Received: 25 June 2024 | Revised: 8 October 2024 | Accepted: 18 December 2024 | Published online: 31 December 2024

REVIEW

Medinformatics
2024, Vol. 00(00) 1–16
DOI: 10.47852/bonviewMEDIN42023692

BON VIEW PUBLISHING

A Comprehensive Survey of Genetic Programming Applications in Modern Biological Research

Mohammad Wahab Khan[1,*]

[1]*Department of Bioinformatics, Pondicherry University, India*

Abstract: Genetic programming (GP) has emerged as a powerful tool over the past two decades, leveraging evolutionary algorithms to navigate high-dimensional solution spaces effectively. This paper provides a comprehensive survey of GP's applications across various scientific domains, with a particular focus on bioinformatics and drug discovery. We discuss how GP facilitates the quantification, localization, and functional analysis of proteins. We highlighted its role in improving mass spectrometric (MS) peptide detectability through advanced pre-processing techniques. By enhancing the identification accuracy of peptides in proteomics, GP has significantly surpassed traditional methods. Additionally, we explore GP's capabilities in pattern matching and motif discovery within protein and DNA sequences, underscoring its utility in cancer research and biomarker detection. The paper also examines the integration of GP with machine learning strategies to address challenges in mass spectrometry, enabling the identification of biomarkers from complex datasets. Furthermore, we discuss innovative GP-based methods for predicting protein structure and function, as well as its application in drug discovery, where it outperforms conventional machine learning techniques in predicting pharmacokinetic properties. Through this survey, we aim to elucidate the versatility and effectiveness of GP in tackling complex biological problems, paving the way for future research and applications in the life sciences.

Keywords: genetic programming, proteomics, bioinformatics, protein, cancer, motif

1. Genetic Programming Introduction

Artificial intelligence encompasses the development of intelligent machines that operate based on their own set of instructions. One of the latest techniques in this field is genetic programming (GP), which originated from the evolutionary method known as genetic algorithm (GA) [1]. GP involves the evolution and development of programs to solve complex problems or expressions of varying sizes and shapes [2]. It utilizes the Neo-Darwinian theory to run automatic programs over many generations, using concepts such as mutation, crossover, and gene duplication to create new functions and sub-populations. The tree-based representation of programs proposed by Langdon [3] has made GP the most commonly used form for computer program development. This method solves problems in a systematic and domain-independent manner, starting from a high-level statement of what needs to be done. Stochastically operated GP transforms populations of programs into novel, hopefully better populations over generations. To better understand the flexible nature of GP, refer to Figure 1 for a typical GP flowchart.

Koza's GP is based on a set of five preparative steps. These steps include:

1) Defining the set of terminals for each branch of the program to be evolved. This includes the problem's independent variables, zero-argument functions, and random constants.
2) Determining the set of primitive functions for each branch of the program. This includes arithmetic operations, mathematical functions, Boolean operations, conditional operators, and functions causing iteration, functions causing recursion, and any other domain-specific functions.
3) Establishing a fitness measure that quantifies the rightness of a solution to the problem. This measure can incorporate any measurable, observable, or calculable behavior or characteristic.
4) Setting certain parameters to control the program's execution. This includes determining the population size and the probabilities for crossover and mutation.
5) Establishing termination criteria and criteria for determining the program's result. This typically involves setting a maximum number of generations to be run, which serves as a necessary condition for problem-specific success.

The primitive set of a GP system is defined by the sets of allowed functions and terminals, which indirectly determine the search space for GP. In order for the terminal and function sets to be effective, they must meet the requirements of closure, sufficiency, and universality. Fitness is measured in terms of "what needs to be done", rather than "how to do it", and helps to identify the best elements in the search space. The remaining two control parameters and termination criteria affect the quality and speed of search.

GP utilizes syntax trees to represent programs, with variables and constants as leaves of the tree (termed "terminals") and

*Corresponding author: Mohammad Wahab Khan, Department of Bioinformatics, Pondicherry University, India. Email: wahab@bicpu.edu.in

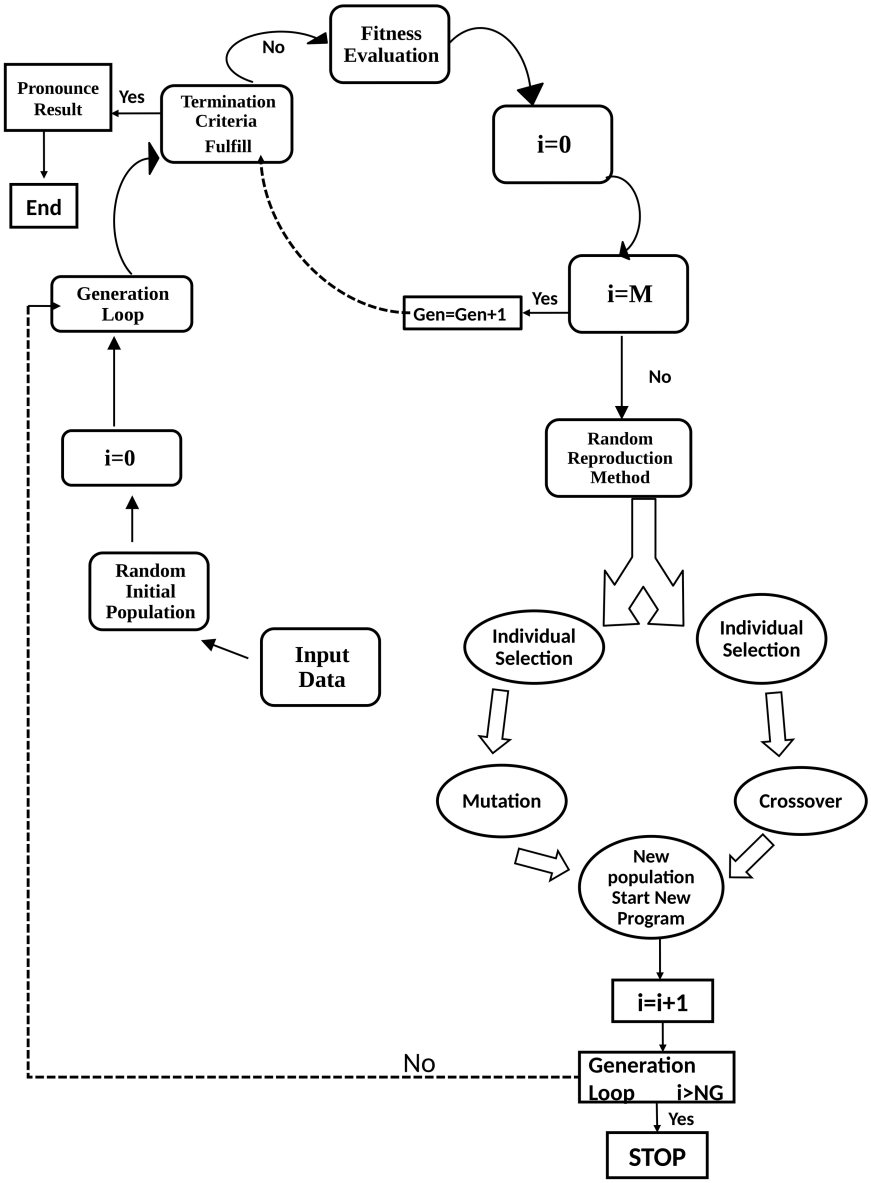

Figure 1. A typical genetic programming flow chart illustrating the step-by-step process of evolving solutions. This flow chart includes stages such as initialization, selection, crossover, mutation, and evaluation of fitness, demonstrating how genetic programming iteratively refines candidate solutions to optimize performance in problem-solving tasks.

arithmetic operations as internal nodes ("functions"). The GP tree is represented using Polish notation, commonly used in Lisp programming. Figure 2 provides an example of the tree representation of the program 2X2+4X+6 written as (+(*(* X X) 2) (+(+ 2 3) (+ 4 X))). The variables and constants in the program (X, 2, 3, and 4) are leaves of the tree. In GP, they are called terminals, while the arithmetic operations (+, *) are internal nodes known as functions.

Crossover and mutation are unique features of GP that set it apart from other evolutionary algorithms. Sub-tree crossover is the most commonly used form of crossover, which randomly selects a crossover point in each parent tree and creates offspring by replacing the sub-tree rooted at the crossover point in a copy of the first parent with a copy of the sub-tree rooted at the crossover point in the second parent. Figure 3 illustrates a valid crossover operation using two parent expressions.

$$\text{Parent 1 : } 2X + X + 3Y \text{ represented as} (+(+XX)(+(*Y3)X))$$
$$\text{Parent 2: } X2 + 6X \text{ represented as} (+(*XX)(*(+(-52)3)X))$$

Parent 1 has input variables 'x' and 'y' and a constant '2 &3' while parent 2 has one input variable X and constant 6. If "*"from parent 1 and the "+" from parent 2 are chosen as the crossover points, then the two offsprings are given by,

$$\text{Offspring 1:} 2X + 6 + X \text{ represented as} (+(+XX)(+(+(-52)3)X))$$
$$\text{Offspring 2: } X2 + 3YX \text{ represented as} (+(*XX)(*(+Y3)X))$$

Copies are used to avoid disrupting the original individuals. The selection of subtrees many times helps to create the offspring programs multiple times.

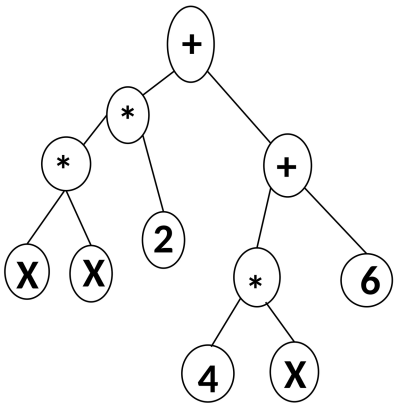

Figure 2. Basic genetic programming tree representation showcasing the hierarchical structure of programs. This figure depicts how genetic programs are represented as tree-like structures, with nodes representing functions and terminals, allowing for the visualization of the program's logical flow and its components in the context of evolutionary computation.

GP, a key tool within Evolutionary Intelligence, harnesses nature-inspired optimization techniques. Earlier studies delve into its impact across diverse fields, from research and engineering to education and Explainable AI (XAI) [4]. Through recombination and mutation in Evolutionary Algorithms, GP offers a balance between model accuracy and interpretability, showcasing promise in critical applications like self-driving cars and healthcare [5]. Additionally, study reported GP's role in education [6] and introduced current research trends in this evolving field [7]. GP has more robustness and utility advantages over other evolutionary algorithms, such as GAs, differential evolution (DE), and particle swarm optimization (PSO). GP excels in its ability to evolve complex models that can adaptively optimize solutions in proteomics, cancer research, and protein stability assessments. Unlike GAs, which operate on fixed-length chromosomes and may struggle with high-dimensional problems, GP generates variable-length solutions, enabling it to capture the complexity of biological interactions more effectively. In contrast to DE, which relies on a population of candidate solutions that evolve through mutation and recombination, GP can create entirely new structures and functions, making it particularly adept at discovering novel

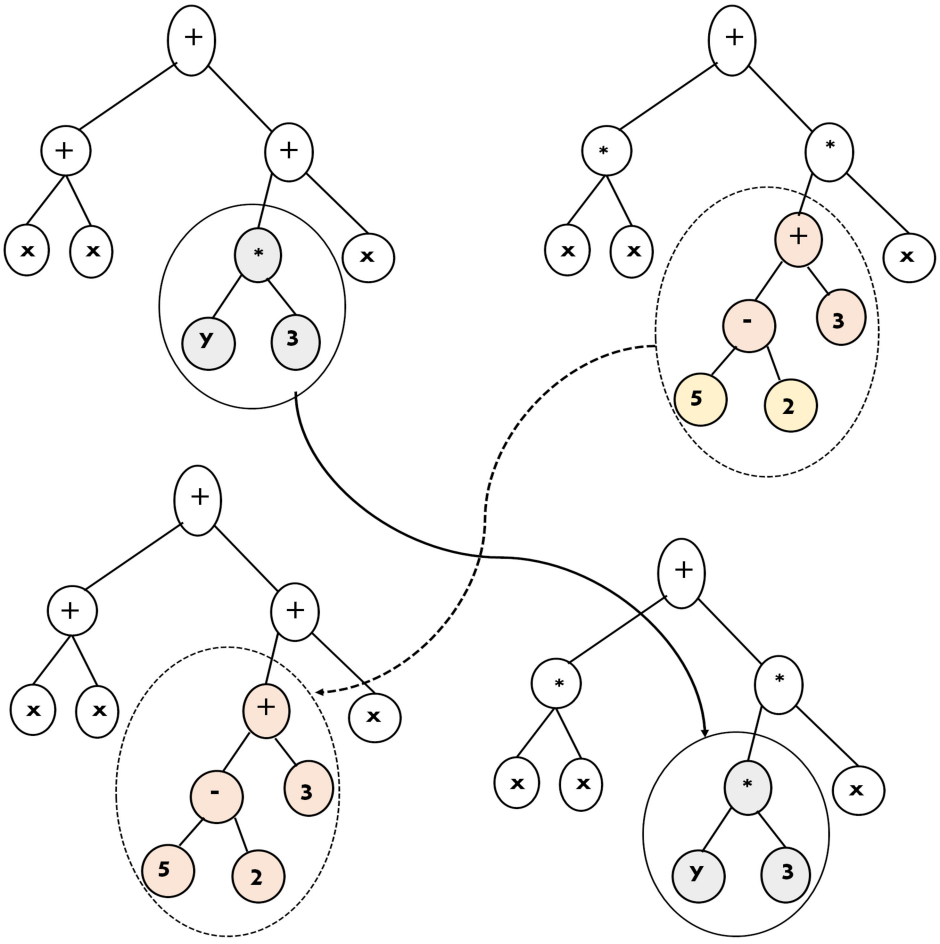

Figure 3. GP crossover representation detailing the genetic recombination process. This figure illustrates how two parent trees are combined to produce offspring by exchanging subtrees, highlighting the mechanisms of genetic diversity and innovation that enhance the search for optimal solutions in genetic programming.

Table 1. Comparison of genetic programming with other evolutionary algorithms, emphasizing key differences in conceptual models, strengths, and weaknesses

Algorithm	Conceptual model	Strengths	Weaknesses
Genetic Programming	Tree-based representation	Flexible and can evolve solutions directly	Computationally intensive and risk of bloat
Genetic Algorithms	Fixed-length chromosome	Simple implementation and robust	Limited to predefined structures and less flexible
Evolutionary Strategies	Real-valued representation	Effective for continuous optimization	Less effective for discrete problems
Differential Evolution	Population-based optimization	Good for global optimization	May struggle with local optima

bimolecular patterns. Additionally, while PSO focuses on optimizing a swarm of particles in a defined search space, GP's tree-based representation allows for a more nuanced exploration of solution landscapes, facilitating better adaptability in dynamic environments. Table 1 illustrates the key differences in conceptual models, strengths, and weaknesses by comparing GP with other evolutionary algorithms. By emphasizing these comparative strengths, the GP method can be positioned as a transformative tool in the ongoing quest to unravel the complexities of biological systems [8].

2. GP Application

GP has found widespread application in various scientific fields, including genomics [9], engineering, economics, forecasting, computer science, medicine, and biology [10]. Table 2 provides a summary of the applications of GP across various fields, highlighting specific examples and their outcomes. This article delves into the profound impact of GP on the realms of bioinformatics and proteomics, visually represented in Figure 4. The comprehensive literature survey segments findings into six key domains: Bioinformatics and proteomics, Protein Quantification and Localization, Protein Structure and Function Prediction, Protein-protein Interaction, Motif Discovery and RNA Secondary Structure Predictions, and Drug Discovery and Cancer Biology. These categories offer a holistic view of GP's versatile applications in these critical areas. This paper presents GP as a novel method with significant applicability in proteomics, cancer research, and the assessment of protein stability. This method includes its ability to evolve variable-length solutions, capture complex biological interactions, and outperform traditional evolutionary algorithms like GAs, DE, and PSO. The organization

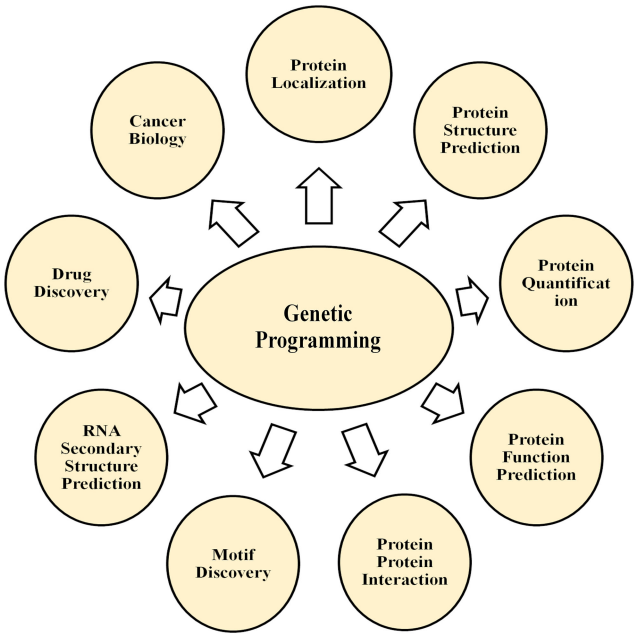

Figure 4. Graphical abstract of genetic programming application, summarizing the key concepts and methodologies employed in various fields. This figure provides a visual overview of how genetic programming is utilized in bioinformatics, drug discovery, and other domains, emphasizing its adaptability and effectiveness in solving complex problems through evolutionary strategies.

Table 2. Summary of applications of genetic programming across various fields, highlighting specific examples and outcomes

Field	Application	Specific examples	Outcomes
Engineering	Optimization Problems	Design of efficient structures	Improved performance and reduced costs in engineering designs.
Economics	Market Prediction	Stock price forecasting	Enhanced prediction accuracy for financial markets.
Computer Science	Algorithm Development	Automated coding and software optimization	Increased efficiency in code generation and optimization.
Medicine	Disease Diagnosis	Predictive models for patient outcomes	Improved diagnostic accuracy and treatment planning.
Biology	Genetic Analysis	Gene expression profiling	Identification of significant genetic markers for diseases.

of this paper is structured to first introduce the foundational concepts and methodologies, followed by a detailed analysis of application results, and conclude with a discussion on the implications of findings and future directions for research. This framework ensures clarity and facilitates a deeper understanding of the transformative potential of GP in addressing critical challenges in bioinformatics.

2.1. Bioinformatics and proteomics

Bioinformatics is a multidisciplinary field that combines information science, statistics, mathematics, engineering, and computer science to analyze biological data. The completion of the "Human Genome Project" has led to a significant increase in biological data over the past two decades, making bioinformatics an essential tool for research in biological science. From gene prediction to genome analysis, pathways reconstruction, comparative genomics, genetics of diseases, analysis of gene and protein expression, structural bioinformatics, to network and system biology, bioinformatics covers a wide range of research areas [11].

Proteomics is a field of study that deals with the identification, characterization, and quantification of proteins in living cells or tissue. Proteins are essential building blocks of any living system, composed of a repertoire of twenty different amino acids. The tertiary structure of a protein determines its functions, and the sequence of amino acid residues is critical for the three-dimensional structure of a protein in a living organism [12, 13]. Therefore, research in proteomics focuses on analyzing splice variants, polymorphism, amino acid sequence analysis, post-translational modifications, cellular localization, protein interacting partners, and potential function [14]. Spatial proteomics studies protein localization and dynamics, essential for cell biology and disease research. Mass spectrometry and imaging generate extensive data, analyzed with machine learning. Reported study has shown machine learning algorithms' roles, successful applications, and challenges in this field, aiding medical and drug discovery research [15]. These technologies generate vast amounts of data that require interpretation using data analysis tools to obtain meaningful information. The determination of protein structure is also essential in biology, and the two most common forms of determining structure experimentally are X-ray crystallography and NMR. However, these methods are resource-intensive and time-consuming, requiring expert handling of data. As a result, alternative methods for determining tertiary structure from primary structure are needed. Machine learning algorithms excel at prediction and pattern recognition but often unexplainable, particularly in complex biomedical problems. GP offers a solution for explainable machine learning in bioinformatics, translating patterns into clinical actions. A study used a linear GP (LGP) algorithm for bioinformatics classification, analyzing influential features and their effects, individual or synergistic [16]. Table 3 summarizes various case studies illustrating the effective use of GP in the fields of bioinformatics and drug discovery.

2.2. Protein quantification and localization

Protein quantification and localization are crucial aspects of research in cell biology, biotechnology, and medicine [17]. The concentration of a protein plays a significant role in determining its functionality, as well as in protein expression studies and clinical diagnosis of protein dynamics in body fluids [18]. Mass spectrometric (MS) analysis is a commonly used method for determining the relative and absolute quantification of proteins. This technique utilizes the mass/charge ratio to differentiate different analytes in protein analysis. MS mass spectrometry also uses differential labeled isotope peptide surrogates, which can be produced synthetically or via an artificial protein of concatenated peptides for protein quantification [19]. To ensure successful quantification, it is essential to know which peptides are readily detectable under the MS conditions used for analysis. An in silico method GP has been applied to determine MS peptide detectability from their calculated physio-chemical properties [20]. A new GP-based method has been developed to pre-process noisy MS/MS spectra, enhancing peptide identification accuracy in proteomics research. By distinguishing noise and signal peaks, GP significantly boosts the identification rate to 99.4% with the de novo sequencing tool PEAKS, surpassing the 80.1% achieved by traditional methods. Application of the GP pre-processing method also improves peptide identification with the SEQUEST database search tool, establishing its superiority over other approaches [21]. Biomarkers are molecular agents that help in the diagnosis and prognosis of disease. Detection of biomarkers in mass spectroscopy data is a challenging task due to high dimensional and small sample size. In addressing the intricacies of spectroscopy experiments, researchers have turned to advanced computational techniques. Utilizing evolutionary machine learning, specifically GP, has proven effective in identifying biomarkers within full-scan MS data and tandem LC-MS/MS data. These GP-based machine learning strategies involve the treatment of biomarkers as features, encompassing feature

Table 3. Overview of case studies demonstrating the successful application of genetic programming in bioinformatics and drug discovery

Case study	Application area	Metrics	Specific outcomes
Mass Spectrometry Peptide Detection	Proteomics	High identification accuracy	Significantly improved peptide identification rates compared to conventional methods.
Biomarker Discovery	Cancer Research	Enhanced detection efficiency	Effective identification of biomarkers from complex mass spectrometry data.
Protein-Protein Interaction Prediction	Bioinformatics	Improved accuracy and reduced computational time	Successful prediction of functional associations among proteins.
Motif Discovery	Gene Regulation	Competitive performance against specialist models	Effective identification of gene regulatory motifs.
Drug Discovery	Pharmaceutical Research	Increased efficiency in lead identification	Accelerated the discovery of potential drug candidates.

selection, feature extraction, and classification of mass spectrometry data to facilitate biomarker detection. In an article, a robust GP framework is highlighted for surpassing leading feature selection methods in biomarker discovery from mass spectrometry data, demonstrating enhanced classification performance across diverse machine learning algorithms – a game-changing solution for addressing the crucial challenge of deriving meaningful insights from high-dimensional, sample-scarce MS datasets. Furthermore, the study introduces a GP-based approach focused on feature selection for high-dimensional symbolic regression, surpassing existing techniques by enhancing model accuracy and identifying more concise sets of relevant features [22]. Additionally, a GP-based framework is presented, seamlessly integrating feature construction and selection to outperform both standalone and state-of-the-art methods, showcasing superior classification performance with a more condensed set of informative features [23].

In addition to quantification, identifying a protein's cellular location is equally important. Various methods and techniques have been developed to address the question of protein cellular localization, including extracellular, membranous, or nuclear locations [24, 25]. Through quality-based iterative classification programs, GP has demonstrated its efficacy in determining a protein's cellular localization. One approach involves utilizing the Kyte-Doolittle [26] hydrophobicity scale and the SWAP-1 induction technique to identify trans-membrane domains in protein sequences [27], thereby refining algorithms for protein classification. Furthermore, the application of automatic function-enabled GP enables the iterative classification of proteins using statistical methods. This approach has achieved a notable 76% accuracy rate in categorizing protein sequences into distinct cellular locations such as extracellular, intracellular, nuclear, membrane integral, and membrane-anchored [28]. The success of these methods hinges on the correlation between a protein's cellular location and the prevalence of specific amino acids [29]. For instance, extracellular proteins exhibit a lower percentage of hydrophobic residues but a higher percentage of cysteine residues, crucial for forming stable extracellular disulfide bonds [30]. On the other hand, integral membrane and anchored membrane proteins are characterized by high hydrophobicity and serve essential functions within the membrane structure [31].

GP has also been used to develop methods for classifying compounds as inhibitors and non-inhibitors for cyclin-dependent kinases [32]. Additionally, evolutionary computation methods like GP and swarm intelligence have been used to solve problems of big data analytics [33]. Recent research has shown that GP can be used to understand the behavioral output of individuals. GP has yielded multi-objective results to determine the sub-cellular location of proteins with high accuracy rates [34]. The research revealed that proteins located inside the nucleus are characterized by a high presence of positively charged residues and a low presence of aromatic residues. On the other hand, proteins found within cells display relatively high levels of negatively charged and aliphatic residues but have a low content of cysteine [35]. Reported study provides a comprehensive overview of the computational prediction of sub-cellular protein localization, emphasizing the significance of proper protein distribution within cellular compartments for their functionality. It delves into various in silico methods utilized to tackle this challenge, presenting a detailed analysis of tools, input features, machine learning strategies, and evaluation metrics employed in this domain [36]. To overcome this challenge, GP offers a solution to evolve classifiers capable of simultaneously taking into account both global information, such as amino acid frequencies, and local sequence motifs. The research revealed that the motifs utilized for protein classification were not predefined; instead, they evolved concurrently with other classification rules [37]. This approach represents a low-level automation-like implementation method for evolving sequence classifiers for sub-cellular location [38]. The location of proteins on the cell surface is believed to play a crucial role in determining their immunogenicity. Certain secreted proteins, especially those released by bacteria into the extracellular environment, are considered important factors in pathogen virulence. These proteins hold promise as potential candidates for subunit vaccines [39]. Table 4 details the methods for quantification and analysis employed in GP for protein analysis, highlighting their uses and benefits.

2.3. Protein structure and function prediction

Automatically determining protein structure from the amino acid sequence has only become more important after the breakthrough developments in high throughput sequencing in the last few years. Protein structure prediction has undergone a transformation driven by neural networks, achieving accuracies of 2.1 Å [40]. A new method combines aquaphotomics and GP for precise macronutrient prediction in aquaponic systems, enabling efficient water quality monitoring and energy savings [41]. A GP-based approach offers accurate diabetic foot diagnosis, outperforming current methods with a user-friendly visualization tool [42]. Though much research has been done, inferring a protein structure ab initio by calculating the folding process using physics is difficult and also very costly computationally. A more common approach is to compare a new sequence to a database of existing structures in a process known as threading or to compare the sequence to an existing related sequence whose structure is already known. Newly developed methods educate support vector machine (SVM), which has introduced many kernels to know apart between related and unrelated sequences [43, 44]. Table 5

Table 4. Quantification and analysis techniques used in genetic programming for protein analysis, outlining their applications and advantages

Technique	Application	Advantages
Mass Spectrometry Analysis	Protein quantification	High sensitivity and specificity for peptide detection.
Machine Learning Integration	Feature extraction	Enhanced accuracy in identifying significant features.
Structural Prediction Algorithms	Protein structure prediction	Improved prediction of protein folding and stability.
Functional Analysis Models	Understanding protein functions	Ability to model complex interactions and functions.

Table 5. Overview of databases and tools used in genetic programming, including descriptions and their relevance to research applications

Resource	Description	Relevance to research
GP-Tree	A software tool for visualizing and manipulating genetic programming trees.	Facilitates the understanding of tree structures and their evolution during the genetic programming process.
ECJ (Evolutionary Computation in Java)	A widely-used framework for implementing genetic algorithms and genetic programming.	Provides a robust platform for developing and testing genetic programming algorithms in various applications.
DEAP (Distributed Evolutionary Algorithms in Python)	A Python framework for implementing evolutionary algorithms, including genetic programming.	Enables rapid prototyping and experimentation with genetic programming techniques in a user-friendly environment.
OpenAI's GPT	A state-of-the-art language model that can be used for generating code and algorithms.	Assists in automating the generation of genetic programming solutions and enhancing algorithm development.
Genetic Programming Toolkit (GPT)	A collection of tools designed specifically for implementing genetic programming.	Offers various algorithms and utilities that streamline the genetic programming process and improve efficiency.
Bioinformatics Databases (e.g., UniProtKB, GenBank)	Comprehensive databases for protein and gene sequences.	Provide essential data for training and validating genetic programming models in bioinformatics applications.

provides a summary of the databases and tools utilized in GP, along with their descriptions and significance for research applications.

Homology modeling is advancing the comprehension of the structure and function of proteins that are similar or distantly related by analyzing protein sequences through computational methods. Various approaches, such as the Smith and Waterman algorithm [45], as well as heuristic methods like BLAST [46] and FASTA [47], aim to identify sequence similarities between protein pairs. By utilizing aggregated statistics from related proteins, these methods generate more intricate models. Techniques like profile analysis [48] and Hidden Markov Models (HMMs) [49] exclusively employ related sequences in the process of model generation. Handstad and his team, as described in their study [50], have implemented a successful GP technique to unveil novel motif kernels by identifying shared occurrences of distinct sequence motifs. This GP kernel has demonstrated remarkable efficacy in tackling the intricate fold recognition challenge compared to other existing methods. The key strength lies in their methodology's ability to construct motif sets that delineate similarities within subgroups of both related and unrelated proteins.

In the realm of vast biological sequence repositories, notable entities like PROSITE [51] stand out for their comprehensive annotations. The automated discovery of patterns within these bio-sequence databases poses a significant challenge. A sequence pattern or motif acts as a distinct marker that identifies a cluster of related bio-sequences. Leveraging computational tools, these patterns can serve as a mechanism for database queries, shedding light on the fundamental biological and evolutionary characteristics shared by a group of sequences [52].

Employing diverse computational approaches, researchers utilize an array of representation languages for bio-sequence identification, including regular languages, context-free, and other languages, as well as probabilistic representations [53]. Drug discovery research faces challenges like developing new drugs during disease outbreaks and combating drug resistance. Virtual screening, powered by deep learning algorithms and neural networks, aids in identifying drug targets from large molecular databases, improving processes like peptide synthesis and toxicity prediction [54].

A new probabilistic regular motif language for protein sequences was evaluated using GP with Lamarckian evolution to evolve SRE-DNA motifs for aligned sets of protein sequences [55]. Stochastic Regular Expressions (SRE) is a probabilistic regular expression language that uses codon-level probabilities within conserved sets. It is essentially a conventional regular expression language embellished with probability fields. It is similar to a stochastic regular language where a number of mathematical properties of the language have been proven [56]. The viability of SRE-DNA, as a new motif language, and to test the practicality of logic, grammar-based GP in an application of bioinformatics has been investigated. Newly sequenced proteins are continuously deposited into the expanding global archives. Automated machine learning techniques are employed to uncover insights into their biological structure and function [57].

GP was utilized to introduce new motifs into the protein sequences of the DEAD box and manganese superoxide dismutase protein families. Despite not specifying the motif length, they successfully evolved new motifs. When tested against the SWISS-PROT database, these newly evolved consensus motifs showed the ability to detect both protein families either as well as or slightly better than existing methods. Interestingly, similar human-written motifs were already present in the PROSITE database [58].

Protein structure quality differentiation is dependent on the structure's energy. Predicting energy functions automatically is a significant challenge, but GP has been used to create innovative energy forms that can compete in advanced experiments like the CASP test [59]. The researchers employed the Nelder-Mead algorithm [60], which is known for directly optimizing a weighted sum of energies in multiple dimensions that evolved throughout the GP process. The results from GP highlighted significant variations in energy-based structural comparisons.

Oculopharyngeal muscular dystrophy (OPMD) is a neuromuscular disease characterized by muscular weakness that typically manifests in early middle age. GP was utilized to identify protein conformation defects in fluorescence microscopy images related to OPMD. The primary objective of the research was to establish specific medical criteria for detecting OPMD in microscopic images due to the varying sizes of cell images and muscular intranuclear inclusions (INIs) [61]. A bin threshold-based technique was used to filter the image backgrounds into a histogram margin for the purpose of texture features extracting from a measurable region. Reported method combines two

techniques Histogram Region of Interest Fixed by Thresholds (HRIFT) and automated feature synthesis (AFS), to capture the color of INIs and to identify OPMD by means of GP and Expectation Maximization algorithm (GP-EM) for classification improvement [62]. Computational methods and tools are used to know the function of uncharacterized protein based on the study of characterized protein families and in comparative genomics. To get the accurate descriptions of protein function experimentalist needs repeated cycles of laboratory experiments and curation of data in databases. This is, of course, a time-consuming process. The annotation of databases for new or related proteins from the same or a different organism is needed. Adequate precautions for this type of annotation to rapidly bring added value to large data otherwise be a large collection of unannotated sequences [63, 64]. An accurate portrayal of fundamental biology through human-designed computational methods alone is often insufficient, leading to a missed opportunity for crucial sequence-to-function relationships. The preparation of input data for these methods relies heavily on human knowledge and expertise. GP has been employed to demonstrate how an open-ended evolutionary algorithm can autonomously uncover features in raw amino acid sequences that are associated with protein function [65]. The evolutionary algorithm stands out in its ability to self-select target functions while learning these sequence attributes. To unveil unexpected links between cellular processes, the researchers examined protein function from a sequence perspective. A recent surprising discovery emerged from exploring the role of ubiquitination in transcription and translation [66].

The classification of proteins based on shared biological functions presents a significant challenge. A heuristic method called MAHATMA, which is founded on GP, has been developed to identify specific features within a particular protein family for the purpose of classifying proteins with unknown functional classes more effectively. The MAHATMA method integrates not only conventional GP operators but also problem-specific operators, resulting in improved specificity and hit rate with reduced sensitivity in protein classification [37]. Various computational methods exist for metabolomics and the discovery of metabolic markers [67, 68]. Identifying new proteins with high-quality traits and functions is a complex task, and numerous machine learning methods are employed for protein function prediction [69]. A novel approach called POET, utilizing GP, has been implemented to enhance screening and mutagenesis in directed evolution, facilitating the discovery of proteins with optimal functionality [70]. POET, a GP-powered computational platform, enhances directed protein evolution to discover peptides with 400% improved MRI contrast functionality. This tool empowers protein engineers to rapidly engineer novel proteins with enhanced traits and capabilities [71]. Another groundbreaking development is GADP-align, a hybrid framework of GA and Dynamic Programming that revolutionizes protein structure alignment, delivering precise pairwise alignments for challenging proteins. This computational tool unlocks new insights into protein relationships and functions [72].

2.4. Protein-protein interaction

Protein-protein interaction (PPI) networks help us to identify and understand the biological process in living systems, which control the regulatory and physiological mechanism. To solve these problems, various experimental methods have been applied since many years. But in recent years, many different computational methods have arisen to solve these problems with the aim of reducing time and costs [73]. Problems of protein-protein interactions, functional association prediction from attributes obtained from different sources and methods, are binary classification problem. This problem was tackled by traditional machine learning methods [74]. GP was applied to this domain that show the feasibility and robustness if a given pair of proteins interacts. GP has been used for prediction of functional connection of protein-protein interaction because of its potential flexibility in many aspects, such as the definition of operations [75]. To decrease the computational time and the solution size, a well-founded Tarpeian bloat control mechanism was used with GP to improve the accuracy and readability of equations evolved in protein-protein interactions [76]. Protein-protein interaction network structure helps us to understand the function of complex. In a recent study, GP was utilized to search for quality functions within a specific set of structures, enabling the automatic identification of common network properties among these structures. This innovative approach facilitated the classification and differentiation of various structure types, demonstrating its potential for further exploration in the field [77]. Future research endeavors could focus on employing GP to investigate protein-DNA, protein-Drug, and ligand interactions, potentially yielding valuable insights into the functional associations of protein-DNA interactions in cellular processes. Moreover, GP has exhibited promising results as a classifier and predictor of binding energy in protein-protein complexes linked to cancer [78].

In another study, the combination of GP and SVMs was employed to predict protein-peptide binding sites using both protein three-dimensional structures and one-dimensional sequence data. This method showcases the potential of GP in advancing predictive modeling and analysis within the realm of molecular interactions [79]. A GP-based symbolic regression approach was used to predict protein-protein interactions related to cancer. The model achieved high accuracy and generalization ability on a dataset of 135 PPI complexes. It also showed potential in discriminating cancer-related PPIs from those of other diseases [78]. Furthermore, a novel ensemble machine learning approach called SPPPred, leveraging GP for feature construction, demonstrates improved performance in predicting binding residues in protein-peptide complexes. It shows promising results on both cross-validation and independent test sets [80].

2.5. Motif discovery and RNA secondary structure predictions

Motif discovery is an important bioinformatics problem for cognizance of gene regulation. Sequence-based approaches using human specialist motif models were unable to show adequate real process. Potentiality of GP has also been employed to evolve human competitive models [81]. Their results exhibit both great challenges and potentials. No models have effectively learned and performed in a standard manner. This may be attributed to issues with data appropriateness or computational challenges in motif discovery. When widely tested different data sets come under the purview, the same models started to show corresponding performance to existing approaches based on specialist models. On the basis of their findings, we can conclude that further well-established GP approaches need to be evolved to learn different levels of effective evaluation models from strict to lose ones. For motif discovery development, various quantitative to cardinal and classification for earning feasibility needs to be done. For the evaluation of gene chip performance, GP has been used to evolve DNA motifs. To evolve DNA motifs, a new context-free grammar

method Backus Nauru form was implied with GP. The automatically produced thymine followed by one or more adenine motif is better at predicting poor DNA sequences than an existing human-generated regular expression. In this regard, GP has been implemented for pairwise sequence comparison. Their GP evaluation scheme has used pairwise sequence comparison algorithm to compare a program's output sequence and the correct sequence. The optimal edit distance between sequences is efficiently computed using dynamic programming [82]. GP has also been employed to control the complex dynamics of artificial biochemical networks (ABN). ABNs are computational models inspired by the biochemical networks which underlie the cellular activities of biological organisms. Their finding shows how evolved ABNs may be used to control chaotic dynamics in both discrete and continuous dynamical systems [83]. GP has also been applied to Gene function prediction and their localization through a mathematical discriminate function. Through the use of GP, it could have been possible to find a discriminate function that predicts the gene action into some function, and their location without experimental equipment [84]. Repetitive DNA base sequences, microsatellites, SSR tracts, ALU, etc., are instinctually found in their biological chromosomes. GP along with linear time series prediction programs has been used to discover the hierarchical repeating sequences [85]. They have observed the evolution of long repeated sequences of instructions. The chances of them being found purely at random are infinitesimal.

GP approach has been also used to find common RNA secondary structure elements through biochemical, biophysical, and phylogenetic analysis. GP has also been applied in prediction of consensus structural motifs in a family of co-regulated RNA sequences. A tool GPRM was designed that predicts the common secondary structure elements within a set of homologous RNA sequences [86]. Mi-RNAs, a class of non-coding RNA (ncRNA) typically consisting of 21–25 nucleotides, play a crucial role as negative translation regulators in multicellular organisms [87]. The structure of RNA molecules is fundamental to their function and classification. However, predicting RNA structure poses challenges due to complexities like knots and pseudo-knots formation during folding. Various computational tools, including RNAFold, have been developed to predict RNA structures [88]. Recently, a novel GP-based method has emerged as a promising approach for accurate RNA structure prediction, offering enhanced efficiency and reduced time and energy consumption compared to existing programs [89].

Both sequence-based and profile-based computational strategies have been employed to uncover the structural features of mi-RNAs [90]. Analyzing a protein sequence based on its secondary structure attributes and evolutionary conservation across diverse species can provide valuable insights into the protein's functionality, structure, and evolutionary lineage [87]. Computational methods have facilitated comparative studies of known mi-RNA precursors, shedding light on their secondary structures and sequences [91]. Evolutionary algorithms have been utilized to address the phylogenetic shadowing problem, enabling the derivation of a comprehensive conservation profile across mi-RNA precursors and flanking sequences [91].

By combining phylogenic profiles with structural filters, researchers have successfully identified novel mi-RNAs. Advanced ab initio methods controlled automatically have enabled the discovery of target-specific mi-RNAs without relying on comparative genomics approaches for sequence homology. GP has played a key role in developing specialized classifier programs, incorporating multiple regular expressions (motifs) matched against secondary structure sequences. These classifiers have been trained on fixed-length sequences, simulating the process of shifting a window in regular steps over a genomic region, addressing scanning challenges effectively [92].

The utilization of GP in machine learning has demonstrated its potential to predict the efficacy of small interfering RNAs for targeted therapeutics, thus making significant strides in the field of biomedicine. This approach addresses critical challenges associated with drug target selection, model evolution, and feature selection, setting a robust foundation for future computational research [93].

In a separate study, the article investigates the prediction of RNA secondary structures using the DP-SSP algorithm, which is grounded in dynamic programming techniques. This innovative method aims to enhance both the efficiency and reliability of RNA structure predictions through advanced generative programming design [94].

Additionally, the introduction of SgpNet, a pioneering framework that employs GP, offers a method for inferring asynchronous Boolean networks from single-cell data. SgpNet effectively aligns state transition graphs derived from the data, preserving network sparsity while achieving accuracy without imposing artificial constraints on the structures of Boolean functions. The framework also shows promise for scalability to larger networks through the application of parallelization techniques [95].

2.6. Drug discovery and cancer biology

Computational biology is playing a vital role in accelerating the processes of drug discovery and development by reducing both costs and time associated with these endeavors. This field has effectively addressed predictive pharmacokinetics by estimating the processes of adsorption, distribution, metabolism, excretion, and toxicity (ADMET) that a drug undergoes within the patient's body. A comparative study of GP and other machine learning techniques [96] assessed their ability to predict oral bioavailability (%F), median oral lethal dose (LD50), and plasma protein binding levels (%PPB). In all instances, GP outperformed linear regression and SVM with a first-degree polynomial kernel, owing to the advantages of fitness clouds and the slope coefficient fitness indicator. Furthermore, GP has been utilized in drug discovery development [97] and in the quantitative structure-activity relationship (QSAR) investigation of docking energy [98]. For example, the process of tracking the progression of symptoms in Parkinson's disease (PD) is complex and time-consuming, requiring specialized examinations in hospital clinics. The Unified Parkinson's Disease Rating Scale (UPDRS) is the most commonly used tool for tracking the progression of PD symptoms. To streamline this process, a computational intelligence method known as GP has been employed to automate the assessment of PD symptoms [99].

Early, non-invasive identification of PD is essential for effective treatment. A study that compared various machine learning techniques for diagnosing PD found that white-box approaches, such as Cartesian GP and Decision Trees, provide both accurate classification and clear models, thus enhancing interpretability for clinicians. These results can inform the development of cost-effective diagnostic protocols using handwriting and drawing samples [100].

The study aimed to improve the accuracy of PD diagnosis by enhancing decision tree induction through GAs. By integrating GP and GAs with the J48 algorithm, the classification

performance significantly increased on a real biomedical dataset, raising accuracy from 80.51% to 90.76% and surpassing traditional methods [101]. Cheminformatics is essential for managing chemical data and predicting toxic effects across a range of industries. Study introduces a quantum-inspired GP model aimed at improving the accuracy of toxicity predictions. This model outperforms traditional methods, such as neural networks, by providing more precise linear equations and enhanced selection processes through the use of quantum computing. Furthermore, dynamic modeling of metabolic pathways has been successfully achieved using GP, integrating a variety of empirical data, single nucleotide polymorphisms, and mass spectrometry data. The evolutionary capabilities of GP enable the incorporation of diverse and complex molecular data into a cohesive dynamic model [102].

Machine learning is a promising tool for exploring the relationship between genetic material and tumor pathologies and for guiding cancer treatment decisions. GP classifies colon tissues as healthy or cancerous using data from patients with acute myeloid and acute lymphoblastic leukemia [103]. Additionally, GP has classified breast cancer patients based on seventy gene expression signatures [103]. Study examines the GP symbolic classifier (GPSC) for accurately categorizing breast cancer subtypes. Despite challenges like numerous gene expressions and imbalanced datasets, the GPSC method with oversampling techniques achieved a high classification accuracy of 0.992, which improved to 0.994 with a decision tree classifier [104]. Furthermore, the text discusses addressing classification challenges in imbalanced breast cancer datasets using GP with two fitness functions: the F2 score and Distance score (D score). The models, F2GP and DGP, achieved accuracies of up to 100%, effectively distinguishing benign and malignant cases by reducing bias and focusing on minority class learning [105].

There have been several empirical studies addressing breast cancer using machine learning. GP and machine learning algorithms-based approach has reported high accuracy system to differentiate between benign and malignant breast tumors. The aim of this study was to optimize the learning algorithm [106]. Researchers have previously developed automated classification techniques based on artificial neural networks and case-based reasoning to assist physicians in interpreting mammography results. Ludwig utilized GP on BI-RADS findings data to enhance mammography prognosis. The investigation followed a two-fold approach: initially applying standard GP, followed by the development and evaluation of a distributed version alongside other existing methods. The performance of the GP method was notably strong, and the models generated by the classifiers exhibited a level of transparency that made them easy to comprehend [107]. A significant drawback of GP is the lengthy training process for classifier models. Nevertheless, researchers have effectively integrated various mathematical models, machine learning algorithms, and GP techniques to predict cancer diagnoses with high accuracy, particularly for melanoma, breast cancer, and respiratory cancers. In medical science, GP has been used to predict responses to anticancer therapies by analyzing the NCI-60 microarray dataset, revealing strong correlations between gene expression and drug responses for medications like Fluorouracil and Fludarabine [108]. This approach has demonstrated superior accuracy compared to traditional methods, prompting further exploration of GP in drug discovery and development, as well as in addressing gaps in cancer diagnosis data [109].

GP has been effectively employed for early detection and classification of breast cancer. A multi-objective GP approach analyzed digital mammograms to identify suspicious areas. GP has also enabled automatic detection of breast cancer using mammography and gene expression datasets [110]. Evolutionary programming is also showing promising results in predicting potential drugs or inhibitors for particular disease. GP was used to analyze descriptors for serine protease inhibitors of Mycobacterium tuberculosis (Mtb) and discover new inhibitors. The best model identified 126 potential anti-tubercular agents among 918 phytochemical compounds, aiding in drug development for tuberculosis [111].

Cancer remains a leading cause of death in developing countries, with oral cancer being prevalent among both men and women. Early detection is vital for improving survival rates, yet it often proves to be challenging and time-consuming. This study incorporates advanced image processing techniques, such as Gabor filters for noise reduction and GAs for tumor feature extraction and segmentation, leading to enhanced detection capabilities. Additionally, GP is recognized for its role in improving image classification solutions [112]. An automated method utilizing GP has been developed for the precise classification of retinal diseases from optical coherence tomography images. This innovative approach selects optimal feature extraction methods and parameters, achieving superior accuracy compared to traditional techniques by analyzing 800 images of retinal diseases alongside normal cases [113]. In another study, GP was evaluated for its effectiveness in predicting survival rates in oral cancer patients. The GP method outperformed both SVM and logistic regression in prognostic accuracy by identifying key features such as smoking history and histological differentiation. The automatic feature selection capability of GP makes it an invaluable tool for assisting physicians in cancer diagnosis and prognosis [114].

GP presents a dynamic and flexible method for classifying skin images by employing feature selection and construction to improve diagnostic accuracy. Unlike traditional techniques, GP-based approaches yield interpretable models that enhance performance by extracting informative features from skin images. This capability assists dermatologists in quickly and effectively identifying crucial image characteristics in real-time clinical environments [115]. Researchers have developed a GP method specifically for detecting skin malignancies, integrating feature selection and construction to boost classification performance. This approach leverages local binary patterns and wavelet decomposition to extract features from skin images, resulting in improved accuracy compared to conventional classification algorithms. The interpretability of the GP method supports dermatologists in pinpointing essential features for diagnosis [116].

GP with a novel Euclidean distance-based fitness function demonstrates superior performance in diagnosing chronic kidney disease compared to methods like K-nearest neighbors (KNN) and KNN with PSO. Using an imbalanced dataset of 400 patients, GP achieved an impressive accuracy of 99.33% and an AUC value of 0.99 through ten-fold cross-validation [117]. In another study, a Computer-Aided Diagnosis (CADx) system for Alzheimer's disease employing GP successfully classified patients by selecting discriminant features through a majority voting scheme, outperforming alternative methods. This showcases GP's effectiveness in diagnosing Alzheimer's and its suitability for CADx systems that utilize spontaneous speech analysis [118].

A comparative study between GP and Cox regression for cardiovascular risk predictions using data from the SMART study revealed that both models had similar discrimination and calibration abilities. However, GP proved to be more automated and required less expertise, highlighting its potential for developing automated clinical prediction models [119].

Additionally, a new evolutionary learning technique utilizing GP was developed to analyze mutated lung cancer genes for early diagnosis. This model effectively selected 23 features from a pool of 1500 and achieved a high accuracy of 95.67% and an AUC of 98.79%, surpassing existing prediction methods [120].

Furthermore, research on COVID-19 data using GP focused on generating mathematical models to estimate confirmed, deceased, and recovered cases. The models yielded high R^2 scores and accurately depicted epidemiological curves for specific countries and globally, closely aligning with real data [121]. Another study specifically targeted predicting COVID-19 cases in India, particularly in states like Maharashtra, Gujarat, and Delhi. This research emphasized the importance of analyzing the impact of COVID-19 in India and forecasting its trends. The developed prediction models proved reliable for time series forecasting of confirmed cases and deaths, providing valuable insights into the spread of the virus in the country [122].

3. Conclusion

This study highlights the significant rise of GP in addressing complex bioinformatics challenges, showcasing its robustness and utility over traditional evolutionary algorithms. The findings reveal that GP excels in fundamental tasks such as motif discovery, pattern recognition, and the analysis of protein structure and function. Notably, GP has demonstrated superior performance in MS peptide detectability, significantly surpassing conventional methods in identification accuracy. Moreover, the application of GP in cancer research and biomarker detection illustrates its capability to manage high-dimensional data effectively, thereby enhancing diagnostic accuracy. However, current computational limitations restrict the complexity and dynamics of problem-solving in this field. To overcome these challenges, it is essential for biologists to adopt targeted, concise approaches while computer scientists focus on refining algorithm performance.

Future research should prioritize the development of advanced evolutionary algorithms to explore protein-DNA, protein-drug, and ligand interactions, aiming for effective functional associations that could illuminate cell-cell interactions. Additionally, GP should be applied to predict actual mass spectrometry peak intensities rather than relying solely on binary classifications. The potential of new evolutionary methods extends to whole genome analysis, phylogenetic, and drug development, emphasizing the need for GP in predicting protein domains, novel folds, and deeper insights into bioinformatics. Overall, this study underscores the transformative potential of GP in advancing the fields of genomics, proteomics, and drug discovery.

Acknowledgement

I would like to extend my appreciation to Department of Bioinformatics, Pondicherry University for their computational support.

Ethical Statement

This study does not contain any studies with human or animal subjects performed by the author.

Conflicts of Interest

The author declares that he has no conflicts of interest to this work.

Data Availability Statement

Data sharing is not applicable to this article as no new data were created or analyzed in this study.

Author Contribution Statement

Mohammad Wahab Khan: Conceptualization, Methodology, Formal analysis, Investigation, Resources, Data curation, Writing – original draft, Writing – review & editing, Visualization, Supervision, Project administration.

References

[1] Koza, J. R. (1994). Genetic programming as a means for programming computers by natural selection. *Statistics and Computing*, *4*(2), 87–112. https://doi.org/10.1007/BF00175355

[2] Holland, J. H. (1992). *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*. USA: The MIT Press.

[3] Langdon, W. B. (1998). *Genetic programming and data structures: Genetic programming + data structures = automatic programming!* UK: Kluwer Academic Publishers.

[4] Neuman, W. R. (2023). *Evolutionary Intelligence: How technology will make us smarter*. USA: The MIT Press.

[5] Mei, Y., Chen, Q., Lensen, A., Xue, B., & Zhang, M. (2023). Explainable artificial intelligence by genetic programming: A survey. *IEEE Transactions on Evolutionary Computation*, *27*(3), 621–641. https://doi.org/10.1109/TEVC.2022.3225509

[6] Pillay, N. (2020). The impact of genetic programming in education. *Genetic Programming and Evolvable Machines*, *21*(1), 87–97. https://doi.org/10.1007/s10710-019-09362-4

[7] Khan, A., Qureshi, A. S., Wahab, N., Hussain, M., & Hamza, M. Y. (2021). A recent survey on the applications of genetic programming in image processing. *Computational Intelligence*, *37*(4), 1745–1778. https://doi.org/10.1111/coin.12459

[8] Kahourzade, S., Mahmoudi, A., & Mokhlis, H. B. (2015). A comparative study of multi-objective optimal power flow based on particle swarm, evolutionary programming, and genetic algorithm. *Electrical Engineering*, *97*(1), 1–12. https://doi.org/10.1007/s00202-014-0307-0

[9] Khan, M. W., & Alam, M. (2012). A survey of application: Genomics and genetic programming, a new frontier. *Genomics*, *100*(2), 65–71. https://doi.org/10.1016/j.ygeno.2012.05.014

[10] Arı, D., & Alagöz, B. B. (2021). A review of genetic programming: Popular techniques, fundamental aspects, software tools and applications. *Sakarya University Journal of Science*, *25*(2), 397–416. https://doi.org/10.16984/saufenbilder.793333

[11] Raslan, M. A., Raslan, S. A., Shehata, E. M., Mahmoud, A. S., & Sabri, N. A. (2023). Advances in the applications of bioinformatics and chemoinformatics. *Pharmaceuticals*, *16*(7), 1050. https://doi.org/10.3390/ph16071050

[12] Murray, J. E., Laurieri, N., & Delgoda, R. (2017). Proteins. In S. Badal & R. Delgoda (Eds.), *Pharmacognosy: Fundamentals, applications and strategies* (pp. 447–494). Academic Press. https://doi.org/10.1016/B978-0-12-802104-0.00024-X

[13] Sorokina, I., Mushegian, A. R., & Koonin, E. V. (2022). Is protein folding a thermodynamically unfavorable, active, energy-dependent process? *International Journal of Molecular Sciences*, *23*(1), 521. https://doi.org/10.3390/ijms23010521

[14] Cui, M., Cheng, C., & Zhang, L. (2022). High-throughput proteomics: A methodological mini-review. *Laboratory Investigation*, *102*(11), 1170–1181. https://doi.org/10.1038/s41374-022-00830-7

[15] Mou, M., Pan, Z., Lu, M., Sun, H., Wang, Y., Luo, Y., & Zhu, F. (2022). Application of machine learning in spatial proteomics. *Journal of Chemical Information and Modeling*, *62*(23), 5875–5895. https://doi.org/10.1021/acs.jcim.2c01161

[16] Hu, T. (2020). Can genetic programming perform explainable machine learning for bioinformatics? In W. Banzhaf, E. Goodman, L. Sheneman, L. Trujillo & B. Worzel (Eds.), *Genetic programming theory and practice XVII* (pp. 63–77). Springer International Publishing. https://doi.org/10.1007/978-3-030-39958-0_4

[17] Royer, C. A., Tyers, M., & Tollis, S. (2023). Absolute quantification of protein number and dynamics in single cells. *Current Opinion in Structural Biology*, *82*, 102673. https://doi.org/10.1016/j.sbi.2023.102673

[18] Hartl, J., Kurth, F., Kappert, K., Horst, D., Mülleder, M., Hartmann, G., & Ralser, M. (2023). Quantitative protein biomarker panels: A path to improved clinical practice through proteomics. *EMBO Molecular Medicine*, *15*(4), e16061. https://doi.org/10.15252/emmm.202216061

[19] Nosti, A. J., Barrio, L. C., Calderón Celis, F., Soldado, A., & Encinar, J. R. (2022). Absolute quantification of proteins using element mass spectrometry and generic standards. *Journal of Proteomics*, *256*, 104499. https://doi.org/10.1016/j.jprot.2022.104499

[20] Wedge, D. C., Gaskell, S. J., Hubbard, S. J., Kell, D. B., Lau, K. W., & Eyers, C. (2007). Peptide detectability following ESI mass spectrometry: Prediction using genetic programming. In *Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation*, 2219–2225. https://doi.org/10.1145/1276958.1277382

[21] Azari, S., Xue, B., Zhang, M., & Peng, L. (2019). Preprocessing tandem mass spectra using genetic programming for peptide identification. *Journal of the American Society for Mass Spectrometry*, *30*(7), 1294–1307. https://doi.org/10.1007/s13361-019-02196-5

[22] Al-Helali, B., Chen, Q., Xue, B., & Zhang, M. (2024). Genetic programming for feature selection based on feature removal impact in high-dimensional symbolic regression. *IEEE Transactions on Emerging Topics in Computational Intelligence*, *8*(3), 2269–2282. https://doi.org/10.1109/TETCI.2024.3369407

[23] Ma, J., & Gao, X. (2020). A filter-based feature construction and feature selection approach for classification using genetic programming. *Knowledge-Based Systems*, *196*, 105806. https://doi.org/10.1016/j.knosys.2020.105806

[24] Jiang, Y., Wang, D., Wang, W., & Xu, D. (2021). Computational methods for protein localization prediction. *Computational and Structural Biotechnology Journal*, *19*, 5834–5844. https://doi.org/10.1016/j.csbj.2021.10.023

[25] Nakai, K., & Wei, L. (2022). Recent advances in the prediction of subcellular localization of proteins and related topics. *Frontiers in Bioinformatics*, *2*, 910531. https://doi.org/10.3389/fbinf.2022.910531

[26] Kyte, J., & Doolittle, R. F. (1982). A simple method for displaying the hydropathic character of a protein. *Journal of Molecular Biology*, *157*(1), 105–132. https://doi.org/10.1016/0022-2836(82)90515-0

[27] Weiss, S. M., Cohen, D. M., & Indurkhya, N. (1993). Transmembrane segment prediction from protein sequence data. In *Proceedings. International Conference on Intelligent Systems for Molecular Biology, 1*, 420–428.

[28] Koza, J. R., & Andre, D. (1996). Classifying protein segments as transmembrane domains using architecture-altering operations in genetic programming. In P. J. Angeline & K. E. Kinnear (Eds.), Advances in genetic programming (pp. 155–176). The MIT Press. https://doi.org/10.7551/mitpress/1109.003.0013

[29] Cedano, J., Aloy, P., Perez-Pons, J. A., & Querol, E. (1997). Relation between amino acid composition and cellular location of proteins. *Journal of Molecular Biology*, *266*(3), 594–600. https://doi.org/10.1006/jmbi.1996.0804

[30] Karamanos, N. K., Theocharis, A. D., Piperigkou, Z., Manou, D., Passi, A., Skandalis, S. S., . . . , & Onisto, M. (2021). A guide to the composition and functions of the extracellular matrix. *The FEBS Journal*, *288*(24), 6850–6912. https://doi.org/10.1111/febs.15776

[31] Hegde, R. S., & Keenan, R. J. (2022). The mechanisms of integral membrane protein biogenesis. *Nature Reviews Molecular Cell Biology*, *23*(2), 107–124. https://doi.org/10.1038/s41580-021-00413-2

[32] Abdelbaky, I. Z., Al-Sadek, A. F., & Badr, A. A. (2018). Applying machine learning techniques for classifying cyclin-dependent kinase inhibitors. *International Journal of Advanced Computer Science and Applications*, *9*(11), 229–235. https://doi.org/10.14569/IJACSA.2018.091132

[33] Cheng, S., Ma, L., Lu, H., Lei, X., & Shi, Y. (2021). Evolutionary computation for solving search-based data analytics problems. *Artificial Intelligence Review*, *54*(2), 1321–1348. https://doi.org/10.1007/s10462-020-09882-x

[34] Galván, E., Trujillo, L., & Stapleton, F. (2022). Semantics in multi-objective genetic programming. *Applied Soft Computing*, *115*, 108143. https://doi.org/10.1016/j.asoc.2021.108143

[35] Bhushan, V., & Nita-Lazar, A. (2024). Recent advancements in subcellular proteomics: Growing impact of organellar protein niches on the understanding of cell biology. *Journal of Proteome Research*, *23*(8), 2700–2722. https://doi.org/10.1021/acs.jproteome.3c00839

[36] Kumar, R., & Dhanda, S. K. (2020). Bird eye view of protein subcellular localization prediction. *Life*, *10*(12), 347. https://doi.org/10.3390/life10120347

[37] Tsunoda, D. F., Freitas, A. A., & Lopes, H. S. (2011). A genetic programming method for protein motif discovery and protein classification. *Soft Computing*, *15*(10), 1897–1908. https://doi.org/10.1007/s00500-010-0624-9

[38] Heddad, A., Brameier, M., & MacCallum, R. M. (2004). Evolving regular expression-based sequence classifiers for protein nuclear localisation. In *Applications of Evolutionary Computing: EvoWorkshops 2004: EvoBIO, EvoCOMNET, EvoHOT, EvoIASP, EvoMUSART, and EvoSTOC*, 31–40. https://doi.org/10.1007/978-3-540-24653-4_4

[39] D'Angelo, G., Pilla, R., Tascini, C., & Rampone, S. (2019). A proposal for distinguishing between bacterial and viral meningitis using genetic programming and decision trees. *Soft Computing*, *23*(22), 11775–11791. https://doi.org/10.1007/s00500-018-03729-y

[40] AlQuraishi, M. (2021). Machine learning in protein structure prediction. *Current Opinion in Chemical Biology*, *65*, 1–8. https://doi.org/10.1016/j.cbpa.2021.04.005

[41] Concepcion II, R., Lauguico, S., Alejandrino, J., Dadios, E., Sybingco, E., & Bandala, A. (2022). Aquaphotomics determination of nutrient biomarker for spectrophotometric parameterization of crop growth primary macronutrients using genetic programming. *Information Processing in Agriculture*, *9*(4), 497–513. https://doi.org/10.1016/j.inpa.2021.12.007

[42] D'Angelo, G., Della-Morte, D., Pastore, D., Donadel, G., de Stefano, A., & Palmieri, F. (2023). Identifying patterns in multiple biomarkers to diagnose diabetic foot using an explainable genetic programming-based approach. *Future Generation Computer Systems*, *140*, 138–150. https://doi.org/10.1016/j.future.2022.10.019

[43] Varadi, M., Bordin, N., Orengo, C., & Velankar, S. (2023). The opportunities and challenges posed by the new generation of deep learning-based protein structure predictors. *Current Opinion in Structural Biology*, *79*, 102543. https://doi.org/10.1016/j.sbi.2023.102543

[44] Jisna, V. A., & Jayaraj, P. B. (2021). Protein structure prediction: Conventional and deep learning perspectives. *The Protein Journal*, *40*(4), 522–544. https://doi.org/10.1007/s10930-021-10003-y

[45] Smith, T. F., & Waterman, M. S. (1981). Identification of common molecular subsequences. *Journal of Molecular Biology*, *147*(1), 195–197. https://doi.org/10.1016/0022-2836(81)90087-5

[46] Altschul, S. F., Madden, T. L., Schäffer, A. A., Zhang, J., Zhang, Z., Miller, W., & Lipman, D. J. (1997). Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic Acids Research*, *25*(17), 3389–3402. https://doi.org/10.1093/nar/25.17.3389

[47] Pearson, W. R. (1990). Rapid and sensitive sequence comparison with FASTP and FASTA. *Methods in Enzymology*, *183*, 63–98. https://doi.org/10.1016/0076-6879(90)83007-V

[48] Gribskov, M., & Veretnik, S. (1996). Identification of sequence patterns with profile analysis. In R. F. Doolittle (Ed.), *Methods in enzymology* (pp. 198–212). Academic Press. https://doi.org/10.1016/S0076-6879(96)66015-7

[49] Mardikoraem, M., Wang, Z., Pascual, N., & Woldring, D. (2023). Generative models for protein sequence modeling: Recent advances and future directions. *Briefings in Bioinformatics*, *24*(6), bbad358. https://doi.org/10.1093/bib/bbad358

[50] Håndstad, T., Hestnes, A. J. H., & Sætrom, P. (2007). Motif kernel generated by genetic programming improves remote homology and fold detection. *BMC Bioinformatics*, *8*(1), 23. https://doi.org/10.1186/1471-2105-8-23

[51] Sigrist, C. J. A., de Castro, E., Cerutti, L., Cuche, B. A., Hulo, N., Bridge, A., …, & Xenarios, I. (2013). New and continuing developments at PROSITE. *Nucleic Acids Research*, *41*(D1), D344–D347. https://doi.org/10.1093/nar/gks1067

[52] Gathani, S., Lim, P., & Battle, L. (2020). Debugging database queries: A survey of tools, techniques, and users. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, 1–16. https://doi.org/10.1145/3313831.3376485

[53] Joseph, P. (2023). Comparison of different motif discovery algorithms. *International Journal for Multidisciplinary Research*, *5*(5), 7217. https://doi.org/10.36948/ijfmr.2023.v05i05.7217

[54] Parvatikar, P. P., Patil, S., Khaparkhuntikar, K., Patil, S., Singh, P. K., Sahana, R., …, & Raghu, A. V. (2023). Artificial intelligence: Machine learning approach for screening large database and drug discovery. *Antiviral Research*, *220*, 105740. https://doi.org/10.1016/j.antiviral.2023.105740

[55] Ross, B. J. (2000). Probabilistic pattern matching and the evolution of stochastic regular expressions. *Applied Intelligence*, *13*(3), 285–300. https://doi.org/10.1023/A:1026524328760

[56] Ross, B. J. (2002). Evolving protein motifs using a stochastic regular language with codon-level probabilities. In *The International Conference on Artificial Intelligence and Soft Computing*.

[57] Kouba, P., Kohout, P., Haddadi, F., Bushuiev, A., Samusevich, R., Sedlar, J., …, & Mazurenko, S. (2023). Machine learning-guided protein engineering. *ACS Catalysis*, *13*(21), 13863–13895. https://doi.org/10.1021/acscatal.3c02743

[58] Koza, J. R., & Andre, D. (1999). Automatic discovery of protein motifs using genetic programming. In X. Yao (Ed.), *Evolutionary computation: Theory and applications* (pp. 171–197). World Scientific. https://doi.org/10.1142/9789812817471_0005

[59] Widera, P., Garibaldi, J. M., & Krasnogor, N. (2010). GP challenge: Evolving energy function for protein structure prediction. *Genetic Programming and Evolvable Machines*, *11*(1), 61–88. https://doi.org/10.1007/s10710-009-9087-0

[60] Kolda, T. G., Lewis, R. M., & Torczon, V. (2003). Optimization by direct search: New perspectives on some classical and modern methods. *SIAM Review*, *45*(3), 385–482. https://doi.org/10.1137/S003614450242889

[61] Guo, P. F., & Bhattacharya, P. (2009). An evolutionary approach to feature function generation in application to biomedical image patterns. In *Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation*, 1883–1884. https://doi.org/10.1145/1569901.1570216

[62] Guo, P., & Bhattacharya, P. (2013). Detection of protein conformation defects from fluorescence microscopy images. *Engineering Applications of Artificial Intelligence*, *26*(8), 1936–1941. https://doi.org/10.1016/j.engappai.2013.05.007

[63] Yan, T. C., Yue, Z. X., Xu, H. Q., Liu, Y. H., Hong, Y. F., Chen, G. X., …, & Xie, T. (2023). A systematic review of state-of-the-art strategies for machine learning-based protein function prediction. *Computers in Biology and Medicine*, *154*, 106446. https://doi.org/10.1016/j.compbiomed.2022.106446

[64] Notin, P., Rollins, N., Gal, Y., Sander, C., & Marks, D. (2024). Machine learning for functional protein design. *Nature Biotechnology*, *42*(2), 216–228. https://doi.org/10.1038/s41587-024-02127-0

[65] Brameier, M., Haan, J., Krings, A., & MacCallum, R. M. (2006). Automatic discovery of cross-family sequence features associated with protein function. *BMC Bioinformatics*, *7*, 16. https://doi.org/10.1186/1471-2105-7-16

[66] Dougherty, S. E., Maduka, A. O., Inada, T., & Silva, G. M. (2020). Expanding role of ubiquitin in translational control. *International Journal of Molecular Sciences*, *21*(3), 1151. https://doi.org/10.3390/ijms21031151

[67] Lee, M. Y., & Hu, T. (2019). Computational methods for the discovery of metabolic markers of complex traits. *Metabolites*, *9*(4), 66. https://doi.org/10.3390/metabo9040066

[68] Öztürk, C., Tarım, M., & Arslan, S. (2020). Feature selection and classification of metabolomics data using artificial bee colony programming (ABCP). *International Journal of Data Mining and Bioinformatics*, *23*(2), 101–118. https://doi.org/10.1504/IJDMB.2020.107378

[69] Bonetta, R., & Valentino, G. (2020). Machine learning techniques for protein function prediction. *Proteins: Structure, Function, and Bioinformatics*, *88*(3), 397–413. https://doi.org/10.1002/prot.25832

[70] Miralavy, I., Bricco, A. R., Gilad, A. A., & Banzhaf, W. (2022). Using genetic programming to predict and optimize protein function. *PeerJ Physical Chemistry*, *4*, e24. https://doi.org/10.7717/peerj-pchem.24

[71] Bricco, A. R., Miralavy, I., Bo, S., Perlman, O., Korenchan, D. E., Farrar, C. T., . . . , & Gilad, A. A. (2023). A genetic programming approach to engineering MRI reporter genes. *ACS Synthetic Biology*, *12*(4), 1154–1163. https://doi.org/10.1021/acssynbio.2c00648

[72] Mirzaei, S., Razmara, J., & Lotfi, S. (2021). GADP-align: A genetic algorithm and dynamic programming-based method for structural alignment of proteins. *BioImpacts: BI*, *11*(4), 271–279. https://doi.org/10.34172/bi.2021.37

[73] Durham, J., Zhang, J., Humphreys, I. R., Pei, J., & Cong, Q. (2023). Recent advances in predicting and modeling protein–protein interactions. *Trends in Biochemical Sciences*, *48*(6), 527–538. https://doi.org/10.1016/j.tibs.2023.03.003

[74] Tang, T., Zhang, X., Liu, Y., Peng, H., Zheng, B., Yin, Y., & Zeng, X. (2023). Machine learning on protein–protein interaction prediction: Models, challenges and trends. *Briefings in Bioinformatics*, *24*(2), bbad076. https://doi.org/10.1093/bib/bbad076

[75] Garcia, B., Aler, R., Ledezma, A., & Sanchis, A. (2008). Protein-protein functional association prediction using genetic programming. In *Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation*, 347–348. https://doi.org/10.1145/1389095.1389156

[76] Poli, R. (2003). A simple but theoretically-motivated method to control bloat in genetic programming. In *European Conference on Genetic Programming: 6th European Conference*, 204–217. https://doi.org/10.1007/3-540-36599-0_19

[77] Reid, F., & Hurley, N. (2011). Analysing structure in complex networks using quality functions evolved by genetic programming. In *Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation*, 283–290. https://doi.org/10.1145/2001576.2001616

[78] Vyas, R., Bapat, S., Goel, P., Karthikeyan, M., Tambe, S. S., & Kulkarni, B. D. (2018). Application of genetic programming (GP) formalism for building disease predictive models from protein-protein interactions (PPI) data. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, *15*(1), 27–37. https://doi.org/10.1109/TCBB.2016.2621042

[79] Shafiee, S., & Fathi, A. (2021). Combination of genetic programming and support vector machine-based prediction of protein-peptide binding sites with sequence and structure-based features. *Journal of Computing and Security*, *8*(1), 45–63. https://doi.org/10.22108/jcs.2021.126817.1062

[80] Shafiee, S., Fathi, A., & Taherzadeh, G. (2023). SPPPred: Sequence-based protein-peptide binding residue prediction using genetic programming and ensemble learning. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, *20*(3), 2029–2040. https://doi.org/10.1109/TCBB.2022.3230540

[81] Lo, L. Y., Chan, T. M., Lee, K. H., & Leung, K. S. (2010). Challenges rising from learning motif evaluation functions using genetic programming. In *Proceedings of the 12th Annual Conference on Genetic and Evolutionary Computation*, 171–178. https://doi.org/10.1145/1830483.1830515

[82] Langdon, W. B., & Harrison, A. P. (2009). Evolving DNA motifs to predict GeneChip probe performance. *Algorithms for Molecular Biology*, *4*(1), 6. https://doi.org/10.1186/1748-7188-4-6

[83] Lones, M. A., Tyrrell, A. M., Stepney, S., & Caves, L. S. (2010). Controlling complex dynamics with artificial biochemical networks. In *Genetic Programming: 13th European Conference*, 159–170. https://doi.org/10.1007/978-3-642-12148-7_14

[84] Werner, J. C., & Fogarty, T. C. (2001). Genetic programming applied to gene function identification. In *Cup 2001 of The Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

[85] Langdon, W. B., & Barrett, S. J. (2005). Genetic programming in data mining for drug discovery. In A. Ghosh & L. C. Jain (Eds.), *Evolutionary computation in data mining* (pp. 211–235). Springer. https://doi.org/10.1007/3-540-32358-9_10

[86] Hu, Y. J. (2003). GPRM: A genetic programming approach to finding common RNA secondary structure elements. *Nucleic Acids Research*, *31*(13), 3446–3449. https://doi.org/10.1093/nar/gkg521

[87] Shang, R., Lee, S., Senavirathne, G., & Lai, E. C. (2023). microRNAs in action: Biogenesis, function and regulation. *Nature Reviews Genetics*, *24*(12), 816–833. https://doi.org/10.1038/s41576-023-00611-y

[88] Chen, L., Heikkinen, L., Wang, C., Yang, Y., Sun, H., & Wong, G. (2019). Trends in the development of miRNA bioinformatics tools. *Briefings in Bioinformatics*, *20*(5), 1836–1852. https://doi.org/10.1093/bib/bby054

[89] Langdon, W. B., Petke, J., & Lorenz, R. (2018). Evolving better RNAfold structure prediction. In *Genetic Programming: 21st European Conference*, 220–236. https://doi.org/10.1007/978-3-319-77553-1_14

[90] Riolo, G., Cantara, S., Marzocchi, C., & Ricci, C. (2021). miRNA targets: From prediction tools to experimental validation. *Methods and Protocols*, *4*(1), 1. https://doi.org/10.3390/mps4010001

[91] Berezikov, E., Guryev, V., van de Belt, J., Wienholds, E., Plasterk, R. H. A., & Cuppen, E. (2005). Phylogenetic shadowing and computational identification of human microRNA genes. *Cell*, *120*(1), 21–24. https://doi.org/10.1016/j.cell.2004.12.031

[92] Brameier, M., & Wiuf, C. (2007). Ab initio identification of human microRNAs based on structure motifs. *BMC Bioinformatics*, *8*(1), 478. https://doi.org/10.1186/1471-2105-8-478

[93] Martinelli, D. D. (2024). Machine learning for siRNA efficiency prediction: A systematic review. *Health Sciences Review*, *11*, 100157. https://doi.org/10.1016/j.hsr.2024.100157

[94] Shi, H., & Jing, X. (2022). Efficient generation of RNA secondary structure prediction algorithm under PAR

framework. *Frontiers in Plant Science*, *12*, 830042. https://doi.org/10.3389/fpls.2021.830042

[95] Gao, S., Sun, C., Xiang, C., Qin, K., & Lee, T. H. (2022). Learning asynchronous Boolean networks from single-cell data using multiobjective cooperative genetic programming. *IEEE Transactions on Cybernetics*, *52*(5), 2916–2930. https://doi.org/10.1109/TCYB.2020.3022430

[96] Barrett, S. J., & Langdon, W. B. (2006). Advances in the application of machine learning techniques in drug discovery, design and development. In *Applications of Soft Computing: Recent Trends*, 99–110. https://doi.org/10.1007/978-3-540-36266-1_10

[97] Archetti, F., Lanzeni, S., Messina, E., & Vanneschi, L. (2007). Genetic programming for computational pharmacokinetics in drug discovery and development. *Genetic Programming and Evolvable Machines*, *8*(4), 413–432. https://doi.org/10.1007/s10710-007-9040-z

[98] Archetti, F., Giordani, I., & Vanneschi, L. (2010). Genetic programming for QSAR investigation of docking energy. *Applied Soft Computing*, *10*(1), 170–182. https://doi.org/10.1016/j.asoc.2009.06.013

[99] Castelli, M., Vanneschi, L., & Silva, S. (2014). Prediction of the unified Parkinson's disease rating scale assessment using a genetic programming system with geometric semantic genetic operators. *Expert Systems with Applications*, *41*(10), 4608–4616. https://doi.org/10.1016/j.eswa.2014.01.018

[100] Parziale, A., Senatore, R., Della Cioppa, A., & Marcelli, A. (2021). Cartesian genetic programming for diagnosis of Parkinson disease through handwriting analysis: Performance vs. interpretability issues. *Artificial Intelligence in Medicine*, *111*, 101984. https://doi.org/10.1016/j.artmed.2020.101984

[101] Ghane, M., Ang, M. C., Nilashi, M., & Sorooshian, S. (2022). Enhanced decision tree induction using evolutionary techniques for Parkinson's disease classification. *Biocybernetics and Biomedical Engineering*, *42*(3), 902–920. https://doi.org/10.1016/j.bbe.2022.07.002

[102] Darwish, S. M., Shendi, T. A., & Younes, A. (2019). Quantum-inspired genetic programming model with application to predict toxicity degree for chemical compounds. *Expert Systems*, *36*(4), e12415. https://doi.org/10.1111/exsy.12415

[103] Vanneschi, L., Archetti, F., Castelli, M., & Giordani, I. (2009). Classification of oncologic data with genetic programming. *Journal of Artificial Evolution and Applications*, *2009*(1), 848532. https://doi.org/10.1155/2009/848532

[104] Anđelić, N., & Baressi Šegota, S. (2023). Development of symbolic expressions ensemble for breast cancer type classification using genetic programming symbolic classifier and decision tree classifier. *Cancers*, *15*(13), 3411. https://doi.org/10.3390/cancers15133411

[105] Devarriya, D., Gulati, C., Mansharamani, V., Sakalle, A., & Bhardwaj, A. (2020). Unbalanced breast cancer data classification using novel fitness functions in genetic programming. *Expert Systems with Applications*, *140*, 112866. https://doi.org/10.1016/j.eswa.2019.112866

[106] Dhahri, H., Al Maghayreh, E., Mahmood, A., Elkilani, W., & Faisal Nagi, M. (2019). Automated breast cancer diagnosis based on machine learning algorithms. *Journal of Healthcare Engineering*, *2019*(1), 4253641. https://doi.org/10.1155/2019/4253641

[107] Ludwig, S. A. (2010). Prediction of breast cancer biopsy outcomes using a distributed genetic programming approach. In *Proceedings of the 1st ACM International Health Informatics Symposium*, 694–699. https://doi.org/10.1145/1882992.1883099

[108] Archetti, F., Giordani, I., & Vanneschi, L. (2010). Genetic programming for anticancer therapeutic response prediction using the NCI-60 dataset. *Computers & Operations Research*, *37*(8), 1395–1405. https://doi.org/10.1016/j.cor.2009.02.015

[109] Moreno-Torres, J. G., Llorà, X., Goldberg, D. E., & Bhargava, R. (2013). Repairing fractures between data using genetic programming-based feature extraction: A case study in cancer diagnosis. *Information Sciences*, *222*, 805–823. https://doi.org/10.1016/j.ins.2010.09.018

[110] Nandi, R. J., Nandi, A. K., Rangayyan, R. M., & Scutt, D. (2006). Classification of breast masses in mammograms using genetic programming and feature selection. *Medical and Biological Engineering and Computing*, *44*(8), 683–694. https://doi.org/10.1007/s11517-006-0077-6

[111] Kumari, M., Tiwari, N., & Subbarao, N. (2020). A genetic programming-based approach to identify potential inhibitors of serine protease of Mycobacterium tuberculosis. *Future Medicinal Chemistry*, *12*(2), 147–159. https://doi.org/10.4155/fmc-2018-0560

[112] Joat, R., Thakare, A. P., Kalele, K., & Thakare, V. (2018). Genetic programming approach for oral cancer detection and its image restoration. *International Journal of Trend in Scientific Research and Development*, *2*(3), 2422–2426. https://doi.org/10.31142/ijtsrd12787

[113] Abdulrahman, H., & Khatib, M. (2020). Classification of retina diseases from OCT using genetic programming. *International Journal of Computer Applications*, *177*(45), 41–46. https://doi.org/10.5120/ijca2020919973

[114] Tan, M. S., Tan, J. W., Chang, S. W., Yap, H. J., Abdul Kareem, S., & Zain, R. B. (2016). A genetic programming approach to oral cancer prognosis. *PeerJ*, *4*, e2482. https://doi.org/10.7717/peerj.2482

[115] Ain, Q. U., Al-Sahaf, H., Xue, B., & Zhang, M. (2022). Genetic programming for automatic skin cancer image classification. *Expert Systems with Applications*, *197*, 116680. https://doi.org/10.1016/j.eswa.2022.116680

[116] Ain, Q. U., Al-Sahaf, H., Xue, B., & Zhang, M. (2023). Automatically diagnosing skin cancers from multimodality images using two-stage genetic programming. *IEEE Transactions on Cybernetics*, *53*(5), 2727–2740. https://doi.org/10.1109/TCYB.2022.3182474

[117] Kumar, A., Sinha, N., Bhardwaj, A., & Goel, S. (2022). Clinical risk assessment of chronic kidney disease patients using genetic programming. *Computer Methods in Biomechanics and Biomedical Engineering*, *25*(8), 887–895. https://doi.org/10.1080/10255842.2021.1985476

[118] Nasrolahzadeh, M., Rahnamayan, S., & Haddadnia, J. (2022). Alzheimer's disease diagnosis using genetic programming based on higher order spectra features. *Machine Learning with Applications*, *7*, 100225. https://doi.org/10.1016/j.mlwa.2021.100225

[119] Bannister, C. A., Halcox, J. P., Currie, C. J., Preece, A., & Spasić, I. (2018). A genetic programming approach to development of clinical prediction models: A case study in symptomatic cardiovascular disease. *PLoS ONE*, *13*(9), e0202685. https://doi.org/10.1371/journal.pone.0202685

[120] Sattar, M., Majid, A., Kausar, N., Bilal, M., & Kashif, M. (2022). Lung cancer prediction using multi-gene genetic programming by selecting automatic features from amino acid sequences. *Computational Biology and Chemistry*, *98*, 107638. https://doi.org/10.1016/j.compbiolchem.2022.107638

[121] Anđelić, N., Baressi Šegota, S., Lorencin, I., Mrzljak, V., & Car, Z. (2021). Estimation of COVID-19 epidemic curves using genetic programming algorithm. *Health Informatics Journal*, *27*(1), 1460458220976728. https://doi.org/10.1177/1460458220976728

[122] Salgotra, R., Gandomi, M., & Gandomi, A. H. (2020). Time series analysis and forecast of the COVID-19 pandemic in India using genetic programming. *Chaos, Solitons & Fractals*, *138*, 109945. https://doi.org/10.1016/j.chaos.2020.109945

How to Cite: Khan, M. W. (2024). A Comprehensive Survey of Genetic Programming Applications in Modern Biological Research. *Medinformatics*. https://doi.org/10.47852/bonviewMEDIN42023692