and inheritance in genetic programming for symbolic regression. *Nat Comput* 23: 531. https://doi.org/10.1007/s11047-022-09934-x
29. Lee CM, Ahn CW, Kim MJ (2024) Feature optimization and dropout in genetic programming for data-limited image classification. *Mathematics* 12: 3661. https://doi.org/10.3390/math12233661
30. Makke N, Chawla S (2024) Interpretable scientific discovery with symbolic regression: a review. *Artif Intell Rev* 57: 2. https://doi.org/10.1007/s10462-023-10622-0
31. Archetti F, Giordani I, Vanneschi L (2010) Genetic programming for QSAR investigation of docking energy. *Appl Soft Comput* 10: 170–182. https://doi.org/10.1016/j.asoc.2009.06.013
32. Gardiner EJ, Gillet VJ (2015) Perspectives on knowledge discovery algorithms recently introduced in chemoinformatics: Rough set theory, association rule mining, emerging patterns, and formal concept analysis. *J Chem Inf Model* 55: 1781–1803. https://doi.org/10.1021/acs.jcim.5b00198
33. Orlov AA, Zherebker A, Eletskaya AA, et al. (2019) Examination of molecular space and feasible structures of bioactive components of humic substances by FTICR MS data mining in ChEMBL database. *Sci Rep* 9: 12066. https://doi.org/10.1038/s41598-019-48000-y
34. Zdrazil B (2025) Fifteen years of ChEMBL and its role in cheminformatics and drug discovery. *J Cheminform* 17: 32. https://doi.org/10.1186/s13321-025-00963-z
35. Miralavy I, Bricco AR, Gilad AA, et al. (2022) Using genetic programming to predict and optimize protein function. *PeerJ Phys Chem* 4: e24. https://doi.org/10.7717/peerj-pchem.24

36. Krüger DM, Evers A (2010) Comparison of structure- and ligand-based virtual screening protocols considering hit list complementarity and enrichment factors. *ChemMedChem* 5: 148–158. https://doi.org/10.1002/cmdc.200900314

AIMS Press