

Anirban Mukhopadhyay
Sumanta Ray
Ujjwal Maulik
Sanghamitra Bandyopadhyay

Multiobjective Optimization Algorithms for Bioinformatics


 Springer


Multiobjective Optimization Algorithms for Bioinformatics


Anirban Mukhopadhyay • Sumanta Ray •
Ujjwal Maulik • Sanghamitra Bandyopadhyay


Multiobjective Optimization Algorithms for Bioinformatics

 Springer

Anirban Mukhopadhyay 
Department of Computer Science and
Engineering
University of Kalyani
Kalyani, West Bengal, India

Ujjwal Maulik 
Department of Computer Science and
Engineering
Jadavpur University
Kolkata, West Bengal, India

Sumanta Ray 
Department of Computer Science and
Engineering
Ghani Khan Choudhury Institute of
Engineering and Technology
Malda, West Bengal, India

Sanghamitra Bandyopadhyay 
Machine Intelligence Unit
Indian Statistical Institute
Kolkata, West Bengal, India

ISBN 978-981-97-1630-2 ISBN 978-981-97-1631-9 (eBook)
<https://doi.org/10.1007/978-981-97-1631-9>

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2024

This work is subject to copyright. All rights are solely and exclusively licensed by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors, and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd. The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

If disposing of this product, please recycle the paper.

*To my parents Shri Somnath Mukhopadhyay
and Smt Manjusri Mukhopadhyay, my wife
Anindita, and my son Agnibho
Anirban Mukhopadhyay*

*To my parents Shri Satinath Ray and Smt
Santwana Ray, my wife Snehalika, and my
son Syamantak
Sumanta Ray*

*To my parents Smt Gouri Maulik and Shri
Manoj Kumar Maulik, all my students, and
my son Utsav
Ujjwal Maulik*

*To the loving memory of my parents, Smt
Bandana Banerjee and Shri Satyendra Nath
Banerjee
Sanghamitra Bandyopadhyay*

Preface

A multiobjective optimization problem refers to a class of optimization problems where the goal is to search for solutions that simultaneously optimize multiple, often conflicting, objective functions. This type of problem arises in diverse fields such as engineering, data mining, operations research, manufacturing, robotics, network design, bioinformatics, and beyond, where decision-makers are confronted with the challenge of finding solutions that achieve the best possible trade-offs among a set of competing objectives. This book explores the application of multiobjective evolutionary and other nature-inspired optimization algorithms in the particular field of bioinformatics. The book investigates the principles, methods, and practical aspects of using multiobjective optimization algorithms to solve complex and multifaceted bioinformatics problems.

The goal of bioinformatics and computational biology is to employ data analysis for understanding biological systems and processes. The biological data are often high-dimensional, noisy, and complex, making traditional optimization and data analysis techniques less effective. The motivation behind this book lies in the necessity for sophisticated tools that can handle the complexities and challenges of bioinformatics problems. Multiobjective optimization has been proved to be a promising approach to address these issues. They consider multiple objectives, enabling us to simultaneously optimize conflicting criteria. This is particularly beneficial in situations where choices for different aspects of the problem must be balanced.

This book extensively explores diverse applications of multiobjective evolutionary and nature-inspired optimization techniques, particularly Genetic Algorithms (GA), Particle Swarm Optimization (PSO), and Differential Evolution (DE) in the field of bioinformatics during the last decade. It starts with a gentle introduction in Chap. 1 to the field of multiobjective optimization and its applications in the bioinformatics domain. The subsequent chapters navigate through a spectrum of cutting-edge domains and their applications.

Chapter 2 delves into the concept of multiobjective fuzzy clustering, placing a special emphasis on interactive multiobjective fuzzy clustering (IMOC). IMOC stands out as it involves human decision makers in the clustering process to

iteratively select the most suitable cluster validity indices. It leverages the Non-dominated Sorting Genetic Algorithm (NSGA-II) and utilizes visualization tools to craft resilient gene expression data clusters. The experiments detailed in this chapter provide conclusive evidence of IMOC's exceptional capability in identifying gene clusters that hold significant biological relevance, surpassing other existing methods.

Chapter 3 addresses the problem of multiobjective rank aggregation in the context of gene prioritization. It provides an in-depth investigation of diverse rank aggregation techniques and the diverse distance measures employed for ranking comparisons. The chapter goes on to elucidate the specific objective functions tailored for multiobjective rank aggregation and introduces a novel approach utilizing multiobjective Particle Swarm Optimization (PSO) to address this challenge. Readers will find comprehensive insights through the presentation of experimental results, datasets, and extensive discussions, collectively offering a comprehensive assessment of the effectiveness of multiobjective rank aggregation in the realm of gene prioritization.

Chapter 4 introduces a pioneering multiobjective approach that simultaneously ranks and clusters genes from a microarray gene expression dataset. The chapter provides a comprehensive explanation of this multiobjective approach. We discuss the application of this technique on several real-life gene expression data. The inclusion of experimental results, a comparative analysis against existing techniques, and in-depth discussions sheds light on the practical implications of this approach in the realms of gene ranking and clustering.

In Chap. 5, our focus shifts to multiobjective feature selection for the identification of MicroRNA markers. The chapter commences with an introduction to the multiobjective feature selection process and proceeds to discuss the encoding scheme, initialization, objective computation, reproduction techniques, archive maintenance, and the selection of the final solution within this method. Readers will find detailed experimental results that include a comparative analysis with alternative methods, datasets, and insightful discussions that underscore the significance of this feature selection approach in the realm of microRNA marker identification.

Chapter 6 introduces DiffCoMO, a multiobjective framework designed for the identification of differential microRNA coexpression modules using microarray datasets representing different phenotypes. DiffCoMO distinguishes itself through two distinct objective functions, showcasing its superior ability to capture differential coexpression patterns when compared to other algorithms. The chapter further delves into the co-regulation patterns of transcription factors (TFs) within these modules and their connections to diseases, underscoring DiffCoMO's potential in identifying disease-specific microRNA families and TFs with noteworthy microRNA associations.

In Chap. 7, a multiobjective algorithm for the detection of differentially co-expressed modules in miRNA expression data from various tissues is presented. This algorithm addresses the problem of module detection by constructing a fully connected differentially co-expressed network from the expression data and employing a multiobjective optimization approach. This approach simultaneously

minimizes differential dissimilarity and functional dissimilarity among miRNA pairs. By leveraging the NSGA-II-based technique, the algorithm effectively identifies critical miRNA subsets. Comparative assessments reveal its superior module detection capabilities, and subsequent biological analyses uncover robust associations between these miRNAs and various cancer types, underscoring the algorithm's potential significance.

Chapter 8 explores a multiobjective approach for feature selection in the prediction of protein subcellular localization based on sequence data. The study initiates by collecting apoptosis protein sequences from various subcellular locations and considering diverse amino acid compositions to construct feature sets. A multiobjective particle swarm optimization (PSO) approach is employed to identify a concise, non-redundant set of informative features. The performance of this approach is evaluated by comparing it with single-objective methods such as sequential forward search, sequential backward search, and mRMR schemes.

In Chap. 9, we examine various semantic similarity measures to evaluate their effectiveness in distinguishing between interacting and non-interacting protein pairs within Protein-Protein Interaction (PPI) data across different Gene Ontology (GO) domains. Within this chapter, we present a multiobjective feature selection approach that relies on Differential Evolution (DEMO). This method is employed to identify the most compact subset of similarity measures for both the entire GO domain and three distinct individual GO domains. The performance of the DEMO-based feature selection algorithm is assessed in comparison to other established techniques, consistently demonstrating its effectiveness.

Chapter 10 is centered on the identification of protein complexes within the human Protein-Protein Interaction network and the exploration of their connections to various disorders. This chapter provides an in-depth description of a multiobjective protein complex detection technique based on NSGA-II. This technique optimizes two objective functions related to topological properties and Gene Ontology. The experimental results showcase performance comparisons against existing methods, an analysis of the predicted complexes, and their associations with disorders and diseases.

Finally, Chap. 11 introduces a multiobjective biclustering approach named MOBICLUST, applied to the HIV-1 Human Protein-Protein Interaction Network. The chapter initiates with a fundamental overview of the challenge related to mining quasi-bicliques and introduces the MOBICLUST algorithm. The experimental results highlight the effectiveness of MOBICLUST using artificial data and its practical application to the PPI dataset, shedding light on the biological relevance of the discovered quasi-bicliques.

While going through the chapters, readers will gain a comprehensive understanding of the versatile applications of multiobjective optimization in the field of bioinformatics. This book will not only be a valuable resource for researchers, practitioners, and students but will also be an invitation to explore the practical techniques that address complex challenges in the domain. Targeted for those in data mining, bioinformatics, evolutionary algorithms, and optimization, it offers insightful perspectives on using multiobjective optimization to address real-world

issues. Throughout the chapters, we thoroughly explore these techniques, providing in-depth insights into their methodologies and practical applications. Readers will be able to explore their remarkable effectiveness in addressing real-world challenges, particularly in data-rich and complex environments. The book can act as a bridge between theoretical concepts and practical applications. It encourages innovative solutions to navigate the growing complexity of data in the fields of data bioinformatics and computational biology.

We would like to extend our appreciation to Springer for considering and supporting this book project. Their support has been helpful in bringing this project to fruition. We are also grateful for the permissions generously granted by various publishers, including Springer, IEEE, and Elsevier. These permissions enable us to incorporate portions of our previously published articles in their journals into the content of this volume. Finally, we want to express our heartfelt thanks to our family members, colleagues, and students for their support throughout this endeavor.

Kalyani, India
Malda, India
Kolkata, India
Kolkata, India
24/11/2023

Anirban Mukhopadhyay
Sumanta Ray
Ujjwal Maulik
Sanghamitra Bandyopadhyay

Contents

1	Introduction	1
1.1	Concepts of Multiobjective Optimization	2
1.2	MOO in Data Mining and Machine Learning.....	5
1.2.1	Multiobjective Optimization in Clustering	6
1.2.2	Multiobjective Optimization in Classification.....	8
1.2.3	Multiobjective Optimization in Feature Selection	10
1.2.4	Multiobjective Optimization in Association Rule Mining.....	11
1.2.5	Multiobjective Optimization in Other Data Mining Tasks.....	13
1.3	Multiobjective Optimization for Bioinformatics Tasks.....	13
1.3.1	Gene Expression Analysis	14
1.3.2	Gene Clustering.....	15
1.3.3	Coexpression Clustering	16
1.3.4	Gene and MicroRNA Marker Detection.....	18
1.3.5	Module Detection in Biological Networks	19
1.4	Summary and Scope of the Book.....	20
2	Multiobjective Interactive Fuzzy Clustering for Gene Expression Data	23
2.1	Clustering and Validity Indices	24
2.1.1	Fuzzy C-means Clustering	24
2.1.2	Hierarchical Clustering	25
2.1.3	Cluster Validity Indices.....	26
2.2	Multiobjective Fuzzy Clustering	28
2.2.1	NSGA-II Algorithm	29
2.2.2	Multiobjective Clustering	30
2.3	Interactive Multiobjective Fuzzy Clustering (IMOC)	31
2.4	Experimental Results.....	33
2.4.1	Datasets for Experiments.....	34
2.4.2	Performance Measures	35

- 2.4.3 Input Parameters 36
- 2.4.4 Results and Discussion 36
- 2.4.5 Statistical Significance Test 39
- 2.5 Summary 40
- 3 Multiobjective Rank Aggregation for Gene Prioritization 41**
 - 3.1 Introduction 41
 - 3.2 Rank Aggregation Techniques 42
 - 3.2.1 MC4 Algorithm 43
 - 3.2.2 MCT Algorithm 43
 - 3.2.3 Robust Rank Aggregation 44
 - 3.2.4 Condorcet Ranking 44
 - 3.2.5 Rank Aggregation by Voting 45
 - 3.3 Distance Metrics for Ranking 45
 - 3.3.1 Kendall’s Tau Distance (τ) 45
 - 3.3.2 Spearman’s Footrule Distance (ρ) 46
 - 3.4 Objective Functions for Multiobjective Rank Aggregation 47
 - 3.5 Multiobjective PSO-based Rank Aggregation 48
 - 3.5.1 Encoding Mechanism of a Particle 49
 - 3.5.2 Initialization 49
 - 3.5.3 Computing the Fitness Values 50
 - 3.5.4 Updating the Position and Velocity 51
 - 3.5.5 Updating the Non-dominated Archive 52
 - 3.5.6 Overall Algorithm 52
 - 3.6 Experimental Results 54
 - 3.6.1 Datasets and Preprocessing 54
 - 3.6.2 Results and Discussion 56
 - 3.7 Summary 72
- 4 Multiobjective Simultaneous Gene Ranking and Clustering 75**
 - 4.1 Introduction 75
 - 4.2 Multiobjective Simultaneous Clustering and Gene Ranking 76
 - 4.2.1 Chromosome Representation and Initial Population 76
 - 4.2.2 Fitness Computation 77
 - 4.2.3 Crossover and Mutation 78
 - 4.2.4 Selection, Elitism, and Termination 78
 - 4.2.5 Final Solution Selection 79
 - 4.3 Experimental Results 79
 - 4.3.1 Experimental Design 80
 - 4.3.2 Result and Discussion 82
 - 4.4 Summary 85
- 5 Multiobjective Feature Selection for Identifying MicroRNA Markers 87**
 - 5.1 Introduction 87
 - 5.2 Multiobjective Feature Selection 88

5.2.1	Encoding Scheme and Initialization	88
5.2.2	Computing the Objectives	89
5.2.3	Reproduction Using Selection, Crossover, and Mutation	90
5.2.4	Maintaining an Archive	90
5.2.5	Selecting the Final Solution	91
5.3	Experimental Results	91
5.3.1	Comparative Methods	91
5.3.2	Datasets and Preprocessing	92
5.3.3	Evaluation Metrics	93
5.3.4	Results and Discussion	94
5.4	Summary	97
6	Multiobjective Approach to Detection of Differentially Coexpressed Modules	99
6.1	Introduction	99
6.2	DiffCoMO: Differential Coexpressed Module Detection	100
6.2.1	Differential Coexpression of Gene in Two Phenotypes ...	101
6.2.2	The DiffCoMO Framework	102
6.2.3	Evaluating Objective Functions	104
6.3	Experimental Results	104
6.3.1	Description of Dataset	105
6.3.2	Comparing DiffCoMO with Some State of the Art	106
6.3.3	Statistical Significance of Identified Modules	108
6.3.4	Performance on a Simulated Dataset	109
6.3.5	Biological Validation of Modules	111
6.3.6	Performance of DiffCoMO in Expression Data with Large Samples	115
6.4	Summary	116
7	Multiobjective Approach to Cancer-Associated MicroRNA Module Detection	119
7.1	Introduction	119
7.2	Construction of Differential Coexpression Network	121
7.3	Semantic Similarity Measure for MicroRNA Pairs	122
7.4	Multiobjective Module Detection	123
7.4.1	Chromosome Encoding	123
7.4.2	Computation of Objective Functions	124
7.4.3	Process of Obtaining Non-dominated Solutions	124
7.4.4	Obtaining the miRNA Subset from the Non-dominated Solutions	124
7.5	Experimental Results	125
7.5.1	Dataset Details and Preprocessing	125
7.5.2	Parameter Setting	126
7.5.3	Results	127

7.5.4	Statistical Significance of the Identified Module	128
7.5.5	Comparison with State-of-the-Art Algorithms	129
7.5.6	Biological Relevance Study	131
7.6	Summary	134
8	Multiobjective Approach to Prediction of Protein Subcellular Locations	135
8.1	Introduction	135
8.2	Feature Extraction from Amino Acid Sequence	137
8.3	Relevance and Redundancy of Features	137
8.4	Multiobjective PSO-Based Feature Selection Technique	138
8.4.1	Particle Encoding	139
8.4.2	Initialization and Inputs	139
8.4.3	Objective Functions	139
8.4.4	Updating Position and Velocity	140
8.4.5	Updating Archive	141
8.4.6	Final Solution Selection	141
8.4.7	Overall MOPSO Algorithm	142
8.5	Other Comparative Methods	142
8.6	Dataset and Preprocessing	144
8.7	Experimental Results	147
8.7.1	Results	148
8.7.2	Results on Independent Dataset	153
8.8	Summary	154
9	Multiobjective Approach to Gene Ontology-Based Protein-Protein Interaction Prediction	155
9.1	Introduction	155
9.2	GO-Based Semantic Similarity	156
9.2.1	Resnik Measure	157
9.2.2	Lin Measure	157
9.2.3	Jiang-Conrath Measure	157
9.2.4	Relevance Measure	158
9.2.5	Cosine Measure	158
9.2.6	Kappa Measure	158
9.2.7	Czekanowski-Dice Measure	159
9.2.8	Weighted Jaccard Measure	159
9.2.9	Graph-Based Similarity Measure	160
9.2.10	Avg, Max, Rcmx	161
9.3	Dataset Preparation	162
9.3.1	Calculation of GO-Based Semantic Similarity of Protein Pairs	162
9.3.2	Dataset Creation	163
9.4	DEMO-Based Feature Selection	164
9.4.1	Chromosome Encoding	164
9.4.2	Evaluating Chromosomes	164

9.4.3	Offspring Creation	165
9.4.4	Truncation of Population	165
9.4.5	Selecting the Final Solution	165
9.5	Experimental Results	166
9.6	Summary	168
10	Multiobjective Approach to Protein Complex Detection	171
10.1	Introduction	171
10.2	Multiobjective Protein Complex Detection	173
10.2.1	Chromosome Representation	173
10.2.2	Population Initialization	173
10.2.3	Representation of Objective Functions	173
10.2.4	Mutation Procedure	175
10.2.5	Final Solution	175
10.3	Experimental Results	176
10.3.1	Performance Comparisons Among Different Methods ...	176
10.3.2	Analysis of Predicted Complexes	179
10.3.3	Association of Predicted Complexes in Disorders/Diseases	188
10.4	Summary	191
11	Multiobjective Biclustering for Analyzing HIV-1-Human Protein-Protein Interaction Network	195
11.1	Introduction	195
11.2	Strong PPI Module Finding Using Biclustering	196
11.2.1	Biclustering	196
11.2.2	Bipartite Graph Representation of PPIN	197
11.2.3	Quasi-Biclique Finding Through Biclustering	198
11.3	Multiobjective Biclustering for Finding Quasi-Bicliques	199
11.3.1	MOBICLUST Algorithm	199
11.4	Evaluation of MOBICLUST Using Artificial Data	201
11.4.1	Preparing the Artificial Dataset	201
11.4.2	Performance Metric	201
11.4.3	Results of Comparison	202
11.5	Analysis of Quasi-Bicliques from HIV-1-Human PPIN	202
11.5.1	Preparation of the HIV-1-Human PPIN	202
11.5.2	Results of MOBICLUST Biclustering	203
11.5.3	Biological Significance of the Quasi-Bicliques	206
11.5.4	Biological Significance of the Strong Bipartite Module	208
11.6	Summary	217
	References	219
	Index	235

Chapter 1

Introduction



In recent years, significant advancements in biomedical engineering have generated a large volume of biological data, the analysis of which holds immense importance in various medical and biological contexts [1–4]. These applications encompass disease diagnosis, biomarker discovery, drug development, and forensics, playing a critical role in advancing our understanding of the biological world. Typically, these datasets exhibit high dimensionality with a vast number of features and often encapsulate intricate, nonlinear patterns. Notable examples of such high-dimensional data include genomics data, textual data, image retrieval datasets, and bioinformatics datasets, among others. Mining this data reveals novel, intriguing, and potentially valuable patterns. The ultimate goal of any data mining task is to construct an efficient predictive or descriptive model that not only effectively fits or explains the data but also generalizes its findings to new data [1–4]. However, due to the complex nature and sheer size of the input data, optimizing numerous model parameters becomes a daunting task. Conventional mathematical techniques often fall short in modeling such a vast number of parameters, making the creation of efficient deterministic algorithms an elusive objective.

Evolutionary Algorithms (EAs), known for their inherent parallel architecture, have emerged as a promising approach to address the parameter optimization challenge in modeling extensive and noisy datasets to extract meaningful insights [5, 6]. While EAs were initially employed for solving single-objective problems, many real-world challenges involve multiple, conflicting performance metrics or objectives that require simultaneous optimization. In such cases, optimal performance in one objective may lead to unacceptable trade-offs in others, making multiobjective optimization techniques essential in the data mining domain. For instance, in association rule mining, a well-established field in data mining, a rule's evaluation depends on both its support and confidence values, while the quality of a clustering solution necessitates consideration of various conflicting measures of cluster validity indices. These problems inherently possess a multiobjective nature, aiming to optimize all conflicting objectives simultaneously. Several data

mining and machine learning tasks, such as feature selection, classification, clustering/biclustering, association rule mining, and deviation detection, among others, can significantly benefit from Multiobjective Evolutionary Algorithms (MOEAs) [7–9]. This is especially relevant when optimizing a set of parameters in a machine learning model, as neglecting these parameters may result in degraded performance.

Bioinformatics or computational biology is an interdisciplinary field that combines biology, computer science, mathematics, and statistics to analyze and interpret biological data. Multiobjective optimization techniques find valuable applications in various bioinformatics problems, enabling the discovery of solutions that effectively balance multiple criteria or objectives. Some instances of bioinformatics tasks amenable to multiobjective optimization include feature selection, protein structure prediction, drug design, biological network analysis, phylogenetic tree construction, and more. Several works have leveraged multiobjective optimization algorithms, such as multiobjective evolutionary algorithms (MOEAs) or multiobjective particle swarm optimization (MOPSO), to explore and generate a set of solutions referred to as the Pareto front. This Pareto front represents trade-offs between conflicting objectives, allowing biologists and bioinformaticians to select solutions that best align with their specific research goals and requirements. Although numerous MOEAs are available in the literature to address machine learning, data mining, and bioinformatics tasks, there has been not much comprehensive effort to systematically document and synthesize these methods.

1.1 Concepts of Multiobjective Optimization

In many real-world scenarios, it is common to encounter situations where multiple objectives need simultaneous optimization to address a specific problem. The primary challenge in dealing with multiobjective optimization (MOO) lies in the absence of a universally accepted definition of the optimal solution, making it challenging to compare one solution to another. In these cases, multiple solutions may be deemed acceptable and equivalent, especially when the relative importance of the objectives remains unclear. Ultimately, determining the best solution becomes subjective, relying on the preferences and requirements of the decision-maker. Formally, a multiobjective optimization problem (MOP) can be stated as [5, 10]:
Optimize a set of objective functions:

$$\begin{array}{ll}
 f_1(x) & \text{(Objective 1)} \\
 f_2(x) & \text{(Objective 2)} \\
 \vdots & \\
 f_k(x) & \text{(Objective } k)
 \end{array}$$

Subject to:

$$\begin{aligned}
 g_1(x) &\leq 0 && \text{(Inequality Constraint 1)} \\
 g_2(x) &\leq 0 && \text{(Inequality Constraint 2)} \\
 &\vdots && \\
 g_m(x) &\leq 0 && \text{(Inequality Constraint } m) \\
 h_1(x) &= 0 && \text{(Equality Constraint 1)} \\
 h_2(x) &= 0 && \text{(Equality Constraint 2)} \\
 &\vdots && \\
 h_p(x) &= 0 && \text{(Equality Constraint } p)
 \end{aligned}$$

where:

$$\begin{aligned}
 x & \text{ is the vector of decision variables: } x = [x_1, x_2, \dots, x_n]^T, \\
 f_i : \mathbb{R}^n &\rightarrow \mathbb{R} \text{ represents the } i\text{th objective function for } i = 1, 2, \dots, k, \\
 g_i : \mathbb{R}^n &\rightarrow \mathbb{R} \text{ are inequality constraint functions for } i = 1, 2, \dots, m, \\
 h_i : \mathbb{R}^n &\rightarrow \mathbb{R} \text{ are equality constraint functions for } i = 1, 2, \dots, p.
 \end{aligned}$$

for a minimization problem, the concept of dominance relationship can be described as follows: A vector $\mathbf{v} = (v_1, v_2, \dots, v_k)$ is said to dominate another vector $\mathbf{w} = (w_1, w_2, \dots, w_k)$ (denoted as $\mathbf{v} \preceq \mathbf{w}$) if and only if for every dimension i within the range $\{1, 2, \dots, k\}$, it holds that $v_i \leq w_i$ and there exists at least one dimension j within the same range such that $v_j < w_j$. Formally a solution is said to be in Pareto optimal if it cannot be improved in any one objective without degrading at least one of the other objectives. In other words, it represents a point in the solution space where no other feasible solution offers a better trade-off in all objectives simultaneously. Mathematically it can be written as follows: a solution $\mathbf{u} = (u_1, u_2, \dots, u_n) \in F$ is said to be Pareto optimal with respect to F if and only if there is no other solution $\mathbf{x} = (x_1, x_2, \dots, x_n)$ in F for which the vector $\mathbf{v} = (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_k(\mathbf{x}))$ dominates the vector $\mathbf{w} = (f_1(\mathbf{u}), f_2(\mathbf{u}), \dots, f_k(\mathbf{u}))$. Now for a given Multiobjective Optimization Problem (MOP) represented by $F(\mathbf{x})$, the Pareto optimal set P^* is the set of all Pareto optimal solutions in the solution space. Formally,

$$\begin{aligned}
 P^* &= \{\mathbf{x} \in F \mid \neg \exists \mathbf{u} \in F \text{ such that } (f_1(\mathbf{u}), f_2(\mathbf{u}), \dots, f_k(\mathbf{u})) \\
 &\preceq (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_k(\mathbf{x}))\}.
 \end{aligned}$$

Decision-makers can choose from these Pareto optimal solutions based on their preferences and priorities for different objectives.

Genetic Algorithms (GAs), a special type of Evolutionary Algorithms (EAs), are widely recognized as effective search and optimization strategies, and they draw inspiration from the principles of Darwinian evolution [11]. In a GA, the parameters of the search space are represented as strings, often referred to as chromosomes. A population is initialized with a set of these chromosomes, usually done randomly. To evaluate the quality of each chromosome within the population, a fitness function is employed, which is typically associated with the objective function that the algorithm aims to optimize. This function quantifies how well a chromosome performs concerning the problem's objectives. The key principle of GAs is to mimic the evolutionary process by applying various biologically inspired operators. These operators include selection, crossover (recombination), and mutation. Selection aims to choose the fittest individuals from the current population based on their fitness scores. Crossover, inspired by genetic recombination, involves combining genetic material from two parent chromosomes to produce one or more offspring. Mutation introduces small, random changes in a chromosome to promote genetic diversity. The algorithm iteratively evolves the population from one generation to the next by repeatedly applying these operators. The process continues until a specific termination criterion is met, such as reaching a predefined number of generations or satisfying certain convergence conditions. The best chromosome found in the final generation represents the solution to the optimization problem. The classical GAs are designed to work with a single objective function. Over the years, GAs and EAs have been modified to cope with the challenges of multiobjective optimization problems.

Over several years, Multiobjective Evolutionary Algorithms (MOEAs) have witnessed significant evolution, moving from traditional aggregating methods to more sophisticated, elitist Pareto-based strategies. Among the non-Pareto population-based techniques, the Vector Evaluated Genetic Algorithm (VEGA) [12], for instance, employs a unique selection operator and generates multiple subpopulations by applying proportional selection based on each objective function sequentially. In the realm of Pareto-based approaches, several noteworthy non-elitist MOEAs have emerged, including the Multiple Objective Genetic Algorithm (MOGA) [13],

Niched Pareto Genetic Algorithm (NPGA) [14], and Non-dominated Sorting Genetic Algorithm (NSGA) [15]. These approaches incorporate the concept of Pareto optimality into their selection mechanisms but lack elitism, which means they cannot guarantee the preservation of non-dominated solutions obtained during the search.

On the other hand, elitist MOEAs, such as the Strength Pareto Evolutionary Algorithm (SPEA) [16], SPEA2 [17], Pareto Archived Evolutionary Strategy (PAES) [18], Pareto Envelope-based Selection Algorithm (PESA) [19], PESA-II [20], and Non-dominated Sorting Genetic Algorithm-II (NSGA-II) [21], have garnered attention from researchers. Many recent applications of MOEAs in data mining problems have adopted these Pareto-based elitist approaches as their core optimization strategy.

Additionally, some indicator-based approaches, like the S-metric Selection Evolutionary Multiobjective Optimization Algorithm (SMS-EMOA) [22], which relies on the concept of hypervolume [23], have also been introduced. The primary advantage of indicator-based approaches lies in their scalability when dealing with numerous objectives (four or more). However, it is worth noting that hypervolume-based approaches can be computationally expensive. Besides the evolutionary algorithms, other nature-inspired metaheuristic approaches, such as Particle Swarm Optimization (PSO) [24], Differential Evolution [25], and Simulated Annealing [26] have also been modified to handle optimization of multiple objectives.

1.2 MOO in Data Mining and Machine Learning

The primary challenge in data mining and machine learning problems is determining how to evaluate the performance of a candidate model. This evaluation depends on the specific data mining task, as most problems in this field require optimizing multiple criteria. For example, in the context of a feature selection model, its performance may be measured by its ability to accurately classify a dataset while minimizing the size of the selected feature set. Similarly, rule mining problems aim to optimize various rule interestingness measures, such as support, confidence, comprehensibility, and lift [27], simultaneously. Clustering problems present similar challenges, where the objective is to optimize multiple cluster validity indices concurrently to achieve robust and improved clustering results. This is crucial because no single validity index performs well for all types of datasets [5]. Multiobjective Evolutionary Algorithms (MOEAs) provide a set of non-dominated solutions, representing the best possible trade-offs among the objectives. Users can then choose a single solution from this set based on their preferences. Various strategies exist for selecting a final solution, such as generating a consensus solution that incorporates knowledge from all non-dominated solutions. This approach has been successful in addressing challenges like clustering [28] and classifier ensemble problems [29]. Additionally, for certain problems, all non-dominated solutions are considered final solutions, eliminating the need to choose a single solution from the set. For example, in association rule mining [30] or biclustering [31], all non-dominated solutions, representing rules and biclusters, respectively, constitute the final solution set. Consequently, many data mining problems inherently exhibit multiobjective characteristics, making MOEAs a suitable choice for their application in the field of data mining over the past decade. In this section, we explore the use of multiobjective optimization algorithms in various data mining and machine learning tasks.

1.2.1 Multiobjective Optimization in Clustering

Multiobjective clustering methods, driven by the need to optimize various criteria simultaneously, have been widely applied in clustering techniques for over a decade [9]. These methods are particularly beneficial for addressing the diverse and complex nature of clustering problems. One common approach involves framing clustering as an optimization problem where a cluster validity index is maximized to quantify the quality of clusters [9]. Traditional evolutionary clustering techniques often utilize a single validity measure as the fitness value during optimization [32]. However, as the effectiveness of a single validity measure varies across diverse datasets, it is natural to consider the simultaneous optimization of multiple validity measures. Multiobjective evolutionary algorithms (MOEAs) offer a valuable approach to this challenge. MOEAs in clustering aim to optimize multiple cluster validity indices simultaneously, resulting in high-quality clustering solutions. This process generates a set of near-Pareto-optimal solutions, including non-dominated ones. The choice of the most suitable solution depends on user preferences and specific problem characteristics. The literature offers a variety of multiobjective evolutionary clustering algorithms that differ in their choice of MOEA, chromosome encoding methods, objective functions, evolutionary operators, and mechanisms for selecting the final solution from the non-dominated front.

Various Multiobjective Evolutionary Algorithms (MOEAs) serve as foundational optimization tools for multiobjective clustering [8, 9]. PESA-II [20] is utilized in algorithms such as VIENNA (Voronoi Initialized Evolutionary Nearest-Neighbor Algorithm) [33], MOCK-AM (Multiobjective Clustering with automatic K determination Around Medoids) [34], MOCK (Multiobjective Clustering) [35], and MECEA (Multiobjective Evolutionary Clustering Ensemble Algorithm) [36]. NSGA-II [10] is a key component in various multiobjective clustering approaches, including MOEA(Dynamic) [37], VRJGGA (Variable-length Real Jumping Genes Genetic Algorithms) [38], MOGA (Multiobjective Genetic Algorithm) [39], MOGA(medoid) [40], MOES (Multiobjective Evolutionary Strategy) [41], MOGA-SVM (Multiobjective Genetic Algorithm with Support Vector Machine) [28, 42], EMCOC (Evolutionary Multiobjective Clustering for Overlapping Clusters detection) [43], MOGA(mode) [44], DYN-MOGA (Dynamic MOGA) [45], MOVGA (Multiobjective Variable-length Genetic Algorithm) [46], and MOCA (Multiobjective Clustering Algorithms) [47]. SPEA2 [17] is employed as the underlying optimization tool in [48] and [49]. NPGA [14] is applied in MOKGA (Multiobjective K-Means Genetic Algorithm) [50].

In multiobjective clustering, chromosome representation can be categorized into prototype-based and point-based approaches. Prototype-based encoding employs real numbers to represent cluster centers such as centroids, medoids, and modes and has been utilized in various multiobjective clustering algorithms, including MOGA, SiMM-TS, MOGA-SVM, and MOVGA [5]. It offers benefits like shorter chromosome lengths and suitability for handling overlapping clusters. However, it tends to capture round-shaped clusters and may be less effective in high-dimensional

datasets. In contrast, point-based encoding, which encompasses complete clustering solutions, can be further divided into cluster label-based and locus-based adjacency representation. Cluster label-based encoding assigns cluster labels to each position in the chromosome, as seen in algorithms like VIENNA, MOKGA, and GraSC. A variant of this approach is used in MOCK, where chromosomes consist of genes representing links between data points, forming a graph [34, 51]. Several other algorithms, including MECEA [36], AI-NSGA-II [52], and DYN-MOGA [53], adopt this strategy. While point-based encoding is not biased toward convex-shaped clusters, it may require more time to converge, particularly with a large number of data points. Its advantage lies in having a chromosome length independent of the number of clusters encoded, distinguishing it from prototype-based encoding.

Multiobjective clustering commonly employs cluster validity indices as objective functions, with various algorithms optimizing different pairs of such indices depending on the specific clustering problem and dataset characteristics. For example, combinations like overall cluster deviation and cluster connectedness are used in algorithms such as MOCK, while others prefer pairs like J_m [54] and XB [55] to encourage compact and well-separated clusters. Objective functions often rely on cluster validity indices such as DB [56], $Dunn$ [57], XB [55], I [58], and J_m [54]. Additionally, measures like average cluster variance, average between-group sum of squares (ABGSS), cluster connectedness, overall cluster deviation, cluster separation, cluster dominance, and the diameter of the largest cluster are utilized, either individually or in combination, to formulate effective objective functions. In some cases, more than two objective functions are involved, but managing multiple objectives can be challenging for Multiobjective Evolutionary Algorithms (MOEAs). An interactive multiobjective clustering approach addresses this challenge by involving a human decision-maker to determine the most suitable set of objective functions during clustering solution evolution [59]. The selection of objectives remains crucial in multiobjective clustering, significantly impacting clustering quality.

In the realm of multiobjective clustering, the selection of chromosome representation plays a pivotal role in determining the appropriate evolutionary operators, including crossover and mutation. For instance, prototype-based representations, frequently employed in algorithms like MOGA [39], MOGA-SVM [28, 42], and MOVGA [46], tend to favor single-point or two-point crossovers. Some approaches, such as those developed by Ripon et al. [38, 43], opt for the utilization of jumping gene crossover. In contrast, when adopting a centroid-pool-based approach, as exemplified in the work by Won et al. [60], centroids encoded in parent chromosomes are combined, and offspring chromosomes are selected from this centroid pool. On the other hand, point-based encoding strategies, commonly found in algorithms like MOCK and VIENNA, typically employ uniform crossovers. Following crossovers, mutation operators come into play to ensure population diversity. Prototype-based encodings often rely on centroid perturbation, while medoid-based and mode-based encodings utilize random medoid replacement and mode perturbation, respectively. In the case of cluster label-based encoding, a common mutation approach involves replacing the class label of a selected data point

with a random class label. To address the challenges posed by long chromosome lengths, some algorithms introduce specialized mutation operators, such as directed neighborhood-biased mutation, which adaptively alters class labels and has found wide adoption in various algorithms. It is worth noting that these choices related to chromosome representation and mutation strategies significantly influence the efficiency and effectiveness of multiobjective clustering algorithms.

There are several approaches for selecting the ultimate solution from the set of non-dominated solutions generated by the MOEA. These methods can be classified into three categories: the independent objective-based approach, the knee-based approach, and the cluster ensemble-based approach.

In the independent objective-based approach, a distinct cluster validity index, not utilized during the clustering process, is employed to make the final selection from the non-dominated front [39]. This approach is appreciated for its simplicity, but it's worth noting that the choice of the validity index may influence the ultimate result. Since this index is not directly optimized, there may be some doubts about the validity of this approach.

The knee-based approach centers on the selection of the knee solution from the non-dominated front. A knee solution is characterized by a significant change in one objective value when the others change. While employed in algorithms like MOCK [51], this approach lacks a clear rationale for choosing the knee solution as the final one and can be time-intensive.

The cluster ensemble approach aims to consolidate valuable information from non-dominated solutions in multiobjective clustering. Techniques like CSPA, HGPA, and MCLA are commonly used for this purpose, while a novel approach introduced by Mukhopadhyay et al. [28, 42, 61] identifies data points consistently belonging to the same class in most non-dominated solutions. These reliable points are used to train a classifier like SVM or k-NN, which is then employed to assign class labels to remaining data points. This ensemble-based technique has demonstrated superior performance in applications like satellite image segmentation and microarray data clustering, outperforming independent objective-based methods. Each approach has its advantages and limitations, making the selection of the final clustering solution a critical consideration in multiobjective clustering algorithms.

1.2.2 Multiobjective Optimization in Classification

Multiobjective Evolutionary Algorithms (MOEAs) have found extensive applications in classification tasks, with three distinct approaches being commonly explored [7].

The first approach focuses on utilizing MOEAs to evolve a set of effective classification rules [62–64]. A classification rule is typically expressed as an “if-then” statement, where the “if” part comprises attribute-value pairs combined with logical operators, defining conditions, and the “then” part designates the class. For instance,

a rule could be: “if age >18 and income <\$30000 then class= Low income.” These attribute-value pairs must be categorical, necessitating the discretization of continuous attributes. Rule-based classification systems aim to identify a suitable set of rules that effectively represent the training data, optimizing for classification performance. MOEA-based classification approaches commonly employ NSGA-II as the underlying optimization algorithm. Chromosome representation can follow either the Pittsburgh approach (encoding a set of rules in one chromosome) or the Michigan approach (each chromosome encodes one rule) [65]. Various studies have explored different sets of objective functions with the common goal of striking a balance between the accuracy and complexity of the candidate rule set. The final solution is selected using metrics such as classification accuracy, area under the curve (AUC), or a combination of metrics. However, choosing the optimal solution remains a challenge, often involving a trade-off between different evaluation criteria.

The second approach involves employing MOEAs to define class boundaries or hyperplanes within the training data, enhancing the separation of different classes. A promising approach in utilizing MOEAs for classification involves evolving class boundaries capable of effectively distinguishing between different classes, particularly when these boundaries are nonlinear. These nonlinear surfaces can be approximated using hyperplanes, converting the classification problem into a multiobjective optimization challenge. The three objectives in this approach are to minimize misclassified patterns and number of hyperplanes while maximizing classification accuracy. This prevents overfitting and ensures that smaller classes are not disregarded during training. Binary chromosomes of variable lengths encode the parameters of varying hyperplanes. The Constrained Elitist MOEA (CEMOGA) serves as the underlying optimization tool. The final solution selection relies on an aggregation function that combines the objective functions. Although this approach showed promise, further developments beyond the initial attempt reported in [66] are lacking.

The third approach leverages MOEAs for the training process and constructing well-known classifiers like neural networks and decision tree classifiers. It has been leveraged for the training and modeling of standard classifiers like Artificial Neural Networks (ANNs) [6], Support Vector Machines (SVMs) [67], and decision trees [6]. Several MOEAs have been used as the underlying optimization techniques in these approaches. Notably, NSGA-II, a widely adopted MOEA, is found to be the most commonly used approach for model building and training for SVMs [68, 69]. However, the others such as SPEA2 and specialized algorithms like Single-Front Genetic Algorithm (SFGA) have found application in this specific scenarios.

Encoding parameters is a crucial aspect, and binary encoding is a common choice. For example, in Support Vector Machines (SVMs), chromosomes can represent feature subsets and SVM kernel parameters. In contrast, real-number vectors are used to represent SVM parameters in approaches like evoSVM. When dealing with Artificial Neural Networks (ANNs), real numbers can be used to describe network topology and weights, while mixed encodings may include binary, integer, and real-value components, especially in situations like dynamic Recurrent Neural Networks (RNNs), where the encoding must capture complex structures.

Objective functions are designed to balance classification performance and model complexity, usually encompassing various classification performance metrics. In SVMs, the objectives aim to minimize false-positive and false-negative rates while reducing the number of support vectors, effectively managing model complexity. In the case of ANNs, objectives focus on minimizing false positives and false negatives to address imbalanced class problems, highlighting the adaptability of Multiobjective Evolutionary Algorithms (MOEAs). Other objectives optimize classification accuracy while minimizing the size of decision trees, ensuring a trade-off between model accuracy and complexity.

Evolutionary operators, including crossover and mutation, vary depending on the specific problem and encoding method. Binary encodings often make use of standard operators provided by the selected MOEA, such as NSGA-II. Real-number encodings, on the other hand, may use techniques like multipoint crossover and random weight modifications. Specialized mutation approaches, such as hybrid mutations and nonuniform mutations, are also applied to adapt to the encoding scheme and meet the specific requirements of the problem at hand.

Choosing the final solution from the set of non-dominated options is an important decision. Different approaches handle this step in various ways. Some consider the non-dominated classifiers as an ensemble system, consolidating their predictions. Alternatively, a more precise selection criterion is employed, often based on performance metrics like accuracy. For instance, when it comes to fine-tuning SVM parameters, a separate validation dataset is utilized to assess the performance of each non-dominated solution. The solution that achieves the highest accuracy on this validation set is then designated as the final classifier. These MOEA-based methods provide adaptability and allow for customization in classifier design and training, making them suitable for diverse problem domains and specific requirements.

1.2.3 Multiobjective Optimization in Feature Selection

The feature selection problem is commonly framed as an optimization task where the goal is to identify an optimal subset of features using a specific evaluation criterion [6, 7]. Genetic and evolutionary algorithms have gained popularity for addressing this challenge [70]. Typically, these approaches follow a wrapper approach, where feature subsets are encoded in chromosomes, and a feature evaluation criterion serves as the fitness function. The performance of the selected features in classifying (in supervised cases) or clustering (in unsupervised cases) the dataset is used to evaluate their effectiveness. However, the use of a single evaluation criterion may not be universally effective for all datasets, which has led to the emergence of multiobjective feature selection, enhancing the robustness of the process.

A variety of Multiobjective Evolutionary Algorithms (MOEAs) are employed as underlying optimization tools for different feature selection algorithms [71–76]. Binary encoding is a common choice for representing feature subsets, where each