On The Fairness in Advanced Spectral Clustering Methods

Thesis submitted to the
Indian Institute of Technology Guwahati
for the award of the degree

of

Doctor of Philosophy

in

Computer Science and Engineering

Submitted by

Laxita Agrawal

Under the guidance of

Prof. V. Vijaya Saradhi
and
Dr. Teena Sharma

Department of Computer Science and Engineering

Indian Institute of Technology Guwahati

June 13, 2026

Department of Computer Science and Engineering
Indian Institute of Technology Guwahati
Guwahati - 781039 Assam India

Prof. V. Vijaya Saradhi
and
Dr. Teena Sharma
Department of Computer Science and Engineering
IIT Guwahati
Email : saradhi@iitg.ac.in and teena@iitg.ac.in

Certificate

This is to certify that this thesis entitled, **"On The Fairness in Advanced Spectral Clustering Methods"**, being submitted by **Laxita Agrawal**, to the Department of Computer Science and Engineering, Indian Institute of Technology Guwahati, for partial fulfillment of the award of the degree of Doctor of Philosophy, is a bonafide work carried out by her under our supervision and guidance. The thesis, in our opinion, is worthy of consideration for award of the degree of Doctor of Philosophy in accordance with the regulation of the institute. To the best of our knowledge, it has not been submitted elsewhere for the award of the degree.

.....
Prof. V. Vijaya Saradhi
Professor

.....
Dr. Teena Sharma
Assistant Professor

Date: June 13, 2026
Place: Guwahati

Declaration

I certify that:

- The work contained in this thesis is original and has been done by myself and under the general supervision of my supervisor.
- The work reported herein has not been submitted to any other Institute for any degree or diploma.
- Whenever I have used materials (concepts, ideas, text, expressions, data, graphs, diagrams, theoretical analysis, results, etc.) from other sources, I have given due credit by citing them in the text of the thesis and giving their details in the references. Elaborate sentences used verbatim from published work have been clearly identified and quoted.
- I also affirm that no part of this thesis can be considered plagiarism to the best of my knowledge and understanding and take complete responsibility if any complaint arises.
- I am fully aware that my thesis supervisor is not in a position to check for any possible instance of plagiarism within this submitted work.

Date: June 13, 2026
Place: Guwahati

(Laxita Agrawal)

Declaration on the Use of AI

I hereby declare that this PhD thesis titled "On The Fairness In Advanced Spectral Clustering Methods" is my own original work carried out under the supervision of Prof. V. Vijaya Saradhi and Dr. Teena Sharma.

During the preparation of this thesis, I used AI-based tools (such as ChatGPT and Google Gemini) solely for non-creative tasks, including:

- improving grammar, clarity, and organization of text, and
- verifying the correctness of references.

No AI system was used to conduct data analysis, generate scientific insights, design methodologies, or produce research results. All conceptual development, algorithm design, analysis, and conclusions presented in this thesis are entirely my own.

I confirm that no text, code, images, or results have been directly copied from AI outputs without thorough review and modification. I take full responsibility for the accuracy, originality, and integrity of all content in this thesis.

Name: Laxita Agrawal
Roll Number: 206101005

Dedicated to

My Family & Friends

For their unconditional love, blessings and constant inspiration

Acknowledgements

My Ph.D. journey has been a meaningful and inspiring adventure, shaped by the patience, encouragement, and wisdom of many people. I extend my heartfelt gratitude to everyone who supported me throughout the preparation of this doctoral thesis.

First and foremost, I would like to express my heartfelt gratitude to my supervisors, Prof. Vijaya Saradhi and Dr. Teena Sharma, for their unwavering guidance, encouragement, and patience throughout this journey. I am especially grateful to Prof. Vijaya Saradhi for helping me stay focused and disciplined in my research. His clarity of thought and emphasis on rigorous presentation have significantly shaped my research. I am equally indebted to Dr. Teena Sharma for her consistent support and the time she generously devoted to guiding me at every stage. Her understanding and flexibility helped me handle challenges with confidence. I truly appreciate the freedom they gave me to explore ideas while always being available to offer direction and motivation whenever needed.

I would also like to express my sincere gratitude to the members of my Doctoral Committee - Prof. Ashish Anand, Prof. Sanasam Ranbir Singh, and Dr. Suresh Sundaram for their insightful comments and suggestions, which greatly contributed to improving the quality of my work. I am further thankful to the anonymous reviewers of my research for their thoughtful and critical feedback, which helped me enhance the quality of my work.

I would like to express my sincere gratitude to Prof. Tamarapalli Venkatesh, Head of the Department of Computer Science and Engineering at IIT Guwahati, for providing access to the department's facilities and resources. I am thankful to all the faculty members, staff, and security personnel for their constant assistance and support. I also extend my appreciation to the staff of the Academic Affairs office for their support in processing my applications and grant requests. I gratefully acknowledge the MHRD, Government of India, for the financial support provided throughout my Ph.D. years, without which this research would not have been possible.

I extend my appreciation to my fellow research scholars and friends - Adithya, Alik, Sonal, Soumya Bharadwaj, Soumya Mishra, Prachuryya, Swati, Nilotpola,

Sreyas, Rajeswari, Akshay, and Suklav for making my Ph.D. journey both memorable and enjoyable. Their encouragement, support, and willingness to help me navigate challenges, whether academic or personal, have been invaluable. Our many discussions and exchanges of ideas have greatly enriched my research, and their companionship has been an essential part of this experience. I would also like to acknowledge Vivekananda and Shifali for their unwavering support throughout my time at IIT Guwahati. My journey here has been truly memorable, thanks to all of you.

Of course, my Ph.D. colleagues are not the only people I wish to thank. I would also like to acknowledge Nalin for motivating me to pursue this Ph.D. and for valuable support. His encouragement and belief in my potential played a meaningful role in helping me move forward with confidence. I am also grateful to my close ones - Bhasha, Tanvi, Ruchi, Priyanshi, Swapnil, and Tajendar for their constant support and the warmth they brought into my life.

I am profoundly grateful to God and to my family, including my sister Yamini, parents, and grandparents, for their unconditional love, support, care, warmth, and profound encouragement throughout the years. My heartfelt gratitude goes especially to my parents, whose trust and confidence in me made this journey meaningful, secure, and fulfilling. I fall short of words to express how much their support has meant to me.

Abstract

The increasing reliance on machine learning and data-driven decision systems has heightened concerns about fairness, particularly in algorithmic decisions of high-impact domains such as healthcare, finance, and social analytics, where sensitive attributes (such as gender, age, race, etc) influence outcomes. Ensuring equitable treatment across demographic groups has become an essential requirement. While extensive research has addressed fairness in supervised learning, achieving fairness in unsupervised tasks, such as clustering, can be challenging when fairness considerations are not taken into account. This may inadvertently produce imbalanced or biased partitions that disproportionately advantage or disadvantage certain demographic groups. Fair clustering methods aim to mitigate these issues by ensuring that clusters maintain a balanced representation of demographic groups, with recent work introducing balance as a fairness notion to promote proportional representation within clusters.

Among clustering approaches, Spectral Clustering (SC) is a widely used unsupervised method due to its ability to capture complex, non-linear structures in data through graph Laplacians and eigenvector embeddings. However, standard SC operates without considering sensitive attribute information, often resulting in unfair clusters. Existing fair SC methods enforce fairness, but at the expense of clustering quality. In contrast, advanced SC techniques, such as constrained SC and one-step spectral rotation clustering, achieve strong structural performance but overlook fairness completely. This disconnect motivates the need for integrating fairness in advanced SC methods that can simultaneously preserve clustering quality and enforce equitable group representation.

To address this gap, this dissertation presents fairness-aware extensions of advanced SC methods. The research introduces four significant contributions. The first contribution integrates fairness into constrained SC by modifying the Laplacian formulation to allow sensitive attribute information and pairwise constraints to jointly influence the spectral embedding. The second contribution embeds fairness directly into the pairwise constraint representation that encodes demographic information within the constraint structure itself. These in-processing methods enhance fairness without degrading clustering quality. However, these two contributions integrate fairness into the optimization criteria, where the similarity graph is constructed prior to the clustering. However, they cannot correct biases already encoded in the similarity graph. This observation motivates the need for a preprocessing approach that enforces fairness at the graph-construction level itself. The third contribution introduces a graph-level preprocessing framework that constructs a fair and constraint-respecting k-Nearest Neighbor (kNN) graph, ensuring neighborhood fairness and constraint adherence prior to clustering. By embedding these properties directly into

the similarity graph, the framework promotes fairer and more constraint-compliant cluster formations without modifying the clustering algorithm itself.

While these three contributions significantly improve clustering quality by incorporating pairwise constraints into different stages of the SC process, they still rely on the traditional two-step framework, where graph construction and clustering are performed separately. This separation introduces information loss, sub-optimal results, and limits the influence of fairness on the final cluster assignments. In contrast, one-step SC methods unify graph learning and cluster assignment into a single optimization process, thereby achieving better clustering performance. Motivated by these advantages, the fourth contribution integrates fairness into the one-step SC paradigm by introducing a novel group fairness term, resulting in a fair one-step spectral rotation clustering model that jointly performs graph learning, fairness enforcement, and clustering within a single optimization framework. The unified design ensures that fairness fully guides the clustering process, resulting in high-quality, demographically balanced clusters.

All proposed methods are theoretically analyzed and validated through extensive experiments on real-world datasets. The results demonstrate that the methods generally improve fairness in clustering while preserving competitive clustering quality. They outperform existing unfair baselines and standard SC across both fairness and performance metrics. Collectively, these findings demonstrate that fairness can be effectively integrated into advanced SC frameworks without compromising their effectiveness.

The thesis concludes by summarizing these contributions and proposing some relevant future research directions.

Contents

List of Figures

List of Tables

List of Algorithms

List of Symbols

G_t Demographic group t

J_{CM} Penalization term for ML and CL constraints

J_{SC} SC objective criterion

α Disparate impact parameter

δ Balance parameter controlling deviation from demographic proportions

γ Regularization parameter

λ Regularization factor controlling the the constraints and SC objective

$\mathcal{N}_k^{FC}(\mathbf{x}_i)$ Fair-constrained neighborhood of $\mathbf{x}_i$

$\mathcal{N}_k(\mathbf{x}_i)$ kNN set of $\mathbf{x}_i$

$\mathrm{Tr}(\mathbf{X})$ Trace of matrix $\mathbf{X}$

π_ℓ The ℓ-th cluster of the partition

σ, ζ_i Lagrangian multipliers

$\mathbf{D}$ Degree matrix

$\mathbf{F}$ Group membership matrix

$\mathbf{H}$ Cluster indicator matrix

$\mathbf{I}$ Identity matrix

$\mathbf{L}$ Graph Laplacian matrix

$\mathbf{L}_{FC}$ Fair-constrained graph Laplacian

$\mathbf{Q}$ Constraint matrix

$\mathbf{Q}_{CL}$ Matrix encoded with CL constraints

$\mathbf{Q}_{ML}$ Matrix encoded with the ML constraints

$\mathbf{Q}_{fair}$ Fair constraint matrix

$\mathbf{R}$ Rotation matrix

$\mathbf{S}$ Similarity matrix

$\mathbf{S}_{FC}$ Fair-constrained similarity matrix

$\mathbf{W}$ Projection matrix

$\mathbf{X}^{-1}$ Inverse of matrix $\mathbf{X}$

$\mathbf{X}^{T}$ Transpose of matrix $\mathbf{X}$

$\mathbf{f}$ Group membership vector

$\|X\|_{2,1}$ $\ell_{2,1}$ norm of matrix X, defined as $\|X\|_{2,1} = \sum_j \sqrt{\sum_i x_{ij}^2}$

$\|X\|_F$ Frobenius norm of matrix X, defined as $\|X\|_F = \sqrt{\sum_i \sum_j x_{ij}^2}$

c Number of clusters

$g(\mathbf{x}_i)$ Sensitive attribute value of data point $\mathbf{x}_i$

k Number of nearest neighbors

m Number of demographic groups

n Number of samples

List of Acronyms

kNN k-Nearest Neighbor

Acc Accuracy

ADMM Alternating Direction Method of Multipliers

AI Artificial Intelligence

ALM Augmented Lagrange Method

C-kNN Constrained k-Nearest Neighbor

CL Cannot-Link

cSC constrained Spectral Clustering

fair-cSC fair-constrained Spectral Clustering

FC-kNN Fair Constrained k-Nearest Neighbor

FCSC Flexible Constrained Spectral Clustering

FOSSRC Fair One-Step Spectral Rotation clustering

FPCSC Fairness in Pairwise Constrained Spectral Clustering

FSC Fair Spectral Clustering

GFT Group Fairness Term

ILPD Indian Liver Patient Dataset

IRLS Iteratively Re-weighted Least Square

LPP Locality Preserving Projection

ML Must-Link

NCut Normalized Cut

NMI Normalized Mutual Information

OSSRC One Step Spectral Rotation Clustering

PCSC Pairwise Constrained Spectral Clustering

RCut Ratio Cut

RI Rand Index

SC Spectral Clustering

SFSC Scale Fairness Spectral Clustering

SL Spectral Learning

UFSC Unified Fair Spectral Clustering

"A person who never made a mistake never tried anything new."

~Albert Einstein

1 Introduction

1.1 Research Context

In recent years, a growing number of critical decisions have been controlled by Artificial Intelligence (AI) and machine learning algorithms. With their increasing implementation across domains such as business, healthcare, education, and government applications, these automated systems now play a decisive role in shaping people's lives. However, as these automated decision-making systems become more prevalent, concerns have emerged regarding the fairness and ethical reliability of such systems. Since machine learning models often rely on historical data, they may inadvertently inherit and amplify existing societal biases. This can lead to discriminatory outcomes against certain demographic groups defined by

sensitive attributes such as gender, race, or age. As a result, individuals may experience unfair treatment in high-stakes applications such as job recruitment, loan approvals, and medical diagnoses, where algorithmic decisions can profoundly impact their lives. Several well-documented cases have exposed these issues. For example:

- AI systems can unfairly influence judicial outcomes.

  Example: In the U.S. criminal justice system, an algorithm used to predict recidivism was found to falsely predict future criminality among African-American defendants at nearly twice the rate of white defendants [1, 2].

- AI systems can unfairly allocate employment opportunities.

  Example: Amazon's machine learning–based hiring tool was discovered to discriminate against female candidates, particularly for technical and software development positions. The bias stemmed from historical training data that predominantly represented male applicants [3].

- AI systems can reinforce gender bias in targeted advertising.

  Example: Google's ad-targeting algorithm disproportionately displayed high-paying executive job advertisements more frequently to men than to women [4, 5].

Such evidence has prompted extensive research into fairness-aware machine learning, which seeks to ensure equitable treatment and representation of all demographic groups in algorithmic decision-making processes [6, 7]. The growing awareness of algorithmic bias has led to the development of diverse frameworks for defining, quantifying, and improving fairness in machine learning models [8–10]. A comprehensive overview of this field is presented by Barocas *et al.* [11], who formalize the trade-offs between various fairness definitions. These frameworks mainly employ strategies such as data preprocessing [7,12], in-processing [13,14], or post-processing [15,16] to enhance fairness in machine learning systems. Fairness in these systems is generally interpreted through two perspectives: group fairness and individual fairness. *Group fairness* ensures that different demographic groups receive equitable treatment or outcomes from an algorithm, whereas *individual fairness* requires that similar individu-

als be treated similarly regardless of their group membership [17]. Various mathematical definitions have been proposed to operationalize these notions, serving as the foundation for fairness evaluation in machine learning systems [18].

Extensive research has been conducted on fairness in supervised learning, where various notions of fairness have been defined [18–20]. However, these notions are not readily applicable to unsupervised learning problems such as clustering. Unlike supervised learning, unsupervised learning lacks an explicit ground truth or predefined target variable, making it challenging to define and measure fairness. Traditional clustering algorithms, including k-means, hierarchical clustering, and Spectral Clustering (SC), are based on the fundamental principle of grouping similar data instances together, as extensively discussed in the pioneering work of Jain *et al.* [21]. Consequently, when sensitive information is implicitly correlated with feature patterns, the resulting clusters may exhibit demographic bias, over-representing certain groups while under-representing others. In recent years, the problem of fairness in clustering has received growing attention, with several studies investigating how clustering outcomes can be made fair [22, 23]. A pioneering contribution in this direction was made by Chierichetti *et al.* [24], who introduced the notion of balance for fair clustering, where each data point is assigned a label, such as red or blue, representing a sensitive attribute. The objective is to partition the dataset into clusters while preserving balance: the proportion of red and blue points within each cluster should reflect their overall proportion in the entire dataset.

The notion of balance, as provided by Chierichetti *et al.* [24], is as follows:

Balance:

A fairness criterion that quantifies the representation ratio of sensitive groups within each cluster, promoting demographic equity.

Maintaining this balance is crucial for mitigating disparate impact and ensuring that all groups are fairly represented in the clustering outcomes. Building upon the foundational idea of balance, subsequent research extended this concept to SC [25, 26], where clusters are derived from the spectral properties of a similarity graph rather than direct data partitions.

Spectral Clustering

SC is a powerful and widely adopted graph-based clustering algorithm designed to handle complex and high-dimensional data [27–29]. It operates by constructing a similarity graph among data points and performing a spectral decomposition of the corresponding graph Laplacian to identify clusters. Conceptually, SC seeks to find a minimum cut in the graph that partitions the data into groups with strong intra-cluster similarity and weak inter-cluster similarity.

Despite its effectiveness, SC can lead to outcomes that inadvertently reflect or amplify demographic biases present in the data. As a result, the obtained clusters may exhibit representational disparities across sensitive groups. This limitation has motivated a growing body of research aimed at integrating fairness into the SC framework. One of the earliest efforts in this direction was made by Kleindessner *et al.* [25], who introduced fairness in SC by imposing balance constraint on the relaxed optimization formulation of the clustering problem. Their approach modifies the standard SC framework by incorporating a linear constraint that ensures proportional representation of sensitive groups in the embedding space. However, the inclusion of these fairness constraints results in a trade-off between fairness and clustering quality, as measured by traditional objective functions such as RatioCut and Normalized Cut. Further, Wang *et al.* [26] proposed a scaled version to address the high computational cost of [25]. Subsequently, numerous studies have further explored various aspects of fair SC [30–33], focusing on both theoretical guarantees and practical adaptations. These contributions highlight the evolution of fair SC as an important subfield within fairness-aware unsupervised learning. While these methods primarily aim to improve fairness in the obtained clusters, they often do so at the expense of clustering quality. This inherent trade-off between fairness and clustering performance remains a persistent challenge, as enhancing one aspect can lead to deterioration in the other [34].

To overcome the limitations of traditional SC in terms of performance, various advanced methods have been proposed. Among these, constrained SC and one-step SC have emerged as two prominent approaches that enhance clustering quality by incorporating additional

information or optimizing the algorithmic process to achieve more meaningful partitions. The following subsections provide an overview of these methods.

Constrained Spectral Clustering

Constrained SC extends the conventional SC framework by integrating prior knowledge in the form of pairwise constraints. These constraints are typically expressed as Must-Link (ML) and Cannot-Link (CL) constraints, which respectively indicate that certain pairs of data points should or should not belong to the same cluster. By embedding these constraints into the SC formulation, the algorithm can leverage side information to guide the clustering process more effectively. There are two main ways of integrating pairwise constraints into SC. The first involves manipulating the affinity matrix to reflect the imposed relationships between data points [35–37]. The second approach integrate the pairwise constraints directly in the optimization criterion, thereby bounding the feasible space [38–40]. These approaches enable the clustering process to respect prior knowledge. A comprehensive review of pairwise constraints in graph-based clustering can be found in [41].

In this thesis, the term constraints is used in two distinct contexts. First, the optimization constraints with spectral clustering objective function and define the feasible solution space of the optimization problem. Second, the pairwise constraints refer specifically to ML and CL constraints, which encode user provided prior knowledge about the relationships between data points. These pairwise constraints are external to the objective function and are incorporated to guide the clustering process.

One-Step Spectral Rotation Clustering

One Step Spectral Rotation Clustering (OSSRC) represents another advancement aimed at improving the efficiency and robustness of the traditional framework. Standard SC typically involves two main sub-tasks:

1. Constructing a similarity graph using the input data, which often contains noise and redundancy.

2. Applying the k-means algorithm to spectral embedding, which is sensitive to the initial cluster centers and may result in significant deviations from actual results.

Each of these steps introduces potential sources of information loss and sub-optimality, ultimately affecting the quality and stability of the final clustering outcome.

To address these limitations, OSSRC was developed to unify the embedding and clustering processes into a single optimization framework. By integrating graph construction, spectral embedding, and partitioning into a single joint objective, OSSRC enhances computational efficiency, stability, and clustering accuracy. Recent studies have explored various formulations of OSSRC to improve further its scalability and robustness [42–46]. These advancements demonstrate the potential of OSSRC as a reliable and efficient alternative to traditional SC.

Although these extensions significantly enhance clustering performance of SC, they do not explicitly account for fairness considerations. Clusters generated by these methods can still reflect biases inherent in the data, leading to unequal representation across sensitive groups. Consequently, a gap remains in developing methods that can jointly ensure fairness and maintain clustering quality. This gap forms the basis for the motivation of this dissertation.

1.2 Motivation of the Research Work

Despite substantial progress in the development of fairness-aware SC algorithms, achieving an optimal balance between fairness and clustering quality remains a challenge. Most existing fairness-based approaches aim to improve fairness by modifying the spectral embedding; however, these modifications often distort the underlying data structure, resulting in reduced cluster coherence or degraded quality. On the other hand, advanced versions of SC, such as constrained SC and one-step SC, focus solely on enhancing clustering quality and have demonstrated remarkable improvements in quality, efficiency, and robustness; however, they do not explicitly consider fairness. This divergence between fairness-oriented

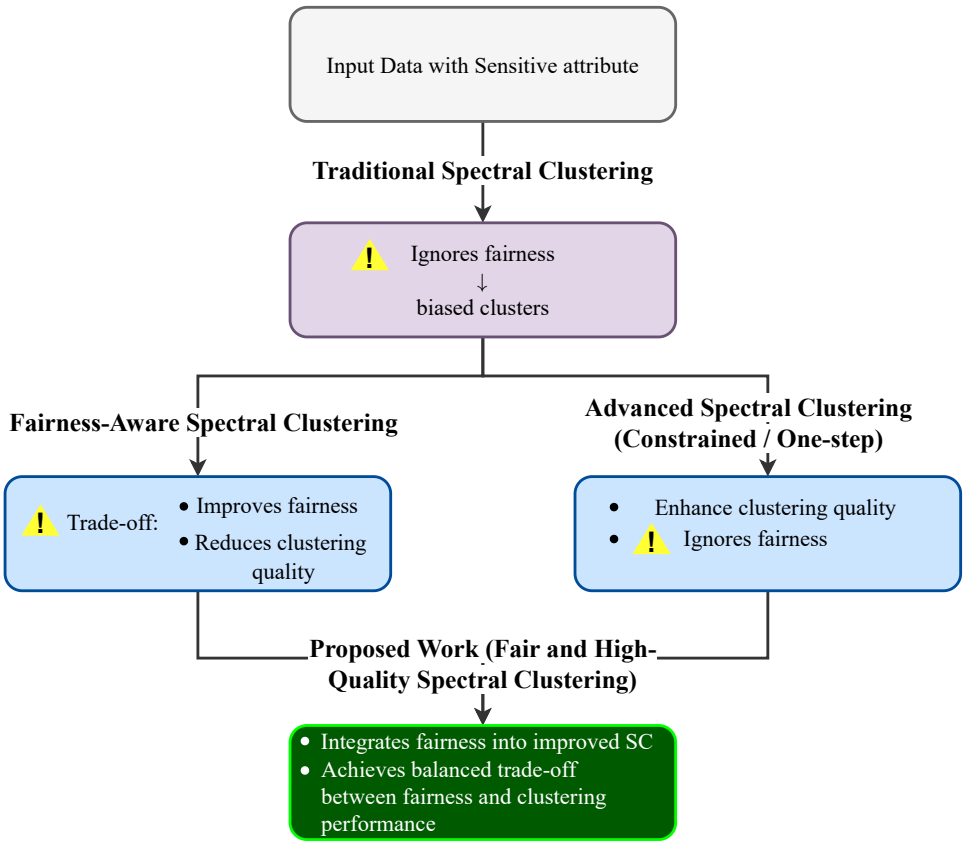


Figure 1.1: *Conceptual overview of the motivation behind the proposed work.*

and performance-aware research highlights a critical gap in current literature.

Addressing this gap requires approaches that can jointly preserve the clustering quality while ensuring fairness across sensitive groups. In real-world scenarios, where data often contains sensitive attributes such as gender, age, or ethnicity, maintaining equitable representation of groups is essential for unbiased data-driven decision-making.

Motivated by these observations, this dissertation explores ways to integrate fairness into advanced SC frameworks. Specifically, it focuses on incorporating fairness into constrained and one-step SC paradigms to achieve clusters that retain their quality while ensuring fair representation. By embedding fairness into these enhanced methods, the work seeks to establish a balanced trade-off between clustering performance and fairness.

To further illustrate the progression of research and motivation, Figure 1.1 presents a conceptual overview of how SC has evolved from traditional approaches to fairness-aware and performance-aware variants, highlighting their respective advantages and limitations.

The figure also emphasizes the need for a framework that integrates fairness considerations into improved SC methods, which forms the core focus of this dissertation.

Thesis Goal:

Develop clustering methods that effectively integrate fairness considerations into advanced SC frameworks, such as constrained and one-step SC, to ensure equitable representation of sensitive groups while enhancing clustering quality.

1.3 Thesis Objectives

Motivated by the above observation, the main objectives of this dissertation are as follows:

- *Fairness-integrated Constrained Spectral Clustering via Laplacian Adjustment:* To formulate a fairness-integrated constrained SC framework that incorporates fairness in the constraint set, followed by Laplacian adjustment. The objective is to ensure equitable cluster representation while minimizing any adverse impact on the quality of clustering.

- *Constrained Spectral Clustering with Pairwise Fairness:* To develop a constrained SC method that incorporates pairwise fairness within the constraint set, thereby promoting balanced relationships among samples from different sensitive groups and improving overall clustering quality.

- *Fair and Constraints-respecting Graph Construction for Spectral Clustering:* To propose a pre-processing framework that constructs a fair and constraint-respecting similarity graph. This approach embeds fairness during the graph formation stage by maintaining balanced neighborhood structures while respecting the provided pairwise constraints.

- *Fairness-aware One-Step Spectral Rotation Clustering:* To introduce a group fairness term, defined using an imbalance measure, within the one-step SC objective. This

integration aims to achieve balanced cluster assignments while preserving both the computational efficiency and clustering quality.

1.4 Thesis Contributions

To achieve the aforementioned objectives and address the inherent trade-off between fairness and clustering quality, this dissertation makes the following major contributions, each focusing on incorporating fairness at different stages of the advanced SC methods. The subsequent subsections provide a brief overview of the proposed methods corresponding to each contribution.

1.4.1 Fairness-aware Constrained Spectral Clustering (In processing approaches)

The first contributory chapter of this dissertation focuses on incorporating fairness directly into the clustering objective. To this end, two fairness-aware constrained SC methods are developed, each designed to integrate fairness considerations into the optimization process by jointly handling pairwise constraints and fairness requirements. Both contributions are discussed below.

Contribution 1: Fairness-integrated Constrained Spectral Clustering via Laplacian Adjustment

The first contribution proposes a novel method named fair-constrained Spectral Clustering (fair-cSC). The proposed method integrates fairness into the ML and CL constraints by defining a fair constraint matrix, ensuring that pairwise relationships do not introduce bias against any particular group. Additionally, a balance constraint is incorporated to enforce fairness across input data points, promoting equal representation of sensitive groups within clusters. Comprehensive experiments on six benchmarked datasets, including ablation studies, demonstrate that the proposed fair-cSC method effectively enhances fairness

while preserving clustering quality. Furthermore, the ablation study provides insights into the method's performance under various settings, reinforcing its robustness and applicability in real-world scenarios.

Contribution 2: Constrained Spectral Clustering with Pairwise Fairness

The second contribution introduces a novel approach to incorporate pairwise fairness into constrained SC for equitable representation of demographic groups. Fairness is integrated into the pairwise constraint set by embedding a linear constraint that considers demographic group information into the SC objective function. A comprehensive set of experiments is conducted on multiple real-world datasets to evaluate the performance of the proposed approach. Clustering quality is assessed using the Rand Index, while fairness is measured using the balance metric. The results demonstrate that the proposed method effectively manages trade-offs, achieving a higher Rand Index compared to SC and improved balance compared to SC, constrained SC, and fair SC.

1.4.2 Fairness and Constraint Integration at Graph-Level (Pre-Processing Approach)

The second contributory chapter of this dissertation focuses on a graph-level pre-processing approach designed to integrate fairness and pairwise constraints into the SC process. This contribution is discussed below.

Contribution 3: Fair and Constraints-respecting Graph Construction for Spectral Clustering

The third contribution presents a novel pre-processing approach that embeds both fairness and pairwise constraints directly into the graph construction phase. Specifically, a Fair Constrained k-Nearest Neighbor (FC-kNN) graph is constructed, where each node's neighborhood satisfies group balance conditions, thereby mitigating disparate impact, while simultaneously integrating pairwise constraints to ensure that prior knowledge is respected.

The resulting fairness-aware, constraint-compliant graph is then used for SC, yielding cluster assignments that uphold demographic balance while maintaining clustering quality. Extensive evaluations on real-world datasets demonstrate that FC-kNN achieves competitive clustering performance while substantially reducing group bias relative to standard and constrained SC baselines.

1.4.3 Fairness in Unified Spectral Clustering (One-Step Optimization)

This final contributory chapter presents a unified fairness-aware framework for SC, emphasizing the integration of fairness into the OSSRC approach. The details of this contribution are discussed below.

Contribution 4: Fairness-aware One-Step Spectral Rotation Clustering

The fourth contribution introduces a Fair One-Step Spectral Rotation clustering (FOSSRC), which utilizes a novel Group Fairness Term (GFT) and integrates all independent stages of SC into a unified framework via adaptive graph learning. This GFT is capable of producing clusters with a balanced representation of sensitive groups. Furthermore, this work also introduces an algorithm using the alternating direction method of the multiplier framework to update the variables. Extensive experiments have been performed on various datasets and compared with benchmarked methods to validate the efficacy of the proposed method.

1.5 Organization of the Thesis

The thesis begins with an introduction to the problem, along with the background, motivation, and a review of related work. The core chapters present the proposed methods, followed by experimental evaluation and analysis. The thesis is structured into seven chapters, each focusing on a specific aspect of the research, as outlined below. The accompanying Figure 1.2 illustrates the work-wise organization, mapping the chapters to the corresponding

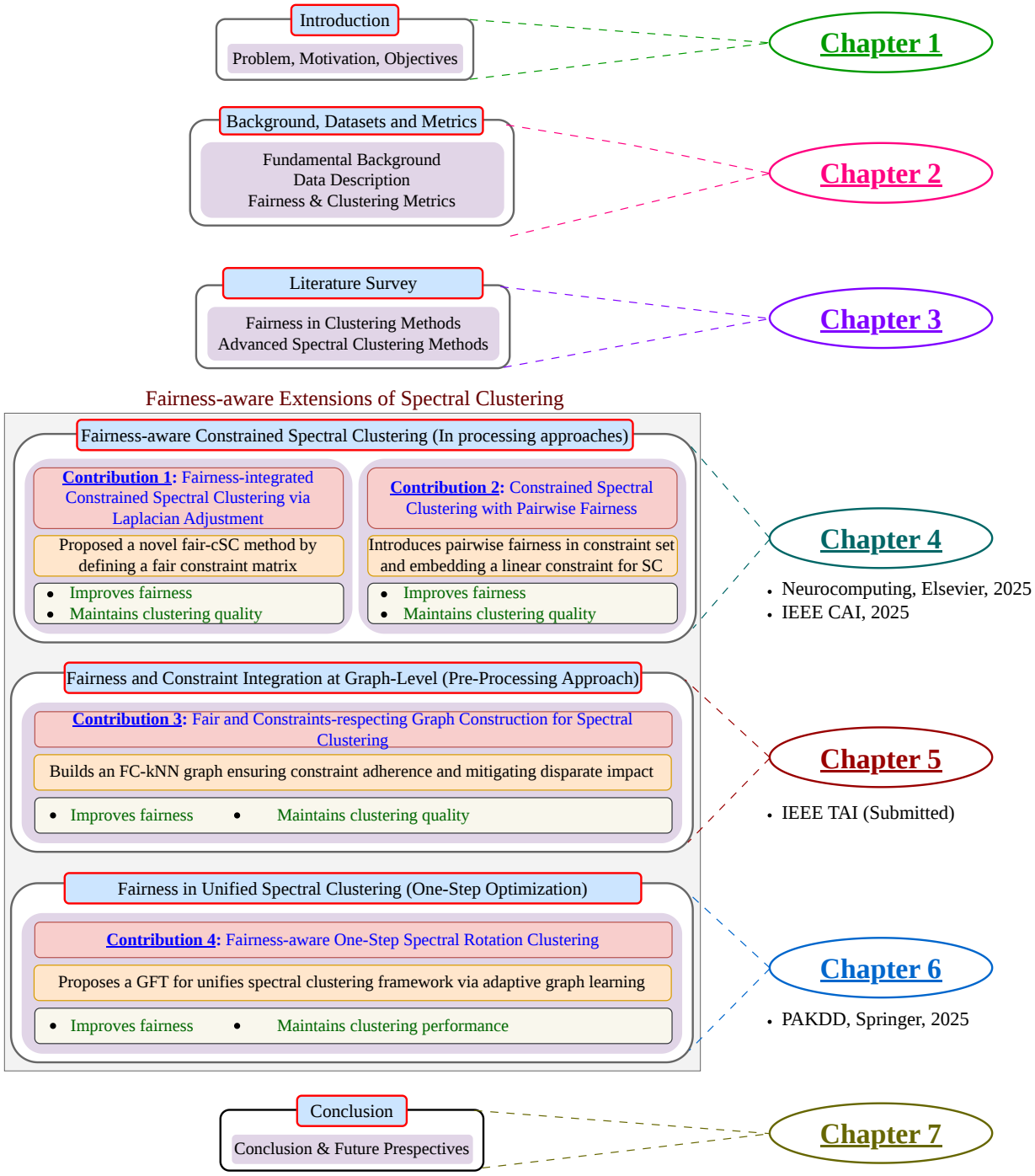


Figure 1.2: *Thesis organization showing chapter flow and research contributions.*

proposed methods and publications.

Chapter 1: Introduction

This chapter introduces the concept of fairness, with a particular focus on SC methods. It presents the background and significance of fairness, highlighting the motivation for

this dissertation and identifying key research gaps in recent literature. The chapter also outlines the main objectives, provides a brief overview of the chapter-wise contributions, summarizes the overall contributions of the dissertation, and describes the organization of the thesis.

Chapter 2: Background, Datasets and Metrics

This chapter establishes the theoretical and empirical groundwork essential for the research presented in this dissertation. It provides a detailed background on the principles of demographic fairness and SC paradigms, including their advanced variants. Furthermore, it describes the real-world datasets utilized for validation and defines the evaluation metrics employed to rigorously quantify clustering quality, fairness, and constraint satisfaction.

Chapter 3: Literature Survey

This chapter provides a comprehensive survey of existing literature at the intersection of spectral clustering and algorithmic fairness. It categorizes prior research based on the stage of fairness intervention and analyzes the limitations of current state-of-the-art methods. The chapter concludes with a summary of the research gaps identified in the current literature.

Chapter 4: Fairness-aware Constrained Spectral Clustering

The first contributory chapter focuses on fairness integration within constrained SC frameworks through two distinct mechanisms. The first part introduces a fair constraint matrix that establishes equitable ML and CL relationships. This is integrated into the clustering objective via Laplacian adjustment, influencing the eigen structure to promote balanced group representation. The second part embeds fairness within the pairwise constraint set by incorporating a linear constraint into the SC objective to ensure equitable cluster assignments. Collectively, these methods provide a framework

for preserving clustering quality while satisfying fairness requirements, supported by extensive experimental validation on real-world datasets.

Chapter 5: Fairness and Constraint Integration at Graph-Level

This chapter presents a graph-level pre-processing framework that integrates fairness and pairwise constraints simultaneously during the construction of the similarity graph. The primary contribution is the development of a FC-kNN method. This approach ensures group balance within each node's neighborhood while extending the graph to incorporate ML and CL constraints, thereby addressing fairness and domain knowledge before the clustering process begins. The chapter provides detailed algorithmic implementation and experimental validation to demonstrate the effectiveness of this graph-based strategy in producing fair, quality-preserving clusters.

Chapter 6: Fairness in Unified Spectral Clustering

The final contributory chapter presents a unified fairness-aware clustering framework that integrates a novel GFT to the cluster indicator matrix. This approach reduces imbalance in cluster assignments and includes a rigorous theoretical analysis of the GFT's effect on clustering outcomes. By integrating this transform into the OSSRC framework, the proposed method enables adaptive adjustments of cluster balance through iterative updates. This results in an efficient, single-step optimization process that achieves fairness without compromising computational performance, as demonstrated through both theoretical proofs and empirical evaluations.

Chapter 7: Conclusions and Future Perspectives

Finally, this chapter concludes the thesis with a summary of the overall contributions and outlines the potential directions for future research in fairness-aware clustering.

"The way to get started is to quit talking and begin doing."

~Walt Disney

2

Background, Datasets and Metrics

This chapter provides the essential background of the fundamental concepts required to understand the work presented in this thesis. It begins with the key notions of demographic fairness and their relevance to clustering, followed by an explanation of the principles of SC and its advanced variants. To bridge theory and practice, the chapter further details the real-world datasets used in the experimental studies and outlines the evaluation metrics employed to rigorously quantify clustering quality, fairness, and constraint satisfaction.

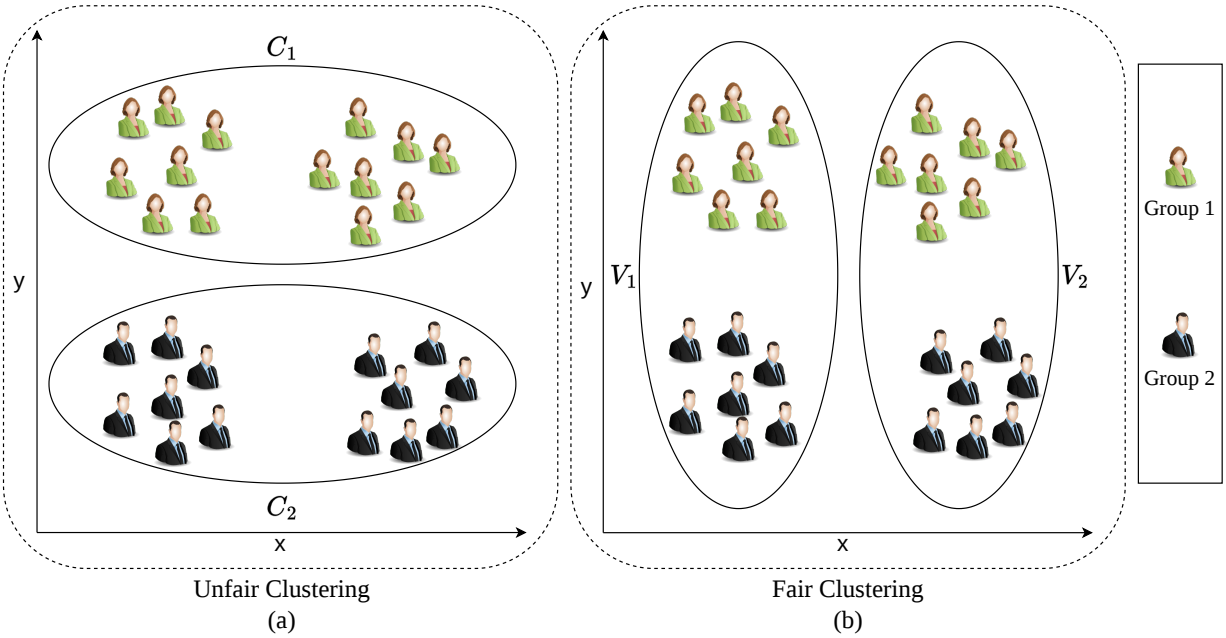


Figure 2.1: *An example of unfair and fair clustering. Given data points with two groups (male and female) are partitioned into two clusters. Clusters $C = C_1 \cup C_2$ depict unfair clustering, and clusters $C = V_1 \cup V_2$ depict fair clustering as samples of each group are equally represented in each cluster.*

2.1 Background

2.1.1 Demographic Fairness

In the context of clustering algorithms, fairness seeks to ensure that the formation of clusters does not favor or disadvantage individuals belonging to specific demographic groups defined by sensitive attributes. Without such considerations, clustering algorithms may unintentionally group individuals based on their sensitive attributes, thereby amplifying existing biases and leading to unfair outcomes. To address this issue, several notions of fairness have been proposed in the literature [22]. One of the most widely adopted notions is balance, introduced by Chierichetti *et al.* [24]. Balance focuses on achieving group-level fairness, requiring that the representation of each demographic group within every cluster reflects, as closely as possible, its overall distribution in the dataset. In other words, a cluster is considered balanced if no group is significantly overrepresented or underrepresented relative to its global proportion. Figure 2.1 provides an illustrative example of unfair and fair clustering outcomes using two demographic groups (Group 1 and Group 2). Formally, the balance of

a cluster is defined as follows:

Definition 1 (Balance). *A dataset* $X = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\}$ *is partitioned into* c *disjoint clusters* $\pi = \{\pi_1, \cdots, \pi_c\}, \bigcup_{\ell=1}^{c} \pi_\ell = X$. *Each instance in* X *is associated with a demographic group* $\mathbf{x}_i \in \{G_1, \cdots, G_m\}, \bigcup_{t=1}^{m} G_t = X$, *where each* G_t *corresponds to one group. Let* $\eta_t = \frac{|G_t|}{n}, t \in \{1, \cdots, m\}$ *be the proportion of group* G_t *in the dataset and* $\eta_t(\ell) = \frac{|\pi_\ell \cap G_t|}{|\pi_\ell|}, \ell \in \{1, \cdots, c\}$ *be the proportion of group* G_t *within cluster* π_ℓ. *The ideal balanace conditions is expressed as:*

$$\forall\, t \in \{1, \ldots, m\}, \qquad \frac{|G_t \cap \pi_\ell|}{|\pi_\ell|} = \frac{|G_t|}{n}. \tag{2.1}$$

and the balance of cluster π_ℓ *is defined as:*

$$Balance(\pi_\ell) = \min_{t \in \{1,\ldots,m\}} \left(\min \left(\frac{\eta_t}{\eta_t(\ell)}, \frac{\eta_t(\ell)}{\eta_t} \right) \right), \tag{2.2}$$

with the convention that $Balance(\pi_\ell) \in [0, 1]$.

Remark 1. *Balance*(π) *evaluates the degree of demographic fairness in resulting clusters. A higher balance value indicates that the proportion of each sensitive group* G_t *within every cluster* π_ℓ *(i.e.,* $\eta_t(\ell)$*) is close to its overall proportion in the dataset (i.e.,* η_t*). In the ideal case, when* $\eta_t(\ell) = \eta_t$ *for all groups and all clusters, the clustering will achieves its maximum balance* 1.

This Definition 1 of balance has been adopted in several fair SC methods to promote fairness across clusters [25, 26, 47].

2.1.2 Spectral Clustering

SC is a widely used graph clustering algorithm that leverages the spectral properties of graph representations of data to identify meaningful clusters. The key idea is to represent the dataset as a similarity graph and then transform this graph into a lower-dimensional space using the eigenvectors of its graph Laplacian matrix. After obtaining the spectral

embedding, a standard clustering algorithm such as k-means is applied to partition the points.

The objective of SC is to partition the graph in a way that minimizes edges between different clusters while maintaining strong connections within each cluster. This is often formalized through graph partitioning criteria, such as the Ratio Cut (RCut) or Normalized Cut (NCut) [27]. The process consists of two main steps:

1. **Graph Construction:** A similarity graph is constructed from the data to capture the similarities between pairs of data points.

2. **Spectral Embedding and Clustering:** The graph Laplacian is computed, and its eigenvectors are used to create a lower-dimensional representation of the data. Clustering is then applied in this new space.

Let $G = \{V, E, S\}$ be an undirected graph represented using a set of data points $\mathbf{X} = \{\mathbf{x}_1, \dots \mathbf{x}_n\}$. G includes $V = \{v_1, \cdots, v_n\}$, denotes the set of vertices, $E \subseteq V \times V$ denotes the set of edges connecting these vertices, and $\mathbf{S} \in \mathbb{R}^{n \times n}$ is the weighted adjacency matrix of G where $\mathbf{S}_{aa} = 0$ and $\mathbf{S}_{ab} \geq 0$ (similarity value of the edge E_{ab} between vertex V_a and V_b). The degree matrix $\mathbf{D} \in \mathbb{R}^{n \times n}$ is the diagonal matrix where $\mathbf{D}_{ii} = \sum_{j=1}^{n} \mathbf{S}_{ij}$, $\forall i \in \{1, \cdots, n\}$. The Laplacian matrix is expressed as $\mathbf{L} = \mathbf{D} - \mathbf{S}$, which captures the graph's structural information. SC aims to divide the graph G into $c \geq 2$ clusters $\pi = \{\pi_1, \cdots, \pi_c\}$, so the clusters are well separated and have high similarity within clusters. To achieve this, the objective function considered is the NCut [27], which finds the optimal partition by minimizing the cut of the graph. The NCut value is measured as follows:

$$\text{NCut}(\pi_1, \dots, \pi_c) = \sum_{i=1}^{c} \frac{Cut(\pi_\ell, \bar{\pi}_\ell)}{\text{Vol}(\pi_\ell)}, \tag{2.3}$$

where $\text{Cut}(\pi_\ell, \bar{\pi}_\ell) = \frac{1}{2} \sum_{i \in \pi_\ell, j \in \bar{\pi}_\ell} \mathbf{S}_{ij}$ shows the sum of the weights of the edges between the vertices in cluster π_ℓ and $\bar{\pi}_\ell$ and $\text{Vol}(\pi_\ell)$ denotes the sum of the edge weights in cluster π_ℓ. Minimizing the NCut in equation (2.3) is equivalent to solving the trace minimization

problem as NCut$(\pi_1, \ldots, \pi_c) = \text{Tr}(\mathbf{H}^T\mathbf{L}\mathbf{H})$ where $\mathbf{H} \in \mathbb{R}^{n\times c}$ is the cluster indicator matrix. The objective function of normalized SC is given as:

$$\min_{\mathbf{H}\in\mathbb{R}^{n\times c}} \text{Tr}(\mathbf{H}^T\mathbf{L}\mathbf{H}) \quad \text{subject to} \quad \mathbf{H}^T\mathbf{D}\mathbf{H} = \mathbf{I}_c. \tag{2.4}$$

By substituting $\mathbf{H} = \mathbf{D}^{-1/2}\mathbf{Y}$ for $\mathbf{Y} \in \mathbb{R}^{n\times c}$, The normalized SC computes an optimal $\mathbf{Y}$, which contains eigenvectors associated with c smallest eigenvalues of $\bar{\mathbf{L}} = \mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2}$. The problem in equation (2.4) becomes:

$$\min_{\mathbf{Y}\in\mathbb{R}^{n\times c}} \text{Tr}(\mathbf{Y}^T\bar{\mathbf{L}}\mathbf{Y}) \quad \text{subject to} \quad \mathbf{Y}^T\mathbf{Y} = \mathbf{I}_c \tag{2.5}$$

The final clustering step is performed by applying k-means to the rows of $\mathbf{Y}$ to produce the final clusters. This converts the continuous embedding into discrete cluster assignments.

The obtained clusters may still be demographically imbalanced, even if the algorithm accurately reflects the data structure. This motivates the exploration of fair SC, where fairness considerations are integrated into the spectral framework to obtain fair clusters.

2.1.2.1 Illustrative Example: The Spectral Clustering

To provide a practical intuition of the mathematical foundations discussed in the previous section, this section presents a step-by-step walk-through of the spectral clustering process using a synthetic 2D dataset (n=20). This illustrative example, visualized in Figure 2.2, serves as a baseline for understanding how graph-based partitions are formed before the introduction of fairness in later chapters.

- **Data and Graph Construction**

  Figure 2.2(a) illustrates the input data, where each point is associated with a sensitive attribute (Male/Female). In Figure 2.2(b), these points are transformed into a similarity graph G, where edges represent the proximity between nodes, calculated using a Gaussian kernel. This graph captures the local geometry of the data, which is essential for identifying non-linear structures. Based on this graph, the adjacency

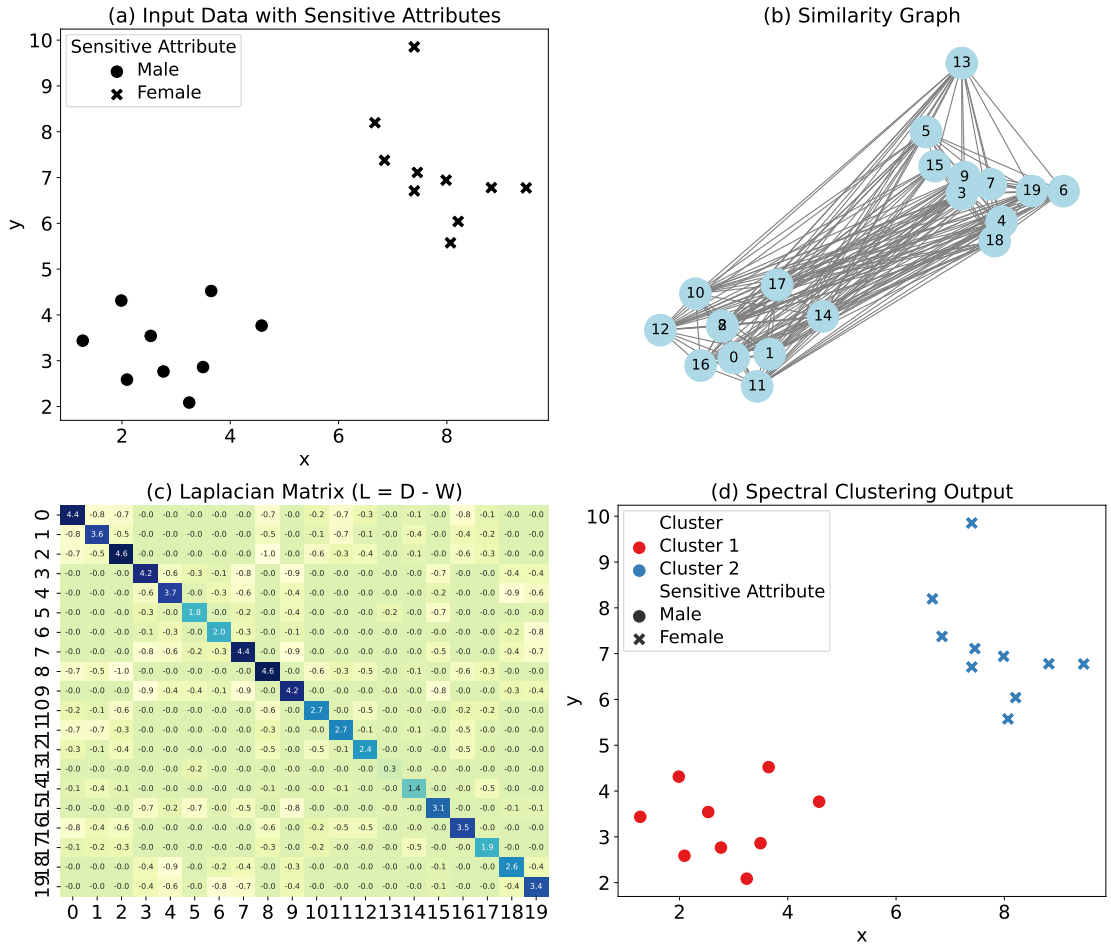


Figure 2.2: *Illustration of spectral clustering on a toy dataset showing (a) input data with sensitive attributes, (b) similarity graph construction, (c) graph Laplacian, and (d) final clusters.*

matrix W and degree matrix D are computed, which are then used to construct the graph Laplacian matrix.

- **Laplacian**

  From the graph, the unnormalized Laplacian matrix $L = D - W$ is derived. The heatmap in Figure 2.2(c) provides a numerical snapshot of the data structure. The values along the diagonal represent the degree of each node, while the off-diagonal values reveal the strength of connectivity between pairs. The emerging block-diagonal patterns are mathematically significant, as the eigen-decomposition of this matrix reveals the cluster components.

- **Cluster Assignment**

  By projecting the data into the spectral embedding space and applying k-means , we obtain the final clusters shown in Figure 2.2(d). While the algorithm successfully captures the geometric density of the blobs, it is important to observe that the clusters are determined without regard for the sensitive attributes. This fairness-blind partitioning highlights the potential for demographic imbalance, justifying the need for the fairness-constrained optimization frameworks proposed in this research.

2.1.3 Fair Spectral Clustering

Kleindessner *et al.* [25] were the first to introduce fairness in SC. They employed the balance condition equation (2.1) directly into the SC framework by deriving a linear constraint that enforces proportional representation of demographic groups within each cluster. The following lemma explains how to incorporate this aim into the NCut minimization objective using a linear constraint on cluster indicator $\mathbf{H}$.

Lemma 2.1.1. *For $t \in m$, let $\mathbf{f}^{(t)} \in \{0,1\}^n$ be the demographic group-indicator vector of G_t, given as:*

$$\mathbf{f}_i^{(t)} = \begin{cases} 1 & \text{if } \mathbf{x}_i \in G_t \\ 0 & \text{otherwise.} \end{cases} \tag{2.6}$$

Then, for every cluster $\ell \in c$:

$$\forall t \in [m-1] : \sum_{i=1}^{n} \left(\boldsymbol{f}_i^{(t)} - \frac{|G_t|}{n} \right) \boldsymbol{H}_{i\ell} = 0 \iff \forall t \in [m] : \frac{|G_t \cap \pi_\ell|}{|\pi_\ell|} = \frac{|G_t|}{n}. \tag{2.7}$$

Using the above lemma, $\mathbf{J}^T\mathbf{H} = \mathbf{0}$ is the balance constraint term where $\mathbf{J} \in \mathbb{R}^{n\times(m-1)}$ is the matrix with vectors: $\mathbf{f}^{(t)} - \left(\frac{|G_t|}{n}\right) \cdot \mathbf{1}_n$ as columns. This is used as a constraint in the objective function of SC, given as,

$$\begin{aligned} \min_{\mathbf{H} \in \mathbb{R}^{n\times c}} \quad & \mathrm{Tr}(\mathbf{H}^T\mathbf{L}\mathbf{H}) \\ \text{subject to} \quad & \mathbf{H}^T\mathbf{D}\mathbf{H} = \mathbf{I}_c, \\ & \mathbf{J}^T\mathbf{H} = \mathbf{0}_{(m-1)\times c}. \end{aligned} \tag{2.8}$$

The matrix $\mathbf{J}$ has rank $m-1$, and therefore $\mathrm{Rank}(\mathbf{J}^T) = m-1$ as well. It is necessary to assume $c \leq n-m+1$ to admit a feasible solution, since otherwise equation (2.8) does not have any solution. Let $\mathbf{P} \in \mathbb{R}^{n\times(n-m+1)}$ be any matrix whose columns form an orthonormal basis of the null-space of $\mathbf{J}^T$. The cluster indicator matrix can be expressed in the form $\mathbf{H} = \mathbf{P}\mathbf{Y}$, for $\mathbf{Y} \in \mathbb{R}^{(n-m+1)\times c}$, and the problem equation (2.8) becomes:

$$\begin{aligned} \min_{\mathbf{Y} \in \mathbb{R}^{(n-m+1)\times c}} \quad & \mathrm{Tr}(\mathbf{Y}^T\mathbf{P}^T\mathbf{L}\mathbf{P}\mathbf{Y}) \\ \text{subject to} \quad & \mathbf{Y}^T\mathbf{P}^T\mathbf{D}\mathbf{P}\mathbf{Y} = \mathbf{I}_c. \end{aligned} \tag{2.9}$$

Assuming that the graph G contains no isolated vertices, the matrix $\mathbf{P}^T\mathbf{D}\mathbf{P}$ is positive definite and hence has a positive definite square root, that is, there exists a matrix $\mathbf{K} \in \mathbb{R}^{(n-m+1)\times(n-m+1)}$ such that $\mathbf{P}^T\mathbf{D}\mathbf{P} = \mathbf{K}^2$. By substituting $\mathbf{Y} = \mathbf{K}^{-1}\mathbf{B}$ for $\mathbf{B} \in \mathbb{R}^{(n-m+1)\times c}$, the optimization problem in equation (2.9) transforms into:

$$\begin{aligned} \min_{\mathbf{B} \in \mathbb{R}^{(n-m+1)\times c}} \quad & \mathrm{Tr}\left(\mathbf{B}^T\mathbf{K}^{-1}\mathbf{P}^T\mathbf{L}\mathbf{P}\mathbf{K}^{-1}\mathbf{B}\right) \\ \text{subject to} \quad & \mathbf{B}^T\mathbf{B} = \mathbf{I}_c. \end{aligned} \tag{2.10}$$

A solution to problem equation (2.10) is obtained by matrix $\mathbf{B}$ whose columns consist of or-

thonormal eigenvectors associated with the c smallest eigenvalues of the matrix $\mathbf{K}^{-1}\mathbf{P}^T\mathbf{L}\mathbf{P}\mathbf{K}^{-1}$. This defines the fair spectral embedding, which in turn forms the basis of the fair normalized SC procedure presented in Algorithm 1.

Algorithm 1: Normalized Fair Spectral Clustering

Input : Adjacency matrix $\mathbf{S} \in \mathbb{R}^{n\times n}$, number of clusters $c \in \mathbb{N}$, group-membership vectors $\mathbf{f}^{(t)} \in \{0,1\}^n$ for $t \in [m]$

Output: A partition of $[n]$ into c clusters

1 Compute the graph Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{S}$, where $\mathbf{D}$ is the degree matrix.

2 Let $\mathbf{J}$ be a matrix with columns $\mathbf{f}^{(t)} - \frac{|G_t|}{n} \cdot \mathbf{1}_n, t \in [m-1]$.

3 Compute a matrix $\mathbf{P}$ whose columns form an orthonormal basis of the null space of $\mathbf{J}^T$.

4 Compute the square root of $\mathbf{P}^T\mathbf{D}\ \mathbf{P}$ as $\mathbf{K}$.

5 Compute some orthonormal eigenvectors corresponding to the c smallest (respecting multiplicities) eigenvalues of $\mathbf{K}^{-1}\mathbf{P}^T\mathbf{L}\mathbf{P}\mathbf{K}^{-1}$.

6 Let $\mathbf{B}$ be the matrix containing these eigenvectors as columns.

7 Apply k-means clustering to the rows of $\mathbf{H} = \mathbf{P}\mathbf{K}^{-1}\mathbf{B} \in \mathbb{R}^{n\times c}$ to obtain the final clustering.

2.1.4 Constrained Spectral Clustering

Pairwise Constraints

Wagstaff and Cardie [48] first introduced pairwise constraints to guide the clustering process. Their work demonstrated that clustering performance can be enhanced by incorporating prior knowledge about which data points should be grouped together and which should be separated. Building on this insight, constrained SC uses this idea by adding pairwise constraints, which are of two types:

1. **ML:** The pair of instances should belong to the same cluster.

$$ML = \{(\mathbf{x}_i, \mathbf{x}_j) \mid \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ should be in same cluster}\}$$

2. **CL:** The pair of instances should not belong to the same cluster.

$$CL = \{\mathbf{x}_i, \mathbf{x}_j \mid \mathbf{x}_i \text{ and } \mathbf{x}_j \text{ should not be in same cluster}\}$$

These constraints help guide the algorithm toward cluster structures that align with user requirements or domain knowledge. In practice, ML and CL constraints are usually represented in a constraint matrix $\mathbf{Q} \in \{-1, 0, +1\}^{n \times n}$ or in a graph where edges encode ML (+1) or CL (-1) relationships, given as:

$$\mathbf{Q}_{i,j} = \begin{cases} +1 & \text{if } (\mathbf{x}_i, \mathbf{x}_j) \in ML \\ -1 & \text{if } (\mathbf{x}_i, \mathbf{x}_j) \in CL \\ 0 & \text{otherwise.} \end{cases} \tag{2.11}$$

Incorporating Pairwise Constraints into Spectral Clustering

Several approaches have been proposed to incorporate pairwise constraints into the SC framework. In standard SC, an affinity matrix and the corresponding graph Laplacian are used to derive a low-dimensional embedding that reflects the similarity structure of the data. Constrained SC modifies this process by incorporating additional information from ML and CL constraints. This is typically achieved in two main ways:

1. **Similarity Graph Adjustment:** Pairwise constraints are incorporated by modifying the similarity graph before clustering. ML pairs are strengthened by increasing their edge weights or adding new edges, while CL pairs are weakened by reducing edge weights or removing edges. This ensures that the graph structure reflects the desired relationships before applying SC [35, 36].

2. **Optimization-based Modification:** Pairwise constraints are embedded explicitly into the SC objective. The optimization problem is modified so that the cluster assignment matrix satisfies the pairwise relationships. This integrates constraints during the clustering process itself [38, 40].

Both approaches integrate pairwise constraints directly into the SC framework and have been widely used to enhance the performance of clustering. However, these methods focus solely on constraint satisfaction and do not consider fairness among demographic groups.

This limitation highlights a clear research gap, the need for further extensions that preserve the strengths of constrained SC while accommodating fairness.

2.1.5 One-Step Spectral Rotation Clustering

The traditional two-step process of SC often leads to instability, as the final clusters depend not only on the spectral embedding but also on the randomness and sensitivity of the subsequent k-means steps. One-step spectral rotation addresses this limitation by integrating similarity learning and clustering into a unified framework. It works directly in the low-dimensional feature space of the original data, reducing noise and dimensionality issues. The OSSRC model jointly performs similarity graph learning, spectral embedding, and spectral rotation into a one-step optimization framework, enabling it to produce more stable and coherent clustering results.

Spectral Rotation

Spectral rotation provides a direct mechanism for converting the continuous spectral embedding into a discrete cluster indicator matrix, thereby avoiding the need for the k-means step, which often introduces instability and sensitivity to initialization. Instead, it applies an orthogonal transformation to the eigenvector matrix, producing a rotated embedding that aligns more closely with an ideal discrete cluster structure [49].

Let equation (2.5) be examined more closely; the relaxed solution to equation (2.5) is not unique. Indeed, for any optimal solution $\mathbf{Y}$, $\mathbf{YR}$ is another optimal solution, where R is an arbitrary orthonormal matrix. The goal is to find a proper orthonormal $\mathbf{R}$ such that the resulting $\mathbf{YR}$ are closer to the discrete indicator matrix solution set than the $\mathbf{Y}$ in k-means. Figure 2.3 illustrates the concept of spectral rotation, which rotates the spectral vectors more closely with the discrete indicator matrix compared to traditional spectral vectors.

Similar to k-means, the goal is to find the optimal $\mathbf{T}$ that minimizes the distance between

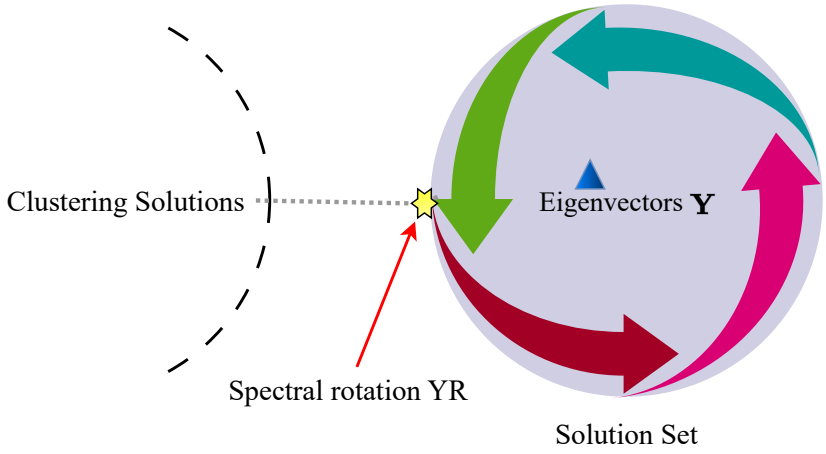


Figure 2.3: *A demonstration of spectral rotation. The curve on the left shows the discrete solution space. The grey circle corresponds to the set of solutions obtained through the orthonormal matrix* $\mathbf{R}$*. The blue triangle shows the outcome of the relaxed SC formulation, while the yellow star indicates the solution produced through an appropriate spectral rotation.*

$\mathbf{YR}$ and a cluster indicator matrix $\mathbf{T}$ [43]. This can be expressed mathematically as:

$$\begin{aligned}
&\min_{\mathbf{T}\in Ind,\,\mathbf{R}^T\mathbf{R}=\mathbf{I}} \|\mathbf{YR}-\mathbf{T}\|_F^2 \\
\Leftrightarrow\ &\min_{\mathbf{T}\in Ind,\,\mathbf{R}^T\mathbf{R}=\mathbf{I}} \mathrm{Tr}\Big[(\mathbf{YR}-\mathbf{T})^T(\mathbf{YR}-\mathbf{T})\Big] \\
\Leftrightarrow\ &\min_{\mathbf{T}\in Ind,\,\mathbf{R}^T\mathbf{R}=\mathbf{I}} \mathrm{Tr}\Big(\mathbf{R}^T\mathbf{Y}^T\mathbf{YR}-2\mathbf{R}^T\mathbf{Y}^T\mathbf{T}+\mathbf{T}^T\mathbf{T}\Big) \\
\Leftrightarrow\ &\min_{\mathbf{T}\in Ind,\,\mathbf{R}^T\mathbf{R}=\mathbf{I}} \big(\mathrm{Tr}(\mathbf{Y}^T\mathbf{Y})-2\ \mathrm{Tr}(\mathbf{Y}^T\mathbf{TR}^T)+\mathrm{Tr}(\mathbf{T}^T\mathbf{T})\big) \\
\Leftrightarrow\ &\min_{\mathbf{T}\in Ind,\,\mathbf{R}^T\mathbf{R}=\mathbf{I}} \Big\|\mathbf{Y}-\mathbf{TR}^T\Big\|_F^2 .
\end{aligned} \tag{2.12}$$

Here, the orthonormal constraint ensures that $\mathbf{T}$ best approximates $\mathbf{YR}$ among all discrete cluster indicator matrices, while $\mathbf{YR}$ is the optimal solution to equation (2.5).

Dimensionality reduction

High-dimensional data often lie on an underlying low-dimensional manifold that reflects their intrinsic structure. To recover this structure, dimensionality reduction methods are commonly used in SC and subspace learning. Locality Preserving Projection (LPP) is one such technique that preserves the local geometric relationships between data points. This

helps capture the intrinsic structure of the data and produces a robust spectral representation for clustering [50]. Following the principles of manifold learning, the goal is to find a projection matrix $\mathbf{W} \in \mathbb{R}^{d \times h}$ that can be combined with data matrix $\mathbf{X}$ to best approximate the low-dimensional manifold. The low-dimensional manifold data is embedded into the Laplacian graph, which gives the objective.

$$\min_{\mathbf{W}} \sum_{i,j}^{n} \|\mathbf{x}_i\mathbf{W} - \mathbf{x}_j\mathbf{W}\|_2^2 S_{ij} \quad \Leftrightarrow \quad \min_{\mathbf{W}} \mathrm{Tr}(\mathbf{W}^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{W}), \tag{2.13}$$

where $\mathbf{S}$ is the similarity graph matrix. For equation (2.13), the similarity matrix is typically constructed using a standard pairwise approach. However, redundant features may distort the learned subspace and degrade clustering performance. To address this, a $\ell_{2,1}$ -norm regularization is applied to the projection matrix $\mathbf{W}$, promoting feature-level sparsity and effectively performing feature selection across all samples while learning the subspace [51]. By combining subspace learning with feature selection, redundant features are reduced, a robust projection matrix is learned, and clustering performance is improved. The resulting formulation is:

$$\min_{\mathbf{W}} \ \mathrm{Tr}\left(\mathbf{W}^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{W}\right) + \alpha\|\mathbf{W}\|_{2,1} \quad \text{s.t.} \quad \mathbf{W}^T\mathbf{W} = \mathbf{I}, \tag{2.14}$$

where α is a non-negative parameter balancing subspace learning and feature selection. This formulation equation (2.14) yields a more robust spectral representation and improved clustering compared to traditional spectral methods such as RCut and NCut.

2.2 Datasets Description

The methods proposed in this dissertation are evaluated using real-world datasets commonly employed in fairness-aware machine learning research, all of which include sensitive attributes. Based on the comprehensive survey presented in [52], which provides a detailed taxonomy and discusses datasets curated for fairness research [53,54], the following datasets

were included in the experiments. Most of these datasets are collected from the UCI repository [55]. A summary of the datasets used in the experiments is provided in Table 2.1, followed by detailed descriptions of each dataset.

Table 2.1: *Summary of datasets*

Dataset	Samples	Features	Class	Sensitive Attribute	Domain	Reference
Adult Income	48,842	14	2	Gender	Socio economic	[56]
Bank	45,211	16	2	Marital status	Marketing	[57]
Crime	1994	127	2	Race	Law	[58]
ILPD	583	10	2	gender	Health	[59]
Hepatitis	155	19	2	Gender	Health	[60]
Credit	30,000	23	2	Gender	Finance	[61]
FriendshipNet	127	-	-	Gender	social networks	[62]
FacebookNet	155	-	-	Gender	social networks	[62]
DrugNet	193	-	-	Ethnicity	social networks	[63]
German Credit	1000	20	2	Gender	Finance	[64]
Catalonia	4753	133	2	Gender	Recidivism Analysis	[65]
Student	395	29	10	Gender	Education	[66]
Law	20798	11	2	Race	Education	[67]

1. **Adult Income**

   The Adult dataset [56] contains information from the 1994 US Census, including the target income details of individuals. The task traditionally involves predicting whether an individual's annual income exceeds $50,000. The dataset is also known as the "Census Income" dataset.

   - **Sensitive Attribute:** `Gender`

2. **Bank**

   The Bank [57] dataset contains records of phone calls made during a marketing campaign run by a Portuguese banking institution. The purpose is to predict if a client will subscribe (yes/no) to a term deposit using different attributes gathered during marketing campaigns.

   - **Sensitive Attribute:** `Martial status`

3. **Communities & Crime**

   The Crime dataset [58] contains crime-related and demographic information for com-

munities across the United States. The target is to predict the violent crime rate in the community.

- **Sensitive Attribute:** `Race`

4. **Indian Liver Patient Dataset (ILPD)**

   The Indian Liver Patient Dataset (ILPD) [59] dataset contains medical and demographic records of individuals collected for the diagnosis of liver disease. The prediction task is to determine whether a patient is affected by liver disease.

   - **Sensitive Attribute:** `Gender`

5. **Hepatitis**

   The Hepatitis [60] dataset's objective is to predict the patient's condition, i.e., whether the patient has been completely cured or not.

   - **Sensitive Attribute:** `Gender`

6. **Credit**

   The Credit dataset [61] contains information on customers' credit card payments, credit data, demographic factors, payment history, and bill settlements in Taiwan. The prediction task is to determine whether a customer will default on a payment.

   - **Sensitive Attribute:** `Gender`

7. **FriendshipNet**

   The FriendshipNet [62] dataset represents a friendship network among students, modeled as an undirected graph. It consists of 127 vertices, where each vertex corresponds to a student, and edges indicate reported friendships between pairs of students.In this dataset, sensitive attribute gender may impact the dynamics of student relationships and the patterns of friendships reported between students.

   - **Sensitive Attribute:** `Gender`

8. **FacebookNet**

   The FacebookNet [62] dataset represents a high school friendship network with 155 vertices, and an edge between two students indicates friendship on Facebook. In this dataset, attribute gender may influence online social connections, potentially shaping the formation of Facebook friendships across genders.

   - **Sensitive Attribute:** `Gender`

9. **DrugNet**

   The DrugNet [63] captures the acquaintanceship among drug users in Hartford, consisting of 193 vertices.

   - **Sensitive Attribute:** `Ethnicity`

10. **German Credit**

    The German Credit dataset [64] contains records of loan applicants from 1973 to 1975 collected as part of a study on computer-assisted credit assessment at a regional bank in southern Germany. Each instance includes demographic and financial attributes used to classify applicants as either good or bad credit risks.

    - **Sensitive Attribute:** `Gender`

11. **Catalonia**

    The Catalonia dataset [65] provides information on juveniles involved in the justice system in Catalonia.

    - **Sensitive Attribute:** `Gender`

12. **Student**

    The Student dataset [66] contains records from two Portuguese secondary schools, including grades in Mathematics and Portuguese, along with demographic, social, and school-related attributes. The task is to predict student achievement.

    - **Sensitive Attribute:** `Gender`

13. **Law**

    The Law dataset [67] originates from a 1991 survey conducted across 163 law schools in the United States. It contains admissions-related information.

    - **Sensitive Attribute:** `Race`

2.3 Evaluation Metrics

To assess the performance of clustering methods, two categories of metrics are employed: fairness metrics, which quantify the degree of representational fairness across sensitive groups, and clustering quality metrics, which measure how well the cluster assignments reflect the underlying data structure.

2.3.1 Fairness Metrics

1. **Balance:** The balance of a cluster measures how evenly the sensitive groups are represented equation (2.2). The higher the balance, the better the fairness. Since fairness should be maintained across all clusters, the balance of an entire clustering solution is defined as the minimum balance among all clusters:

$$\text{Balance} = \min_{\ell=1,\dots,c} (\text{Balance}(\pi_\ell)) \tag{2.15}$$

2.3.2 Clustering Quality Metrics

1. **Rand Index:** The Rand Index (RI) measures the clustering quality based on the clusters assigned and ground truth labels [68]. It evaluates how consistently pairs of points are assigned to the same or different clusters.

$$\text{RI} = \frac{a+b}{\binom{n}{2}}, \tag{2.16}$$

where a = number of pairs in the same cluster in both predicted and ground truth
b = number of pairs in different clusters in both predicted and ground truth
$\binom{n}{2}$ = total number of possible pairs in a dataset of size n RI ranges from 0 to 1, with higher values indicating better clustering performance.

2. **Accuracy:** Accuracy (Acc) measures the proportion of correctly assigned labels.

$$\text{Acc} = \frac{N_c}{N}, \qquad \text{Acc} \in [0, 1] \tag{2.17}$$

where N is the total number of samples, N_c denotes the number of samples whose predicted cluster labels match the corresponding ground-truth class labels.

3. **Normalized Mutual Information:** Normalized Mutual Information (NMI) measures the mutual dependence between the predicted clusters and the ground-truth classes. It is scaled between 0 and 1. A higher value shows a better match between the predicted clusters and the true classes.

$$\text{NMI}(\Omega, C) = \frac{2\, I(\Omega; C)}{H(\Omega) + H(C)} \in [0, 1], \tag{2.18}$$

where Mutual Information is

$$I(\Omega; C) = \sum_i \sum_j p_{ij} \log \left(\frac{p_{ij}}{p_i\, p_j} \right) \tag{2.19}$$

and the Entropy of predicted clusters and true labels

$$H(\Omega) = -\sum_i p_i \log p_i, \qquad H(C) = -\sum_j p_j \log p_j. \tag{2.20}$$

4. **Normalized Cut:** NCut measures the quality of a partition of a graph by evaluating the dissimilarity between clusters equation (2.3). A low NCut value indicates that the clusters are weakly connected by edges, representing better graph partition quality.

5. **Objective function cost:** SC minimizes the relaxed form of the graph partitioning objective. The quality of the resulting clustering is evaluated using the objective function value given in equation (2.4). A lower objective value corresponds to a better clustering that is more aligned with the graph structure.

6. **Constraints Satisfaction:** Constraint satisfaction measures how well a clustering algorithm satisfies the provided ML and CL constraints. A higher value indicates better adherence to the pairwise constraints.

$$\text{Constraint Satisfaction} = S_{ML} + S_{CL}, \tag{2.21}$$

where S_{ML} and S_{CL} are satisfied ML constraints and satisfied CL constraints, respectively.

7. **Silhouette Score:** The Silhouette Score measures the compactness and separation of clusters by comparing the intra-cluster distance with the nearest inter-cluster distance for each sample.

   For a single point i, the silhouette score is defined as:

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}$$

   where:

   - $a(i)$: average distance between point i and all other points in the same cluster (cohesion)
   - $b(i)$: minimum average distance between point i and points in any other cluster (separation)

   The final score for a clustering model is the average of $s(i)$ across all data points. The score ranges from -1 to 1, where a higher value indicates better-defined and well-separated clusters.

2.4 Summary

This chapter has established the theoretical and empirical foundations necessary for the subsequent research. By providing a detailed background on demographic fairness and spectral clustering paradigms, alongside a comprehensive description of the datasets and evaluation metrics. These metrics and datasets will serve as the rigorous benchmark for evaluating the effectiveness of the proposed frameworks in the following contributory chapters. Building upon this foundation, the next chapter provides a critical review of the existing literature and identifies the specific research gaps addressed by this work.

"Many of life's failures are people who did not realize how close they were to success when they gave up."

~Thomas A. Edison

3

Literature Survey

Building upon the foundational concepts and experimental framework established in the previous chapter, this chapter provides a comprehensive survey of the existing literature regarding fairness in clustering, with a specific focus on spectral clustering paradigms. Additionally, it discusses advanced SC approaches, including constrained SC and one-step spectral rotation methods. Finally, the chapter highlights the limitations and research gaps in the current literature, motivating the need to integrate fairness into advanced SC frameworks.

3.1 Fairness in Clustering Methods

Fairness in clustering has gained significant attention as these automated systems are increasingly scrutinized for bias and reviewed in [22, 69]. A fairness notion, called balance, was first proposed by Chierichetti *et al.* [24] for clustering algorithms. It states that each cluster should have the same proportion of sensitive groups. For this, fairlet decomposition algorithms were proposed to enforce balance in center-based clustering for two groups. In order to do this, the original points were divided into small balanced clusters called fairlets, and vanilla clustering was applied. Following this, Backurs *et al.* [70] improved the work of [24] by proposing a scalable fair clustering algorithm based on faster fairlet computation. In this work, a tree metric-based non-linear approximation algorithm was proposed for fair k-medians, achieving nearly linear time complexity. Subsequently, Rosner *et al.* [71] worked for fair k-center clustering by proposing an approximation algorithm and generalizing it to multiple sensitive groups. Beyond fairlet, Bera *et al.* [53] generalized the seminal work of [24] by using a bounded representation of groups in clusters. First, the classical clustering problem is solved and transformed into a fair assignment, converting various clustering objectives into fair solutions while accommodating different sensitive groups simultaneously. Subsequently, Ziko *et al.* [72] proposed a variational approach for fair clustering that incorporates a fair penalty term based on KL divergence into the standard clustering algorithms, including both prototype-based and graph-based. This approach achieved the required fairness while ensuring clustering quality simultaneously. Further, to address the challenge of fairness and heterogeneous data in multi-view clustering, Zheng *et al.* [73] proposed a fairness multi-view clustering method, which integrates group fairness and employs contrastive and non-contrastive learning to handle heterogeneous data. The method achieves effective and unbiased clustering.

3.2 Fairness in Spectral Clustering Methods

A few work has been done on fairness in SC and other graph clustering methods. Kleindessner *et al.* [25] proposed a fair SC method to obtain balanced clusters with respect to sensitive groups. In this work, a balance constraint is designed following the definition in [24] and incorporated a linear constraint on the cluster indicator into the spectral relaxation to achieve fair clusters. Subsequently, a scaled version of [25] is proposed by Wang *et al.* [26], in which a technique is introduced for the spectral computation of fair SC by integrating null-space projection and Hotelling's deflation to speed up the computation. Further, Anagnostopoulos *et al.* [74] address fairness in densest sub-graph problems with two colors, proposing a greedy approximation combined with a spectral method. Gupta *et al.* [30] explored SC under an individual-level fairness constraint. The representation graph is utilized to encode individual-level fairness. They evaluated the effectiveness using a fair-planted partition algorithm derived from the representation graph. Li *et al.* [75] considers group fairness in normalized-cut graph partitioning to ensure that sensitive groups are fairly represented in each cluster. A two-step strategy called FNM is presented for the spectral method. First, an augmented Lagrangian term is aligned with fairness criteria for fair embedding. The next step uses a rounding approach to form c clusters while maintaining partition quality and fairness. Fu *et al.* [31] proposed a model that unifies clique-preserving connections and fairness in SC for temporal graphs. In addition, Zhou *et al.* [47] introduced a new regularized term that simultaneously handles fairness and cluster imbalance problems. The term was integrated into the SC ensemble framework, resulting in fair clusters with equal capacity. Similarly, Yang *et al.* [76] proposed SFSC by introducing the notion of scale fairness, addressing the cluster size imbalance. By incorporating entropy-based adjustments into the Laplacian matrix, SFSC achieves more balanced cluster sizes while maintaining comparable clustering quality. Further, they also proposed a fair SC framework by introducing fair Laplacian matrices to address group, individual, and scale fairness [77].

Recent work has focused on addressing fairness in SC by targeting the graph construction step. A comprehensive review, as summarized in [78, 79], reveals a notable trend in fair

graph methods. Zhang *et al.* [33] examined the impact of graph construction on fair SC and then provided a graph construction method to learn the graph from noisy data. A unified approach for fair SC is proposed by integrating all stages into a single objective function using a dual Laplacian framework.

3.3 Advanced Spectral Clustering Methods

Several studies have focused on enhancing the quality and robustness of SC. These methods address limitations such as noisy data, sensitivity to k-means, and the need to incorporate domain knowledge. Developments in this area include constrained SC and one-step SC, which are reviewed in the following subsections.

3.3.1 Constrained Spectral Clustering

As discussed earlier, pairwise constraints can be incorporated into SC through two main approaches: similarity graph adjustment and optimization-based modification. This section reviews the methods developed under each approach.

- **Similarity Graph Adjustment**

  Constrained clustering enhances the standard framework by integrating domain knowledge as pairwise constraints. These pairwise constraints, ML and CL, are introduced to adapt an incremental clustering method [80], which significantly improves performance. For this purpose, several constrained graph clustering have been reviewed in [41]. Kamvar *et al.* proposed a constrained SC method [37], which modifies the affinity matrix with 1's and 0's for ML and CL constraints, respectively. The graph Laplacian is then modified accordingly, enabling the method to efficiently use the external information for clustering the data. Subsequently, Kulis *et al.* [81] proposed to encode +1 for ML and -1 for CL into the similarity matrix and then utilized kernel k-means rather than SC to partition the data. In [36], Lu *et al.* proposed to propagate pairwise constraints within the affinity matrix using a Gaussian process to extend

constraints to unconstrained objects. This method is particularly suited for two-class problems, with heuristics proposed for extending it to multi-class. Lu *et al.* [82] proposed a method for constrained SC through exhaustive constraint propagation. They break down global propagation into independent label propagation subproblems that can be solved in quadratic time. This method thoroughly propagates pairwise constraints throughout the dataset. As a result, it creates a refined similarity matrix and improves clustering accuracy. They later extended this work with a graph-based interpretation, a stronger motivation for similarity adjustment, broader experimental validation, and a nontrivial extension to multi-source constraint propagation [83].

Furthermore, Jia *et al.* [84] proposed Semi-supervised Adaptive Symmetric NMF. This method learns the similarity matrix and clustering results jointly. By using ML and CL constraints in one optimization process, their approach improves the graph and achieves better clustering performance. Subsequently, Jia *et al.* [85] discussed that sequential propagation of traditional pairwise constraints and affinity refinement can increase errors in SC. They proposed a joint optimization framework that learns both the propagation and affinity matrices at the same time. This approach results in a more accurate cluster structure. Nie *et al.* [86] address the challenges of handling CL constraints and multi-class clustering in semi-supervised learning. They proposed a pairwise constrained structured optimal graph using a novel cannot-link graph regularization and Laplacian rank constraint. Their method achieves significant performance with few pairwise constraints. In [87], Li *et al.* introduced an instance-level multi-model constraint propagation approach, which constructs a unified affinity by combining several modality affinities at the data instance level.

- **Optimization-based Modification**

  Yu *et al.* [88] modified the optimization problem by encoding pairwise information through a subspace projection method. This proposed solution for the constrained eigenvalue problem, however, can handle only ML constraints. Subsequently, a regularization to the spectral embedding is introduced to enforce the pairwise constraints

by Li *et al.* [89], which directly applies to multi-class, managing both ML and CL and efficiently propagates sparse constraints. Similarly, a flexible constrained SC method was proposed by Wang *et al.* [38] to encode the pairwise constraints during optimization. It uses a user-specified parameter α to lower the bound of how effectively the external constraints are satisfied. This algorithm is detailed with two-way clustering. Further, Wang *et al.* extended this from two-way clustering to multi-way clustering [90]. Alternatively, Chen *et al.* [91] provided a unified paradigm that generalizes instance-level constraints into space-level. In this, the distortion measure assesses the similarity and dissimilarity of data points and combines it with the Ncut criterion. Wacquet *et al.* [40] proposed explicitly adding the pairwise constraints as penalization terms with the SC objective function. Then, the optimization problem of the eigenvalue system associated with a signed Laplacian matrix was addressed.

Furthermore, Cucuringu *et al.* [92] introduced a faster-generalized eigenvalue problem for 2-way constrained SC. Liu *et al.* [93] presented the compressed version of the [92] by retaining the same accuracy with less computational time. In a different direction, a multi-view matrix completion method for constrained multi-view SC was introduced by Zhao *et al.* [94], which ensures consistency across views via pairwise constraints and achieves global optimization. Ding *et al.* [95] integrated pairwise constraints with graph cut optimization and proposed a space-efficient semi-supervised SC method using approximate kernel k-means and hidden Markov random fields. Similarly, Chen *et al.* [96] and Zhang *et al.* [97] integrated pairwise constraints into multi-view SC. Yu *et al.* [98] addressed computational complexity, two-step procedures, and constraint violations in constrained SC, proposing an efficient semi-supervised balanced-cut method that adapts to pairwise constraints and partial labels. In another effort, a method for constrained clustering with weak label priors (i.e., pairwise constraints and cluster ratio) was proposed in a unified framework by Zhang *et al.* [99]. This integrated pairwise constraints as a regularizer with the spectral embedding and label proportion was integrated as a constraint with the spectral rotation to enhance the clustering

performance.

3.3.2 One-Step Spectral Clustering

Various one-step SC methods have been proposed in the literature to address the limitations of traditional two-step SC. Yang *et al.* [45] proposed a unified framework for SC to learn discrete clustering labels. The method performs better clustering by integrating a discrete rotation functionality and an adaptive robust module with $l_{2,p}$ loss. Building on this, Kang *et al.* [42] proposed a method that integrates similarity graph learning, continuous label learning, and discrete cluster labels into a unified framework. The model ensures that the similarity matrix has exactly c connected components for c clusters, refines solutions for better discrete labels, and develops multiple kernel learning for adaptivity. Similarly, Pang *et al.* [43] present a joint model of SC that incorporates spectral embedding and rotation to enhance accuracy. They use a scaled orthonormal cluster indicator matrix to reduce approximation errors.

To enhance robustness to outliers and imbalanced data, Tong *et al.* [100] proposed a robust one-step SC method using self-paced learning. It integrates a missing value mapping matrix and a self-paced regularizer to reduce the outliers and fuses affinity matrix learning with spectral decomposition. *Zhu et al.* [101] extended these ideas with a unified deep SC framework that integrates all steps of SC, enhancing robustness to noise and high-dimensional data. The framework implements a two-task deep clustering model with linear activation functions to produce effective clustering results. However, its use of linear activations may limit the ability to capture nonlinear relationships. Further addressing imbalance, Wen *et al.* [44] present one-step spectral rotation clustering for imbalanced high-dimensional data, which addresses clustering of imbalanced, high-dimensional data by integrating self-paced learning with spectral rotation in a unified framework. It performs sample selection to balance classes and combines subspace learning with feature selection for dimensionality reduction, improving clustering performance and reducing bias toward majority classes. Further, they introduced a balance constraint to generate clusters of similar size [102]. Finally,

Zhu *et al.* [103] proposed a novel subspace-guided spectral embedding method for one-step SC. The method introduces an affine self-expression regularization to learn a robust similarity matrix and incorporates spectral rotation as a penalty to produce clustering-friendly embeddings.

Fairness in one-step SC has also been explored by Zhang *et al.* [33], who introduced a unified fair SC approach that employs a node-adaptive graph filter for graph construction, effectively learning graphs from noisy data. Furthermore, they integrated the balance constraint proposed in [25] into their framework as a constrained optimization problem.

3.4 Research Gap Analysis: Need for Fair and Quality-Preserving Spectral Clustering

The literature reviewed in the previous sections highlights substantial progress in both fairness-aware clustering and enhanced SC. However, the developments across these areas have largely evolved independently, resulting in a persistent gap when both fairness and clustering quality must be achieved simultaneously.

- Existing fair SC methods → Weak cluster quality
- Existing advanced SC methods → No fairness consideration
- Trade-off between fairness and quality

The literature shows a clear trade-off between fairness and clustering quality. Fairness-aware SC methods improve representational fairness but often reduce cluster quality. On the other hand, advanced SC methods like constrained SC and one-step spectral rotation improve accuracy, stability, and robustness but do not consider fairness. Consequently, no existing framework simultaneously ensures fairness while retaining clustering quality. Addressing this gap requires an approach that integrates fairness with advanced SC techniques to generate clusters that are both fair and robust.

Research Gap: No SC method simultaneously integrates fairness with advanced quality-enhancing techniques, such as constrained SC or one-step spectral rotation, to achieve clusters that are both equitable and quality-preserving.

Addressing this gap is crucial for real-world applications where both equitable representation and the preservation of cluster quality are essential.

3.4.1 Advances Made Through This Thesis

To bridge these identified gaps, this dissertation introduces a series of methodological advances. Table 3.1 provides a synthesis of these research gaps alongside the proposed advancements in the subsequent contributory chapters.

Table 3.1: *Synthesis of Research Gaps and Proposed Advances*

Research Gap in Literature	Proposed Advancements in Thesis	Chapter
Conflict between Fairness and ML/CL constraints	Fairness in ML/CL constraint set	Chapter 4
Lack of fairness in graph-building	FC-kNN Balanced Graph Construction	Chapter 5
Multi-stage spectral clustering	GFT-based One-Step Optimization	Chapter 6

3.5 Summary

This chapter has provided an extensive survey of the existing literature, categorizing fair clustering methods and advancements in SC techniques and identifying critical limitations in current spectral frameworks. The analysis highlights a significant research gap: the need for methods that can seamlessly integrate fairness into quality-improving SC frameworks. This motivates the research presented in this dissertation. The following chapters introduce the

proposed methodologies for addressing these challenges and achieving a balanced trade-off between fairness and clustering performance.

"The best way to predict your future is to create it."

~Abraham Lincoln

4

Fairness-aware Constrained Spectral Clustering

There are two contributions in this contributory chapter. In the first part, a method is proposed that modifies the graph Laplacian formulation to enforce pairwise constraints and fairness jointly. In this work, for the first time, fairness is introduced in the constraint set. Further, the second part of this chapter embeds fairness with the pairwise constraint representation itself, ensuring equitable treatment of demographic groups while performing constrained SC. Together, these contributions address the limitations of existing methods by enhancing fairness without substantially degrading clustering quality.

4.1 Fairness-integrated Constrained Spectral Clustering via Laplacian Adjustment

4.1.1 Introduction

Though the prior works considered fairness in SC for the input data points, fairness in constrained SC, integrating external information such as ML and CL constraints, has not yet been explored in the literature. The constrained Spectral Clustering (cSC) [40] method incorporates pairwise constraints by formulating a constraint matrix and embedding it into the SC objective using a convex combination approach. Its objective function consists of two terms: one optimizing SC criteria for input data points, and the other is a penalization term for pairwise constraints. However, while cSC effectively incorporates pairwise constraints, it does not account for fairness in the clustering process. As a result, biases in the data or pairwise constraints could unfairly affect the clusters, leading to unequal representation of sensitive groups across clusters. To address this gap, a novel fair-constrained Spectral Clustering (fair-cSC) method is proposed, which ensures that sensitive groups are equally represented in the resulting clusters, even in the presence of pairwise constraints.

The significant contributions made in this work are summarized as follows:

- This work introduces fairness in ML and CL constraints by defining a fair constraint matrix, establishing "fair ML" and "fair CL" constraints. This matrix is integrated into the SC objective, ensuring equitable representation in pairwise relationships.
- To further enhance fairness, a balance constraint [25] is incorporated into the overall objective function, leading to the proposed novel fair-cSC method. This method prioritizes fair ML and CL constraints, creating a fair-constrained spectral space where the derived clusters are equitable.
- The performance of the proposed fair-cSC method is evaluated against the baseline cSC method using key metrics, including balance, RI, objective function value, and number of satisfied pairwise constraints. Extensive experiments conducted on six real-

world datasets demonstrate that the proposed method achieves a higher balance than the baseline cSC while maintaining competitive clustering quality, as indicated by the RI. Additionally, it achieves a lower objective function value and satisfies a greater number of fair ML and CL constraints.

This work introduces a novel and generalized framework for incorporating fairness into cSC. The proposed fair-cSC method is evaluated against unfair versions of cSC methods, as the primary objective is to establish a foundational baseline. This is the first attempt to introduce fairness into pairwise constraints, ensuring equal representation of sensitive groups in pairwise relationships as well as in the input data points.

4.1.2 Baseline: constrained Spectral Clustering (cSC)

The proposed fairness-aware method is built upon the cSC framework [40]. In this framework, the objective function incorporates both ML and CL constraints to guide the clustering process. The cSC optimization problem is defined as follows:

$$J_{cSC} = \lambda \cdot J_{SC} + (1 - \lambda) \cdot J_{CM}, \tag{4.1}$$

In equation (4.1), J_{SC} is the SC criteria on input data points, and J_{CM} denotes the penalization term for pairwise ML and CL constraints. J_{CM} is obtained by,

$$\min_{\mathbf{u} \in \mathbb{R}^n} \mathbf{u}^T \mathbf{L_Q} \mathbf{u}, \tag{4.2}$$

where $\mathbf{u}$ is the cluster indicator vector; $\mathbf{L_Q} = \mathbf{D_Q} - \mathbf{Q}$ is the Laplacian matrix; $\mathbf{D_Q}$ is the degree matrix, and $\mathbf{Q}$ is the constraint matrix given in equation (2.11). Equation (4.1) can be re-written in terms of modified Laplacian $\mathbf{L}_{cSC}$ as,

$$\begin{aligned} \min_{\mathbf{u} \in \mathbb{R}^n} \quad & \mathbf{u}^T \mathbf{L}_{cSC} \mathbf{u} \\ s.t. \quad & \mathbf{u}^T \mathbf{D}_{cSC} \mathbf{u} = 1, \\ & \mathbf{D}_{cSC} \mathbf{u} \perp 1, \end{aligned} \tag{4.3}$$

where,

$$\begin{aligned} \mathbf{L}_{cSC} &= \lambda \cdot \mathbf{L} + (1-\lambda) \cdot \mathbf{L}_Q, \\ \mathbf{D}_{cSC} &= \lambda \cdot \mathbf{D} + (1-\lambda) \cdot \mathbf{D}_Q, \\ \mathbf{S}_{cSC} &= \lambda \cdot \mathbf{S} + (1-\lambda) \cdot \mathbf{Q}. \end{aligned} \tag{4.4}$$

Here, λ is the regularization factor that balances the contributions of SC and pairwise constraints. Note that the above formulation only gets affected in the graph Laplacian matrix.

4.1.3 Proposed Methodology

This section presents the fair-cSC methodology, starting with the problem formulation, definition of fair constraints and the construction of the fair constraint matrix, followed by the fair-cSC method.

4.1.3.1 Problem Formulation

The fairness in terms of balance is applicable only to the input data points provided for clustering [25]. When pairwise constraints are also part of the input, one has to extend the fairness definition to the pairwise constraints as well. However, a straightforward extension is not feasible due to two reasons: (1) the matrix $\mathbf{J}$ in equation (2.8) accounts for input data point-related constraint as per lemma 1, and (2) the pairwise constraints are to be honored even when they violate the balance definition. Therefore, fairness must be ensured, given that the ML and CL constraints need to be handled separately.

4.1.3.2 Fair Constraint Set

The goal is to ensure fairness not only in the input data but also in the external constraint sets. To achieve this, a definition of fair ML and fair CL is introduced, followed by a fair constraint matrix.

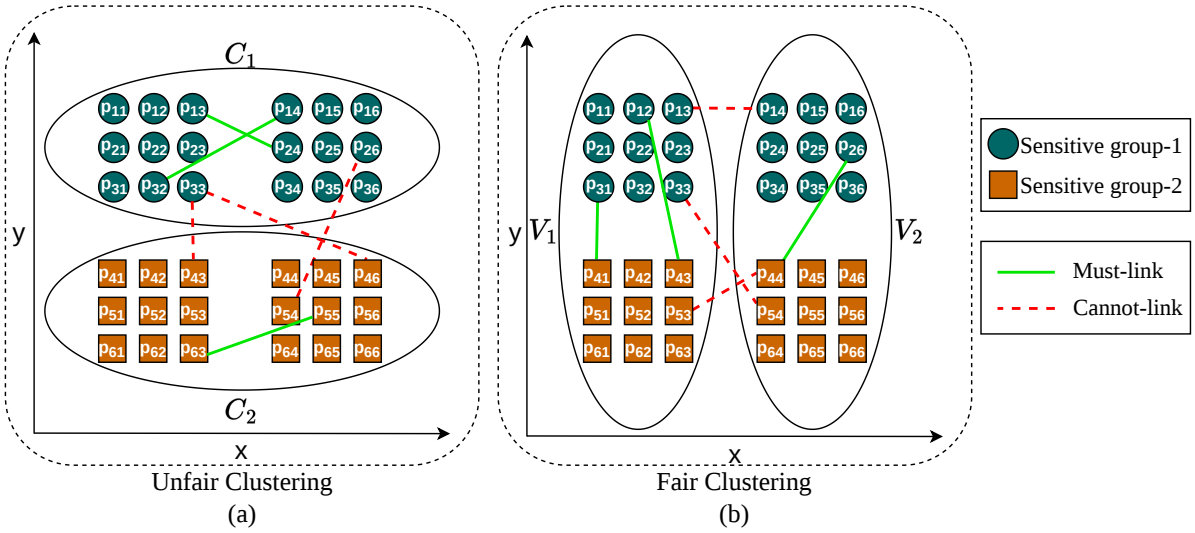


Figure 4.1: *Visual example of unfair and fair clustering. Given data points with two sensitive groups (green and orange colors) are partitioned into two clusters. Clusters $C = C_1 \cup C_2$ depict unfair clustering, and clusters $C = V_1 \cup V_2$ depict fair clustering as samples of each group are equally represented in each cluster.*

For example, consider the pairwise constraints in Figure 4.1(a),"Unfair clustering", including ML pairs $\{(p_{13}, p_{24}), (p_{32}, p_{14}), (p_{63}, p_{55})\}$, and CL pairs $\{(p_{43}, p_{33}), (p_{33}, p_{46}), (p_{26}, p_{54})\}$. With these constraints, the resulting clusters are C_1 and C_2. The ML and CL constraints at the end of clustering are satisfied, leading to improved cluster quality. However, the fairness of the pairwise constraints is not explicitly taken into account, as the main objective of specifying the ML and CL constraints is to improve the clustering quality. Now, Consider the pairwise constraints in Figure 4.1(b), "Fair clustering", including ML pairs $\{(p_{41}, p_{31}), (p_{43}, p_{12}), (p_{44}, p_{26})\}$, and CL pairs $\{(p_{13}, p_{14}), (p_{33}, p_{54}), (p_{53}, p_{44})\}$. With these constraints, the resulting clusters are V_1 and V_2. These constraints, on the other hand, do not improve the clustering quality. They reduce the clustering quality due to the involvement of fair connections of ML and CL. However, they lead to improving fairness in the obtained clusters. This is due to the fact that data points in the ML set belong to different sensitive groups, which leads to an increase in diversity, which in turn leads to an increase in balance. Similarly, the data points belonging to the same sensitive group are specified to be in different clusters. This also results in increasing balance. The above two examples present two extreme scenarios: unfair ML and CL constraints and fair ML and CL constraints.

Definition 2 (Fair ML)**.** *A pair-wise constraint* $(\mathbf{x}_i, \mathbf{x}_j)$ *is said to be fair ML if* $(\mathbf{x}_i, \mathbf{x}_j) \in ML$*, where* $\mathbf{x}_i$ *belong to a group (e.g., female), and* $\mathbf{x}_j$ *belong to a different group (e.g., male).*

Let $\mathbf{Q}_{ML}$ be a matrix coded using the given ML constraint set as follows:

$$\mathbf{Q}_{ML}(i,j) = \begin{cases} 1 & \text{if } (\mathbf{x}_i, \mathbf{x}_j) \in ML \\ 0 & \text{otherwise.} \end{cases} \tag{4.5}$$

$\mathbf{Q}_{ML}$ codes only the ML constraints. Let $\mathbf{Q}_{dsg}$ (where `dsg` stand for different sensitive groups) be a matrix that codes whether a pair of data points $(\mathbf{x}_i, \mathbf{x}_j)$ belong to different sensitive groups based on their sensitive attribute value. For each pair $(\mathbf{x}_i, \mathbf{x}_j)$, if their sensitive attribute values differ, then $\mathbf{Q}_{dsg}(i,j)$ is assigned a value of $+1$; otherwise 0. The following $\mathbf{Q}_{dsg}$ matrix specifies the fair ML constraints:

$$\mathbf{Q}_{dsg}(i,j) = \begin{cases} 1 & \text{if } g(\mathbf{x}_i) \neq g(\mathbf{x}_j) \\ 0 & \text{otherwise,} \end{cases} \tag{4.6}$$

where, $g(\mathbf{x}_i)$ and $g(\mathbf{x}_j)$ represent the sensitive attribute values of data point $\mathbf{x}_i$ and $\mathbf{x}_j$, respectively; that is, $g(\mathbf{x}_i) = t$ if $\mathbf{x}_i \in G_t$. Here, $g(\mathbf{x}_i) \neq g(\mathbf{x}_j)$ ensures that the linked data points belong to different sensitive groups, promoting diversity within clusters. The fair must-link matrix is coded using $\mathbf{Q}_{ML}$ and $\mathbf{Q}_{dsg}$ as:

$$\mathbf{Q}_{ML^{fair}} = \mathbf{Q}_{ML} \odot \mathbf{Q}_{dsg}. \tag{4.7}$$

The notation $\odot$ represents the element-wise multiplication of the two matrices $\mathbf{Q}_{ML}$ and $\mathbf{Q}_{dsg}$.

Definition 3 (Fair CL)**.** *A pair-wise constraint* $(\mathbf{x}_i, \mathbf{x}_j)$ *is said to be fair CL if* $(\mathbf{x}_i, \mathbf{x}_j) \in CL$*, where* $\mathbf{x}_i$ *and* $\mathbf{x}_j$ *belongs to the same sensitive group (e.g.,* $\mathbf{x}_i$ *and* $\mathbf{x}_j$ *are females or* $\mathbf{x}_i$ *and* $\mathbf{x}_j$ *are males).*

Let $\mathbf{Q}_{CL}$ be a matrix coded using the given CL constraint set as follows:

$$\mathbf{Q}_{CL}(i,j) = \begin{cases} -1 & \text{if } (\mathbf{x}_i, \mathbf{x}_j) \in CL \\ 0 & \text{otherwise.} \end{cases} \tag{4.8}$$

$\mathbf{Q}_{CL}$ codes only the CL constraints. Let $\mathbf{Q}_{ssg}$ (where `ssg` stand for same sensitive groups) be a matrix that codes whether a pair of data points $(\mathbf{x}_i, \mathbf{x}_j)$ belong to same sensitive groups based on their sensitive attribute value. For each pair $(\mathbf{x}_i, \mathbf{x}_j)$, if their sensitive attribute values are same, then $\mathbf{Q}_{ssg}(i,j)$ is assigned a value of -1; otherwise 0. The following $\mathbf{Q}_{ssg}$ matrix specifies the fair CL constraints:

$$\mathbf{Q}_{ssg}(i,j) = \begin{cases} -1 & \text{if } g(\mathbf{x}_i) = g(\mathbf{x}_j) \\ 0 & \text{otherwise.} \end{cases} \tag{4.9}$$

Here, $g(\mathbf{x}_i) = g(\mathbf{x}_j)$ ensures that the points from the same sensitive groups can not be clustered together, promoting balance. The fair cannot-link matrix is coded using the above $\mathbf{Q}_{ssg}$ and $\mathbf{Q}_{CL}$ matrices as:

$$\mathbf{Q}_{CL^{fair}} = \mathbf{Q}_{CL} \odot \mathbf{Q}_{ssg}. \tag{4.10}$$

A comparative overview of the fair ML and fair CL constraints is provided in Table 4.1. This table highlights each constraint type's definitions, mathematical formulations, purposes, and illustrative examples. When such pairwise constraints are judiciously given importance while performing clustering, it improves fairness and retains cluster quality.

Table 4.1: *Description of Fair ML and Fair CL constraints.*

Constraint Type	**Definition**	**Mathematical Representation**	**Purpose**	**Example**
Fair ML	Ensures pair of data points belong to different sensitive groups.	$g(x_i) \neq g(x_j)$ (if $\mathbf{x}_i, \mathbf{x}_j$ linked)	Promotes diversity within clusters.	Two points representing different genders are linked.
Fair CL	Ensures pair of data points belong to the same sensitive group.	$g(x_i) = g(x_j)$ (if $\mathbf{x}_i, \mathbf{x}_j$ not linked)	Prevents same-sensitive-group points from being clustered together.	Two points representing the same gender can not linked.

The more fair the ML and fair CL constraints are, the more diversity there is in the resulting clusters, which in turn increases the balance. The fair constraint matrix $\mathbf{Q}_{fair}$

captures this notion. The number of fair ML constraints is given by: $\mathbf{e}^T\mathbf{Q}_{ML^{fair}}\mathbf{e}$ where $\mathbf{e}$ is vector of all 1's. Similarly, the number of fair CL constraints is given by: $\mathbf{e}^T\mathbf{Q}_{CL^{fair}}\mathbf{e}$. The total number of fair constraints is: $\mathbf{e}^T\mathbf{Q}_{ML^{fair}}\mathbf{e} + \mathbf{e}^T\mathbf{Q}_{CL^{fair}}\mathbf{e}$. This is used as the weight in the fair constraint matrix $\mathbf{Q}_{fair}$. The $\mathbf{Q}_{fair}$ matrix retains those ML and CL constraints that are not fair with constant weight. These constraints are identified as unfair ML and CL, ensuring their consistent contribution to the formulation. The fair constraint matrix is further defined in Definition 4.

Definition 4 (Fair constraint matrix)**.** *The elements of the fair constraint matrix ($\mathbf{Q}_{fair}$) are defined as follows. The entries of the matrix represent constraints from ML and CL, classified as either fair or unfair.*

$$\mathbf{Q}_{fair}(i,j) = \begin{cases} +(\mathbf{e}^T\mathbf{Q}_{ML^{fair}}\mathbf{e} + \mathbf{e}^T\mathbf{Q}_{CL^{fair}}\mathbf{e}) & \text{if } (\mathbf{x}_i, \mathbf{x}_j) \in \textit{fair ML} \\ -(\mathbf{e}^T\mathbf{Q}_{ML^{fair}}\mathbf{e} + \mathbf{e}^T\mathbf{Q}_{CL^{fair}}\mathbf{e}) & \text{if } (\mathbf{x}_i, \mathbf{x}_j) \in \textit{fair CL} \\ +1 & \text{if } (\mathbf{x}_i, \mathbf{x}_j) \in \textit{unfair ML} \\ -1 & \text{if } (\mathbf{x}_i, \mathbf{x}_j) \in \textit{unfair CL} \\ 0 & \textit{otherwise.} \end{cases} \tag{4.11}$$

4.1.3.3 fair-cSC

Figure 4.2 illustrates the proposed workflow for fair-cSC. As shown in the figure, fair ML and CL constraints are introduced to ensure fairness in the constraint set. The constraint matrix $\mathbf{Q}$ is transformed into a fair constraint matrix $\mathbf{Q}_{fair}$, which prioritizes fair ML and CL constraints. In addition, the balance constraint [25] is incorporated with the cSC criteria. Then, the fair-cSC algorithm partitions the data into clusters that satisfy fairness while maintaining clustering quality.

With the definition of the $\mathbf{Q}_{fair}$ matrix, which assigns a higher weight to fair constraints, the formulation of fair-cSC is now established. The objective function of the fair-cSC is given

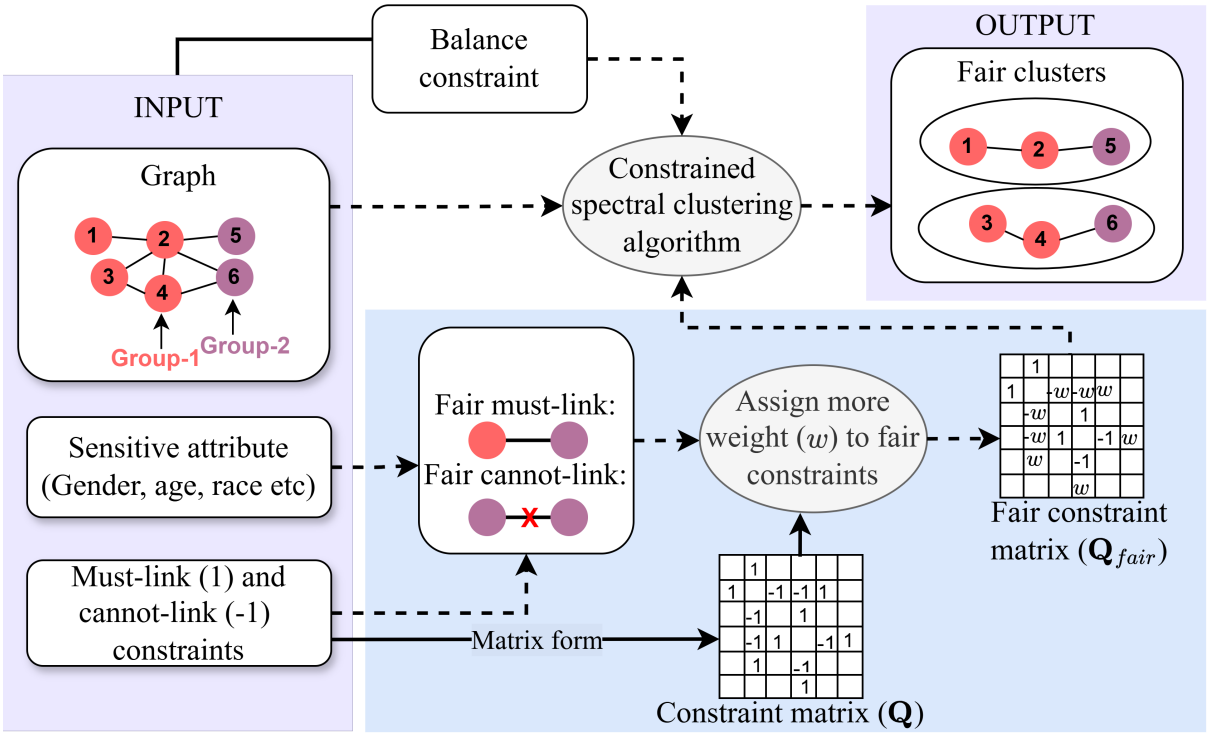


Figure 4.2: *The illustration of the workflow of the proposed fair-cSC method. w is the weight based on the number of fair constraints equation (4.11). The balance constraint is $\mathbf{J}^T\mathbf{u} = 0$ as in equation (2.8).*

in the following equation:

$$J_{fair-cSC} = \lambda \cdot J_{SC} + (1 - \lambda) \cdot J_{fair-CM}, \tag{4.12}$$

where λ is the regularization factor to balance the contribution of these two terms. The first term in equation (4.12) is the SC objective function given in equation (2.4), is expressed as:

$$J_{SC} = \mathbf{u}^T\mathbf{L}\mathbf{u}. \tag{4.13}$$

The second term in equation (4.12) ensures fairness in the pairwise constraints, which is expressed as:

$$J_{fair-CM} = \mathbf{u}^T\mathbf{L}_{\mathbf{Q}_{fair}}\mathbf{u}, \tag{4.14}$$

where $\mathbf{L}_{\mathbf{Q}_{fair}} = \mathbf{D}_{\mathbf{Q}_{fair}} - \mathbf{Q}_{fair}$, and $\mathbf{D}_{\mathbf{Q}_{fair}}$ is the degree matrix of $\mathbf{Q}_{fair}$. It is to be noted that the use of the same symbol, $\mathbf{u}$, in equation (4.13) and (4.14) reflects the unified

nature. In equation (4.13), Obtaining a cluster indicator vector $\mathbf{u}$ yields a minimum cut in the graph. In equation (4.14), Obtaining a cluster indicator vector $\mathbf{u}$ adheres to the given pairwise constraints, ensuring that the clustering respects these constraints. By combining these two terms, a cluster indicator vector $\mathbf{u}$ is found that yields minimum cut as well as maximally satisfies pairwise constraints. From this perspective, $\mathbf{u}$ in equation (4.13) and (4.14) are the same.

Substituting equation (4.13) and (4.14) in equation (4.12) yields the following equation:

$$J_{fair-cSC} = \mathbf{u}^T(\lambda \cdot \mathbf{L} + (1-\lambda) \cdot \mathbf{L}_{\mathbf{Q}_{fair}})\mathbf{u} = \mathbf{u}^T\mathbf{L}_{fair-cSC}\mathbf{u}, \tag{4.15}$$

where

$$\mathbf{L}_{fair-cSC} = \mathbf{D}_{fair-cSC} - \mathbf{S}_{fair-cSC}, \tag{4.16}$$

with

$$\mathbf{D}_{fair-cSC} = \lambda \cdot \mathbf{D} + (1-\lambda) \cdot \mathbf{D}_{\mathbf{Q}_{fair}}, \tag{4.17}$$

$$\mathbf{S}_{fair-cSC} = \lambda \cdot \mathbf{S} + (1-\lambda) \cdot \mathbf{Q}_{fair}. \tag{4.18}$$

The eigenvectors of the Laplacian matrix $\mathbf{L}_{fair-cSC} = \lambda \cdot \mathbf{L} + (1-\lambda) \cdot \mathbf{L}_{\mathbf{Q}_{fair}}$ are used to obtain the fair representation of constrained spectral space. The eigenvalues of $\mathbf{L}_{fair-cSC}$ can exhibit negative values as a result of the negative edge weights introduced by the CL constraints. The signed Laplacian matrix [104] is utilized to address this issue.

$$\bar{\mathbf{L}}_{fair-cSC} = \bar{\mathbf{D}}_{fair-cSC} - \mathbf{S}_{fair-cSC}, \tag{4.19}$$

where $\bar{\mathbf{D}}_{fair-cSC}$ denotes the signed degree matrix as follows:

$$\bar{\mathbf{D}}_{fair-cSC}(i,i) = \sum_{j=1}^{n} |\mathbf{S}_{fair-cSC}(i,j)|. \tag{4.20}$$

In the fair version of normalized cSC, the following equation must be solved:

$$\begin{aligned} \min_{\mathbf{u}\in\mathbb{R}^n} \quad & \mathbf{u}^T\bar{\mathbf{L}}_{fair-cSC}\mathbf{u} \\ s.t. \quad & \mathbf{u}^T\bar{\mathbf{D}}_{fair-cSC}\mathbf{u} = 1, \\ & \mathbf{J}^T\mathbf{u} = 0, \end{aligned} \tag{4.21}$$

where $\mathbf{J}^T\mathbf{u} = 0$ represents the balance constraint, which is taken from equation (2.8) to enforce fairness on the given input data. The proposed method to incorporate the fair version of pairwise constraints with balance constraint [25] into the cSC consists of solving equation (4.21). The method for solving equation (4.21) is explained next, where the rank($\mathbf{J}$) = rank($\mathbf{J}^T$) = $m-1$ and assuming that $c \leq n-m+1$ otherwise equation (4.21) does not have any solution. Substituting $\mathbf{u} = \mathbf{P}\mathbf{y}$ for $\mathbf{y} \in \mathbb{R}^{(n-m+1)}$, where $\mathbf{P} \in \mathbb{R}^{n\times(n-m+1)}$ forms an orthonormal basis of the null space of $\mathbf{J}^T$, then problem in equation (4.21) becomes:

$$\min_{\mathbf{y}\in\mathbb{R}^{(n-m+1)}} \mathbf{y}^T\mathbf{P}^T\bar{\mathbf{L}}_{fair-cSC}\mathbf{P}\mathbf{y}, \quad \text{s.t. } \mathbf{y}^T\mathbf{P}^T\bar{\mathbf{D}}_{fair-cSC}\mathbf{P}\mathbf{y} = 1. \tag{4.22}$$

Under the assumption that G does not have any isolated vertices, then the matrix $\mathbf{P}^T\bar{\mathbf{D}}_{fair-cSC}\mathbf{P}$ is positive definite. This ensures that there exists another matrix $\mathbf{K} \in \mathbb{R}^{(n-m+1)\times(n-m+1)}$, which is also positive definite such that $\mathbf{P}^T\bar{\mathbf{D}}_{fair-cSC}\mathbf{P} = \mathbf{K}^2$. Now substituting $\mathbf{y} = \mathbf{K}^{-1}\mathbf{v}$ for $\mathbf{v} \in \mathbb{R}^{(n-m+1)}$, problem in equation (4.22) becomes:

$$\min_{\mathbf{v}\in\mathbb{R}^{(n-m+1)}} \mathbf{v}^T\bar{\mathcal{L}}_{fair-cSC}\mathbf{v}, \quad \text{s.t. } \mathbf{v}^T\mathbf{v} = 1, \tag{4.23}$$

where $\bar{\mathcal{L}}_{fair-cSC} = \mathbf{K}^{-1}\mathbf{P}^T\bar{\mathbf{L}}_{fair-cSC}\mathbf{P}\mathbf{K}^{-1}$. For $c = 2$ clusters, the fair constrained spectral space is determined using the eigenvector that corresponds to the second smallest eigenvalue of $\bar{\mathcal{L}}_{fair-cSC}$. For more than two clusters ($c > 2$), the solution is obtained by applying k-means to the eigenvectors corresponding to the c smallest eigenvalues of $\bar{\mathcal{L}}_{fair-cSC}$. This way of solving equation (4.21) raises fairness to normalized cSC as given in Algorithm 2.

Algorithm 2: fair-cSC Algorithm

Input : Input data $\mathbf{X} \in \mathbb{R}^{n \times d}$; ML and CL constraints; Sensitive attribute values; threshold λ

Output: Clusters π_1 and π_2.

Construct the affinity matrix $\mathbf{S}$ from $\mathbf{X}$.
Construct $\mathbf{Q}$ matrix as in equation (2.11).
Compute the number of fair constraints $(\mathbf{e}^T\mathbf{Q}_{ML^{fair}}\mathbf{e} + \mathbf{e}^T\mathbf{Q}_{CL^{fair}}\mathbf{e})$.
if $(\mathbf{e}^T\mathbf{Q}_{ML^{fair}}\mathbf{e} + \mathbf{e}^T\mathbf{Q}_{CL^{fair}}\mathbf{e}) = 0$ **then**
    Use equation (4.2) as J_{CM}.
end
else
    Construct $\mathbf{Q}_{fair}$ using equation (4.11).
    Use equation (4.14) as $J_{fair-CM}$.
end
Calculate the Laplacian $\bar{\mathbf{L}}_{fair-cSC}$ via equation (4.19).
Compute $\mathbf{J}$ matrix with columns $\mathbf{f}^{(t)} - \frac{|G_t|}{n} \cdot \mathbf{1}_n,\ t \in m$.
Construct matrix $\mathbf{P}$ whose columns form an orthonormal basis for the null space of $\mathbf{J}^T$.
Construct $\mathbf{K}$ as the square root of $\mathbf{P}^T\bar{\mathbf{D}}_{fair-cSC}\mathbf{P}$.
Compute the eigenvector aligns with the second-smallest eigenvalue of $\mathbf{K}^{-1}\mathbf{P}^T\bar{\mathbf{L}}_{fair-cSC}\mathbf{P}\mathbf{K}^{-1}$.
return $\mathbf{u} = \mathbf{P}\mathbf{K}^{-1}\mathbf{v}$ *(cluster assignment).*

4.1.4 Experiments and Results

The experiments are performed in Python 3.10 by utilizing standard libraries on a system specification with Intel Core i7-8700 CPU 3.20GHz and 12GB RAM. Following the literature [22] to assess the fairness, the experiments have been conducted on the following datasets: Adult [56], Bank [57], Credit [61], Hepatitis [60], ILPD [59], and Communities and Crime [58]. For analysis, the number of ML and CL constraints is varied to interpret the fairness and quality of the clusters. A given number of pairs are randomly selected from the dataset, and ML and CL constraints are generated based on the class labels of the data points. To establish the proposed work as a generalized framework, the cSC method [40] is used as a baseline for comparative evaluation with the proposed fair-cSC method. The cSC and proposed fair-cSC methods are executed on multiple generated sets of pairwise constraints, and the performance is averaged after every ten sets of constraints.

4.1.4.1 Parameters and Baselines

There are three parameters in the proposed method and the baseline method. These are (1) given the input data points, the graph construction method, and the associated parameters. (2) the regularization term λ in equation (4.12) and (3) the total number of pairwise constraints.

$k-$ nearest neighbor graph construction method is used. In this, the number of neighbors, namely, k, is a parameter. A fixed value of $k = \sqrt{n}$ for the fair-cSC and baseline method is used. For all the experiments, a fixed value of $\lambda = 0.5$ for both fair-cSC and baseline cSC is used. That is giving equal weightage to J_{SC} term and J_{CM} terms in the baseline method, and equal weightage is given to J_{SC} and $J_{fair-CM}$ in fair-cSC. The number of pairwsie constraints are generated randomly, varying from 1% to 20% of a number of input data points. For the ablation study, a fixed number of pairwise constraints is generated instead of a number of proportional to the dataset.

The proposed method is compared for fairness measure with unfair variants of constrained SC such as the semi-supervised Spectral Learning (SL) [37] that incorporates pairwise constraints within the affinity matrix. Flexible Constrained Spectral Clustering (FCSC) [38] that adds the ML and CL constraints as a degree of belief constraint with the objective function of SC. The cSC [40], which is followed in this work, integrates ML and CL constraints with the optimization criteria of SC. The clustering results on balance for all baselines are reported across different numbers of pairwise constraints.

4.1.4.2 Evaluation Against Fairness

Figure 4.3 presents the balance values under varying numbers of pairwise constraints. In the Adult dataset, the red colored line denotes the balance achieved using the proposed fair-cSC method. Green, blue, and orange colored lines denote the balance achieved using the baselines cSC, SL, and FCSC, respectively. The red dashed line parallel to the x-axis denotes the maximum achievable balance value in the Adult dataset for the sensitive attribute `gender`. It is observed that the proposed method achieves an increased balance

Table 4.2: *The balance values of fair-cSC method and the compared baseline cSC with varying Pairwise Constraints (PC) and cluster numbers (c).*

Datasets	PC	c=2		c=3		c=4		c=5		c=6	
		cSC	**fair-cSC**	**cSC**	**fair-cSC**	**cSC**	**fair-cSC**	**cSC**	**fair-cSC**	**cSC**	**fair-cSC**
Adult	5%	0.8828	0.9324	0.8319	0.8408	0.7253	0.7712	0.7381	0.7586	0.7303	0.7792
	10%	0.8512	0.8772	0.7457	0.7553	0.7776	0.8131	0.6740	0.7172	0.5177	0.7376
	15%	0.8038	0.8232	0.7781	0.8068	0.4376	0.7550	0.3786	0.6682	0.4215	0.6299
	20%	0.8070	0.8532	0.6993	0.7584	0.5713	0.7116	0.3647	0.6713	0.6713	0.6758
Bank	5%	0.1465	0.1461	0.1275	0.1384	0.1199	0.1239	0.1132	0.1231	0.1031	0.1053
	10%	0.1460	0.1535	0.1192	0.1397	0.1067	0.1249	0.0950	0.1037	0.0926	0.0928
	15%	0.1340	0.1462	0.1228	0.1257	0.1082	0.1123	0.0979	0.0983	0.0963	0.1009
	20%	0.1519	0.1642	0.1304	0.1406	0.1035	0.1110	0.0902	0.1031	0.0898	0.1018
Credit	5%	0.8410	0.8639	0.7353	0.8541	0.6550	0.6890	0.6072	0.6507	0.5776	0.5985
	10%	0.7963	0.8342	0.6980	0.7848	0.6442	0.6728	0.6243	0.6686	0.6189	0.6403
	15%	0.7944	0.8595	0.7126	0.7832	0.6237	0.7146	0.5814	0.6426	0.5822	0.6135
	20%	0.8006	0.8796	0.6885	0.7573	0.6099	0.6581	0.5628	0.5887	0.5616	0.6119
Hepatitis	5%	0.1420	0.1544	0.0594	0.0828	0.0111	0.0125	0.0	0.0	0.0	0.0
	10%	0.0806	0.1342	0.0	0.0276	0.0259	0.0429	0.0	0.0	0.0	0.0
	15%	0.0618	0.0748	0.0339	0.0357	0.0068	0.0255	0.0	0.0111	0.0	0.01
	20%	0.1145	0.1293	0.0336	0.0983	0.0111	0.0162	0.0211	0.0262	0.0	0.0083
ILPD	5%	0.2623	0.2921	0.1751	0.2046	0.1101	0.1438	0.0817	0.0913	0.0524	0.0632
	10%	0.2571	0.2772	0.1275	0.1470	0.0752	0.0810	0.1150	0.1201	0.1254	0.1402
	15%	0.2177	0.2303	0.0416	0.0676	0.1031	0.1436	0.0825	0.0926	0.0195	0.0300
	20%	0.2327	0.2445	0.1534	0.1531	0.0411	0.0646	0.0342	0.0428	0.0676	0.0676
Crime	5%	0.1352	0.7928	0.1198	0.4278	0.0303	0.2781	0.0054	0.0395	0.0075	0.0149
	10%	0.3892	0.7407	0.0670	0.4821	0.0609	0.3323	0.0059	0.0573	0.0025	0.0064
	15%	0.2942	0.4839	0.1595	0.4303	0.0036	0.2752	0.0093	0.0887	0.0036	0.0250
	20%	0.2228	0.4241	0.1507	0.4409	0.0218	0.3021	0.0069	0.0751	0.0005	0.0098

value compared to the baseline methods across all generated numbers of pairwise constraints. A similar observation is made for the Bank, Credit, Hepatitis, ILPD, and Crime datasets. The baseline methods produced solutions with a low balance and became discriminatory due to the additional pairwise constraints.

On the other hand, the fair-cSC method maintains a balanced distribution of sensitive groups in the clusters. With increasing pairwise constraints, fair-cSC solutions maintained a significant increase in balance compared to the SL, FCSC, and cSC. Another observation from Figure 4.3 is that achieving improvements in the balance becomes a challenge beyond a certain number of pairwise constraints.

The proposed fair-cSC method is also compared with the baseline cSC considering c-way clustering and varying numbers of pairwise constraints. Table 4.2 shows the balance values for each number of clusters (c) with pairwise constraints taking 5%, 10%, 15%, and 20% of data for six datasets. In this analysis, the balance value is calculated by varying c from 2 to 6. The table clearly shows that the fair-cSC method achieves a significantly higher balance

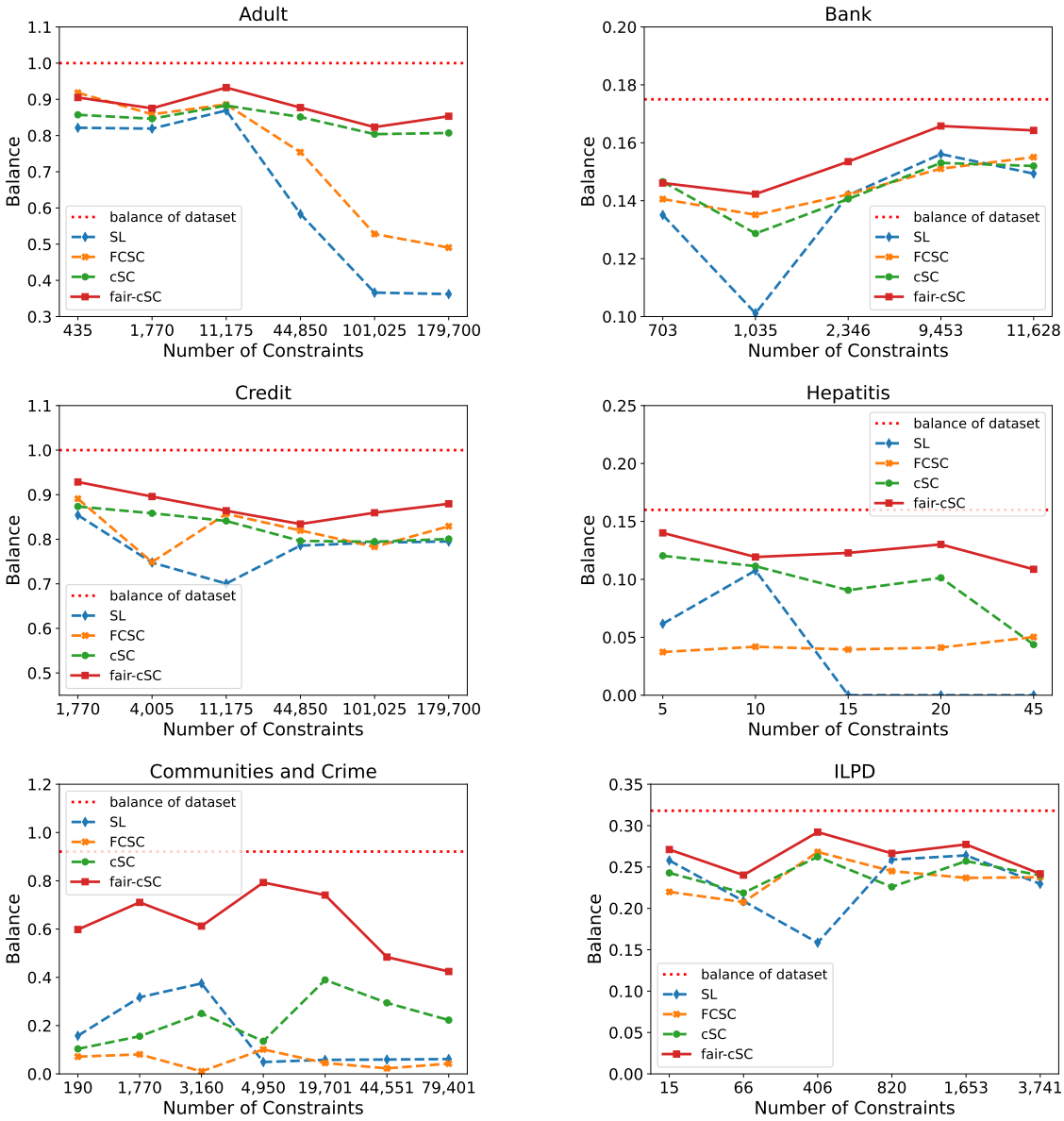


Figure 4.3: *Comparison of the proposed fair-cSC with baseline methods in terms of balance for varying numbers of pairwise constraints.*

than the cSC on all six datasets. It can also be observed that as the number of clusters increases, the balance value tends to degrade. This is expected, as the balance value is measured based on the minimum balance across all the clusters, and an increasing number of clusters generally makes the task more challenging. However, despite this degradation, the proposed fair-cSC method consistently shows improved balance compared to the baseline cSC. This demonstrates the effectiveness of the proposed method, regardless of the number of clusters or pairwise constraints. It is to be noted that the Hepatitis dataset produces a balance value of 0 for $c = 5$ and $c = 6$ clusters due to the inherent imbalance in the dataset itself, with 69 males and 11 females (overall balance: 0.1594). The potential reason could be with the increasing number of clusters c, the lower number of females makes it challenging to distribute them across clusters evenly. Nevertheless, the fair-cSC method improves the balance with higher number of pairwise constraints of 15% and 20%.

4.1.4.3 Evaluation Against Clustering Quality

For a comparative assessment of clustering quality, the Rand index is computed for the proposed and baseline methods. Figure 4.4 shows the Rand index values on both fair-cSC and cSC methods. The obtained cluster quality of the proposed fair-cSC is at par with the baseline cSC method. With an increasing number of pairwise constraints, the Rand index increases in the proposed method as well as the baseline method. However, slight variations in the rand index values are observed, indicating that the fair constraints proposed in fair-cSC subtly affected the clustering quality. The proposed method selects the pairwise constraints, considering clustering quality and fairness. This leads to the fair-cSC maintaining clustering quality that is comparable to cSC. The Rand index trend is also observed to retain when pairwise constraints are increased.

The effect of increasing the weight to the fair ML and CL constraints on the objective function given in equation (4.23) is also measured. The objective function, as defined in equation (4.3), is also computed using the baseline method. Figure 4.5 presents the objective function values. The fair-cSC method achieves a lower objective function value on five out of

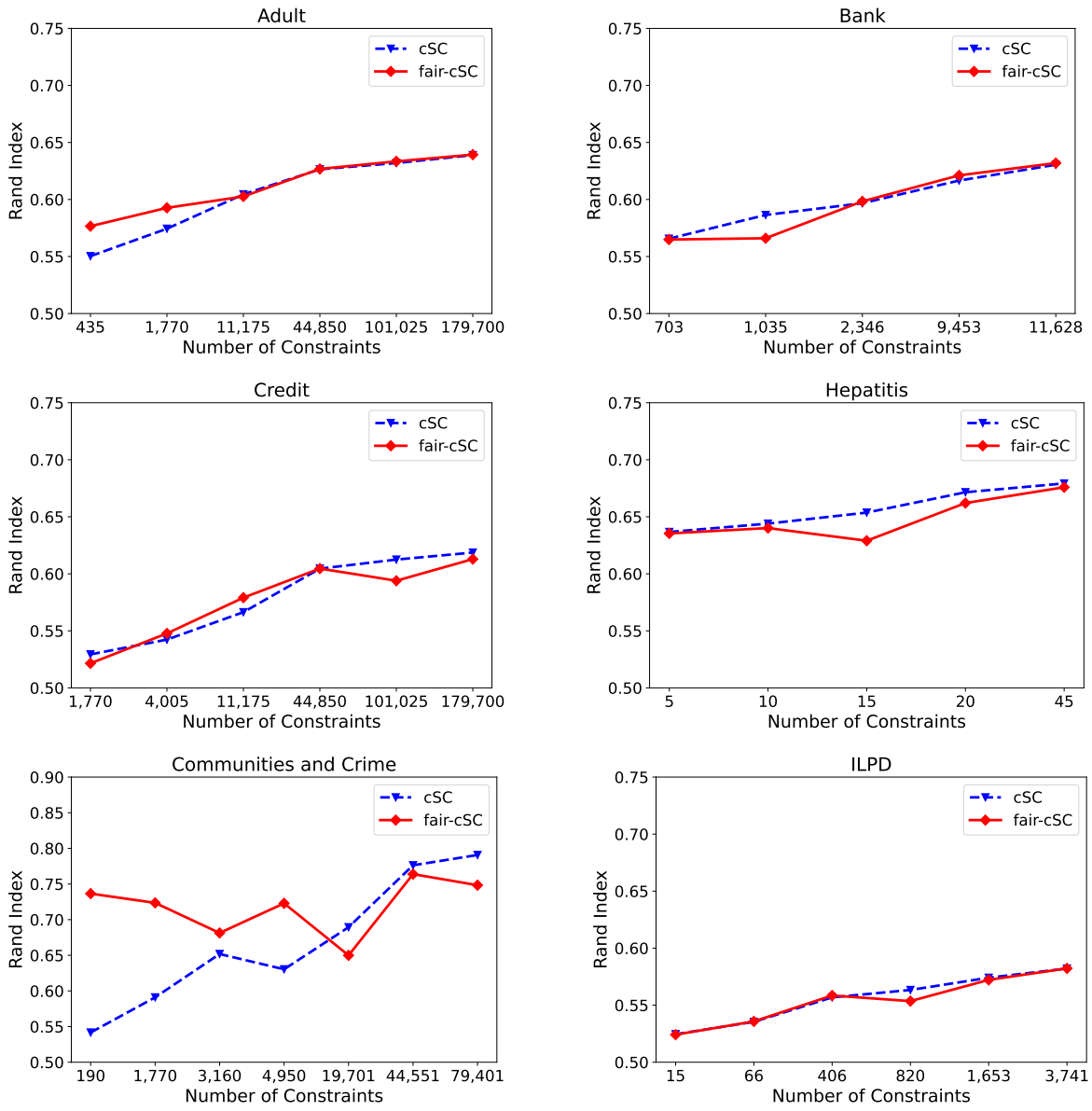


Figure 4.4: *Rand Index of cSC and fair-cSC against the different number of pairwise constraints on all the six datasets with $c = 2$ clusters.*

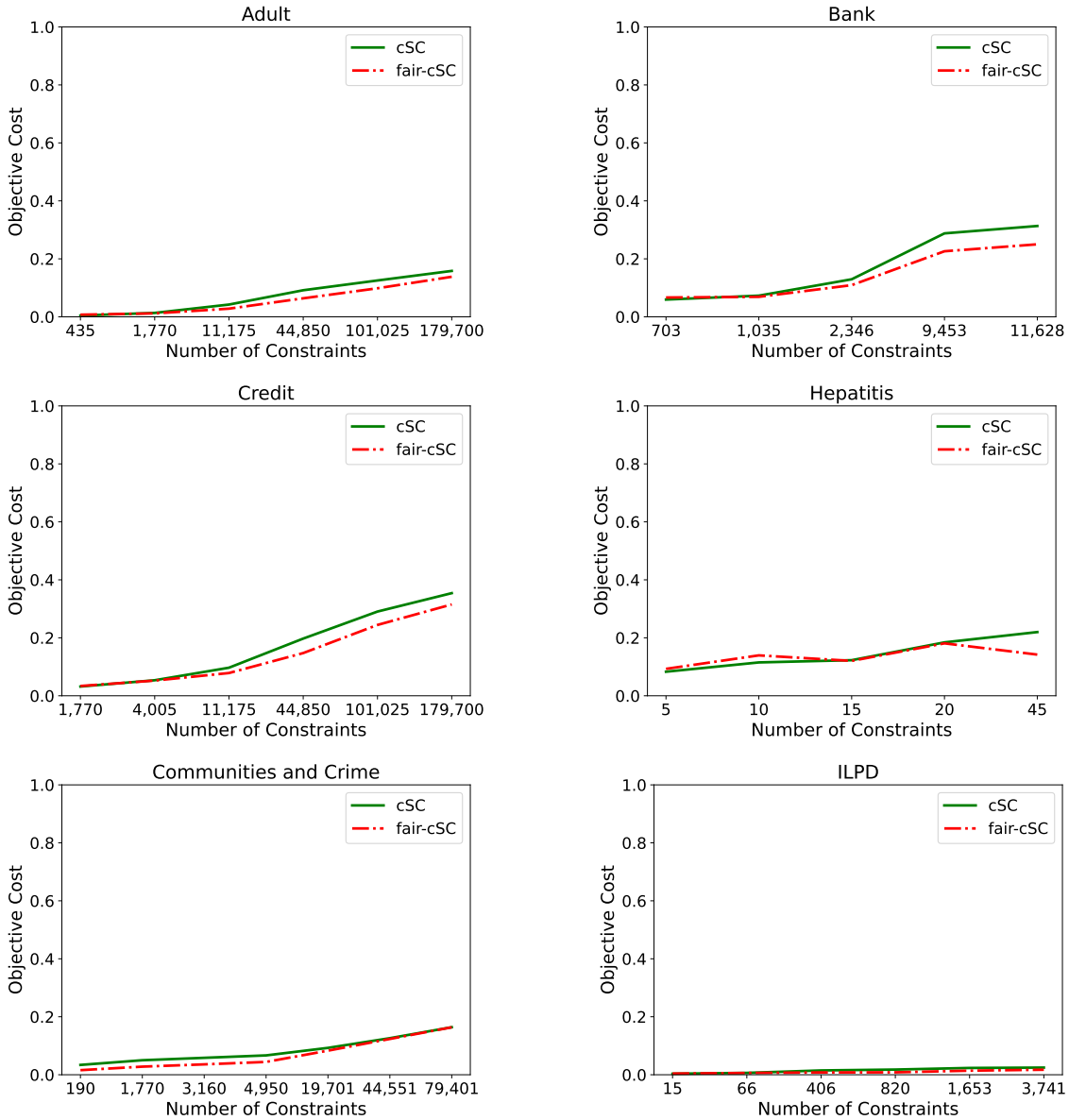


Figure 4.5: *The cost variation over the different number of pairwise constraints for cSC and fair-cSC on all six datasets with $c = 2$ clusters.*

six datasets. By increasing the number of input pairwise constraints, the proposed method retains the trend of achieving lower objective function values. For the Hepatitis dataset, the objective function value of the proposed method increased marginally over the baseline method. These trends indicate that the fair-cSC method achieves fairness without sacrificing the objective function value.

4.1.4.4 Constraints Satisfaction

The number of satisfied constraints determines how well each method adheres to the given pairwise constraints. When a large number of satisfied ML and CL constraints do not consider fairness, they lead to better cluster quality. Conversely, satisfying more fair ML and CL constraints improves balance. The fair-cSC assigns a higher weight to fair ML and CL constraints, as defined in equation (4.11), prioritizing the maximization of satisfying fair constraints. Figure 4.6 shows that fair-cSC consistently satisfies an equal or greater number of pairwise constraints compared to cSC across all datasets. Importantly, the satisfied constraints in fair-cSC are primarily fair ML and CL, reinforcing its ability to ensure fairness in pairwise relationships. This leads to more equitable clustering outcomes, demonstrating the effectiveness of incorporating fairness into constraint satisfaction. Unlike fair-cSC, cSC satisfies constraints without considering fairness, emphasizing the importance of integrating fairness alongside clustering quality.

4.1.4.5 Ablation Studies

Four ablation studies are conducted to analyze the effectiveness of the proposed fair-cSC method. The method is tested under different settings of fair and unfair pairwise constraints as well as varying the values of the regularization factor λ. Overall, the results demonstrate a significant improvement in balance across all cases. A detailed analysis of each study is presented below.

Effect of λ on Balance: Since balance is a crucial metric in assessing fairness, the purpose of this experiment is to observe the impact of varying the λ parameter on the

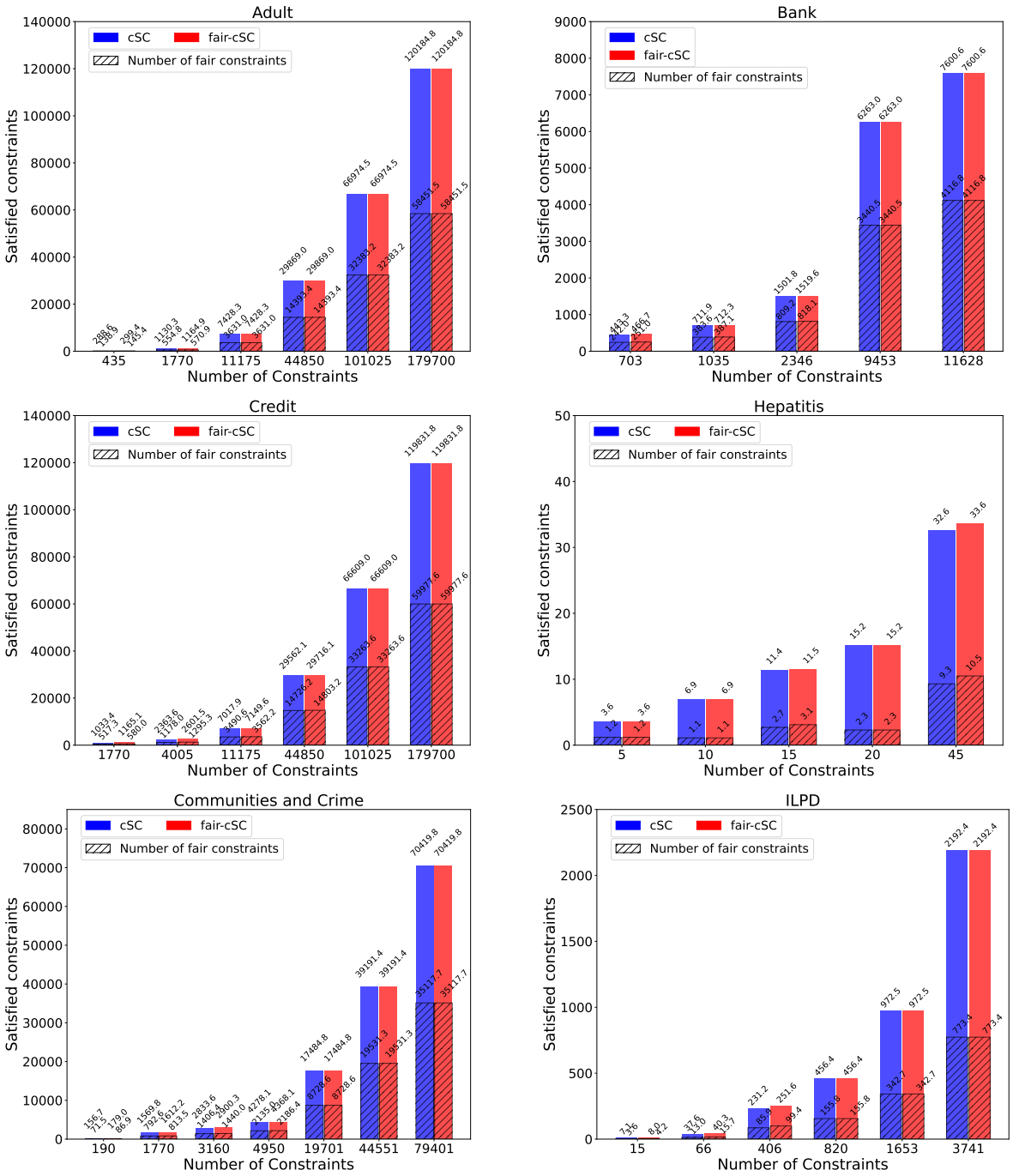


Figure 4.6: *Comparison of total satisfied constraints between cSC and the fair-cSC against different numbers of pairwise constraints on all six datasets with $c = 2$ clusters. The bars with diagonal lines show the number of fair constraints satisfied by these methods.*

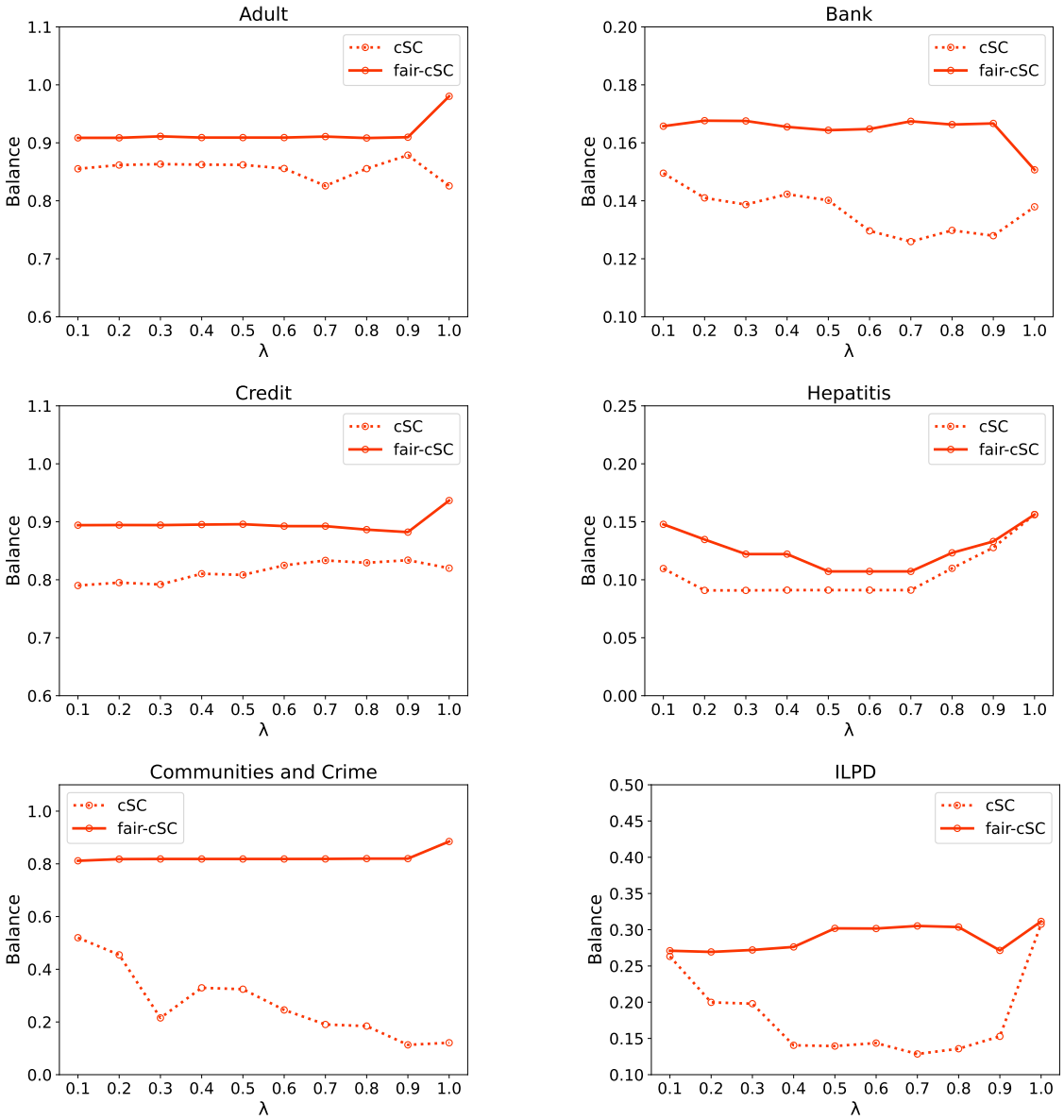


Figure 4.7: *Balance results of fair-cSC and cSC across all six datasets by varying the parameter λ. The number of pairwise constraints is set to 5% of the data points. The plots show the impact of λ on the clustering balance.*

balance value in the proposed fair-cSC and the baseline cSC methods. The number of pairwise constraints is set to 5% of data points. The value of λ is varied from 0.1 to 1 in steps of 0.1. This experiment demonstrates the significance of tuning the λ parameter to obtain optimal balance value in fair-cSC and baseline cSC. A suitable λ improves the clustering performance by balancing the trade-off between pairwise constraints and the clustering objective. Figure 4.7 shows that $\lambda = 0.5$ is most effective across all the datasets, providing a balanced trade-off between SC and pairwise constraints. This guarantees that both objectives are given equal weight, which leads to better clustering performance.

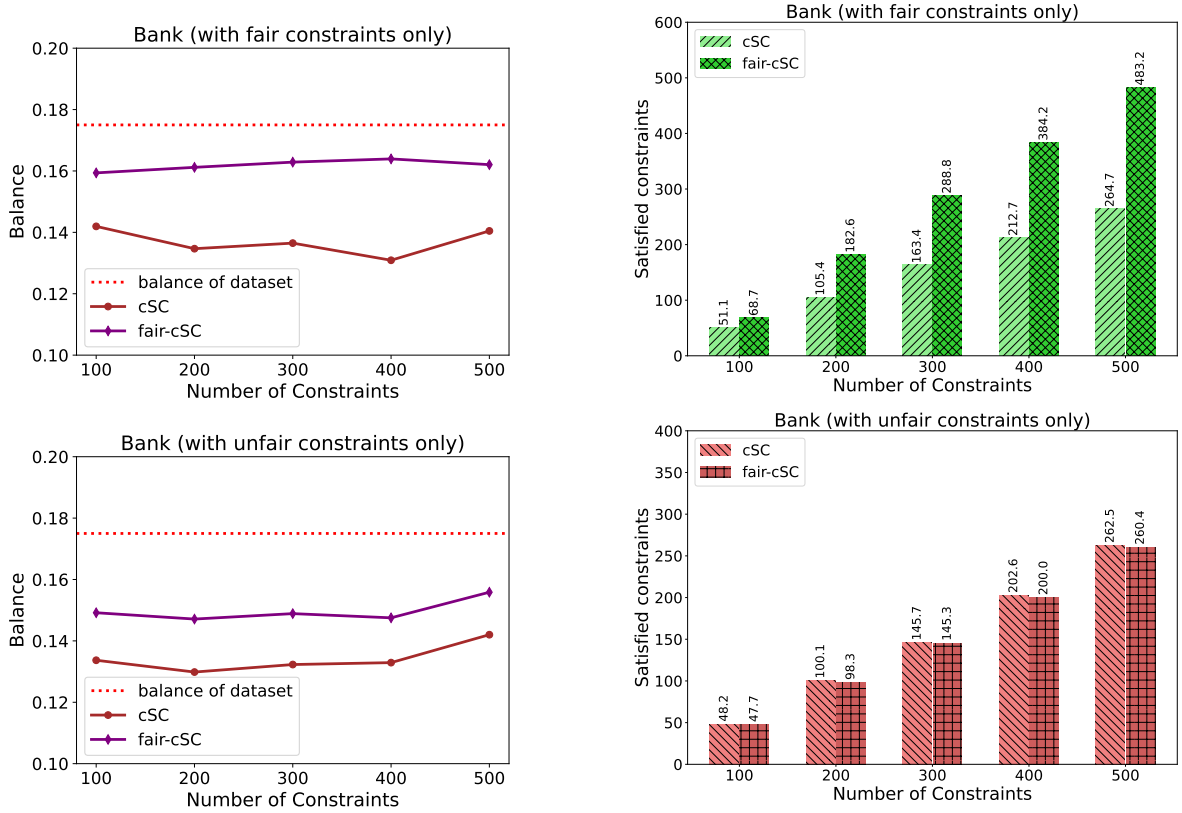


Figure 4.8: *Impact of applying only fair and only unfair pairwise constraints on balance and the number of satisfied constraints for bank dataset. The left side graphs show the effect on balance, and the right side graphs show the number of satisfied constraints. Comparing fair-cSC with cSC to evaluate the maximization of fair constraint satisfaction.*

Impact of Fair vs. Unfair Constraints: The impact of fair and unfair ML and CL constraints on balance is examined using the fair-cSC and baseline cSC. The plots in the first row of Figure 4.8 present the balance by generating all fair ML and CL constraints

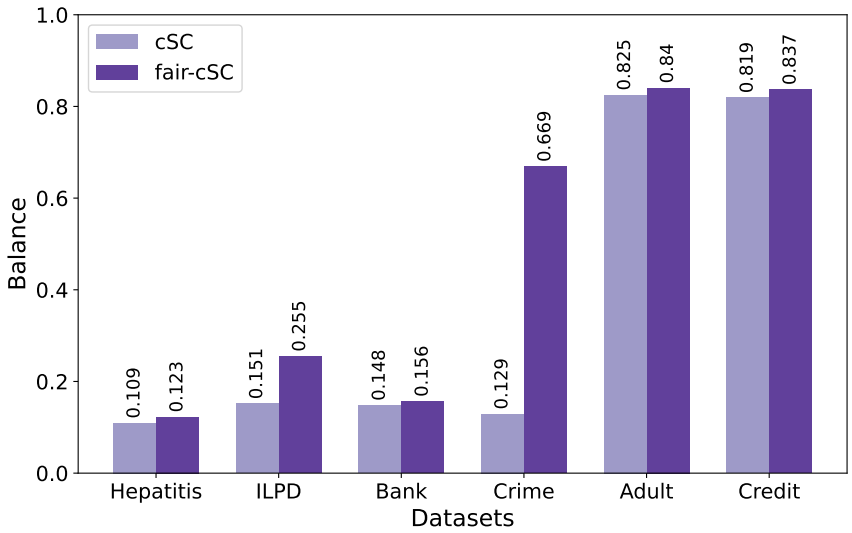


Figure 4.9: *Analysis of fairness (balance value) without considering balance constraint in equation (4.24). The number of pairwise constraints employed is 50 for all the six datasets. The value of λ is set to 0.1 to demonstrate the impact of $\mathbf{Q}_{fair}$ on fair-cSC, and baseline cSC*

on the bank dataset. It is observed that the balance improves consistently in the case of fair-cSC when compared to the baseline cSC. It is also observed that the total number of pairwise constraints satisfied is high in the fair cSC compared to the baseline cSC. This is due to the high weightage given to the fair constraints in the $\mathbf{Q}_{fair}$ matrix.

Also, the plots in the second row of Figure 4.8 present the balance by generating all unfair ML and CL constraints. It is observed that the balance achieved is solely due to the fair SC method, and the number of satisfied constraints is at par with the baseline method.

Influence of the Fair Constraint Matrix: The effect of the fair constraint matrix on balance is analyzed by modifying the fair-cSC objective function. This is done by removing the balance constraint ($\mathbf{J}^T\mathbf{u} = 0$) while keeping the second term unchanged. Figure 4.9 illustrates the effect of $\mathbf{Q}_{fair}$ on balance by solving the following method. This method is the same as in equation (4.21) but without the balance constraint:

$$\lambda \times J_{sc} + (1 - \lambda) \times J_{fair-CM}, \tag{4.24}$$

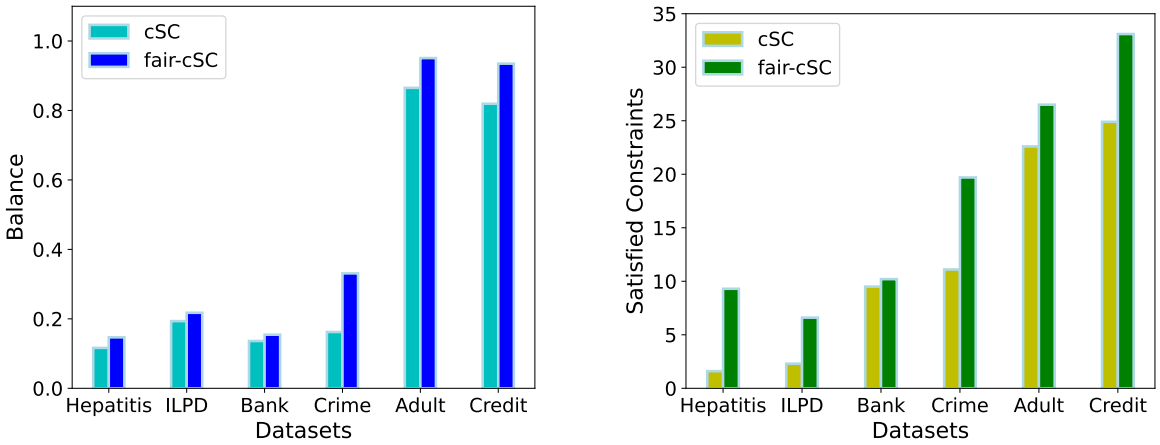


Figure 4.10: *Impact of applying only fair CL constraints on balance (left side) and the number of satisfied constraints (right side) for all the datasets with* $\lambda = 0.5$. *This is conducted with a fixed number of pairwise constraints (i.e., 50).*

where $\lambda = 0.1$. This shows the effect of ensuring fairness on the ML and CL constraint alone and not incorporating the fairness on input data points. Here, the $\mathbf{Q}_{fair}$ matrix shouldering the fairness effect.

Effect of Using Only Fair CL Constraints: This study examines the impact of integrating only fair CL constraints to evaluate the effectiveness of fair-cSC and understand the role of CL constraints. Figure 4.10 shows the effect of fair-cSC on balance and constraint satisfaction by generating only fair CL constraints. Previous works on constrained SC criticized the CL constraints due to their non-transitive property [105], which leads to constraint violations and inconsistency in clustering outcomes. With this concern, the ablation study evaluates how the integration of only fair CL constraints affects the clusters obtained from cSC and fair-cSC across all the datasets. A certain number of CL constraints were randomly generated. The experimental results in Figure 4.10 demonstrate a significant improvement in balance and lower violation for CL constraints than cSC. This suggests that these fair CL constraints maintain an equal proportion of sensitive groups in each cluster. This highlights the effectiveness of the proposed method in maintaining fairness while handling pairwise constraints efficiently. Thus, the proposed method enhances fairness and achieves higher constraint satisfaction, which means the proposed method is more robust and fair. Thus,

the utility of fair CL constraints in achieving fair clusters is validated.

4.2 Constrained Spectral Clustering with Pairwise Fairness

4.2.1 Introduction

This work proposes an approach to achieve an equitable representation of demographic groups in the resulting clusters while performing constrained SC. Firstly, this introduces the Pairwise Constrained Spectral Clustering (PCSC) method. This method ensures that the ML and CL constraints lie in the null space of the cluster indicator matrix while minimizing the NCut objective function. Thereafter, a pairwise fairness definition specific to ML and CL constraints is incorporated in the PCSC to ensure fairness in the presence of ML and CL constraints. The resulting PCSC method, which incorporates fairness considerations, is referred to as Fairness in Pairwise Constrained Spectral Clustering (FPCSC). It is essential to note that the balance notion [25] involves the use of the entire dataset without any external information. In contrast, this work extends it to include ML and CL constraints.

The main contributions of this work can be summarized as:

- This work presents a novel approach for incorporating fairness into constrained SC to ensure equitable representation of demographic groups.
- Pairwise fairness is incorporated into the pairwise constraint set by embedding a linear constraint that considers demographic group information into the SC objective function.
- A comprehensive set of experiments is conducted on real-world datasets, including Hepatitis, Census, and FriendshipNet, to evaluate the performance of the proposed approach in terms of both fairness and clustering quality.

4.2.2 Pairwise Constraints in Spectral Clustering

The standard NCut problem of SC is extended by introducing a linear constraint that fulfills the ML and CL constraints. An algorithm named PCSC is developed. To encode the supervision of ML and CL constraints, the constraint matrix $\mathbf{Q}$ defined in equation (2.11) is used. A constrained optimization task is developed to integrate these constraints into the SC framework by incorporating the constraint matrix $\mathbf{Q}$. By aligning the existing constrained clustering algorithms of [88, 106, 107], A constraint $\mathbf{QH} = 0$ is defined, which forces the solution indicator $\mathbf{H}$ to reside in the feasible subspace defined by this constraint. Adding this constraint, $\mathbf{QH} = 0$ with the NCut optimization of SC in equation (2.4), the optimization problem for PCSC will become:

$$\min_{\mathbf{H}\in\mathbb{R}^{n\times c}} \mathrm{Tr}(\mathbf{H}^T\mathbf{L}\mathbf{H}) \quad \text{s.t.} \quad \mathbf{H}^T\mathbf{D}\mathbf{H} = I_c \quad \text{and} \quad \mathbf{Q}\mathbf{H} = 0_{n\times c}. \tag{4.25}$$

Following [25,30], the optimization problem is solved using the Rayleigh-Ritz theorem by finding a basis in the feasible solution space defined by $\mathbf{QH} = 0$, where $\mathbf{H}$ represents solution matrix. This guarantees that the final partitioning follows the ML and CL constraints when dividing the data into c clusters.

4.2.3 Proposed Methodology

This work presents the fair version of SC in the presence of ML and CL constraints, extending the equation (4.25) by incorporating the balance specific to the ML and CL constraint set. Figure 4.11 presents the outline of the proposed approach. Although these constraints enhance clustering quality, they may adversely affect fairness. Therefore, the approach FPCSC is introduced to ensure the equal representation of binary-valued demographic groups in each cluster. This approach begins by introducing a fair constraint term and subsequently develops a constrained optimization problem to generate clusters that closely satisfy this constraint term.

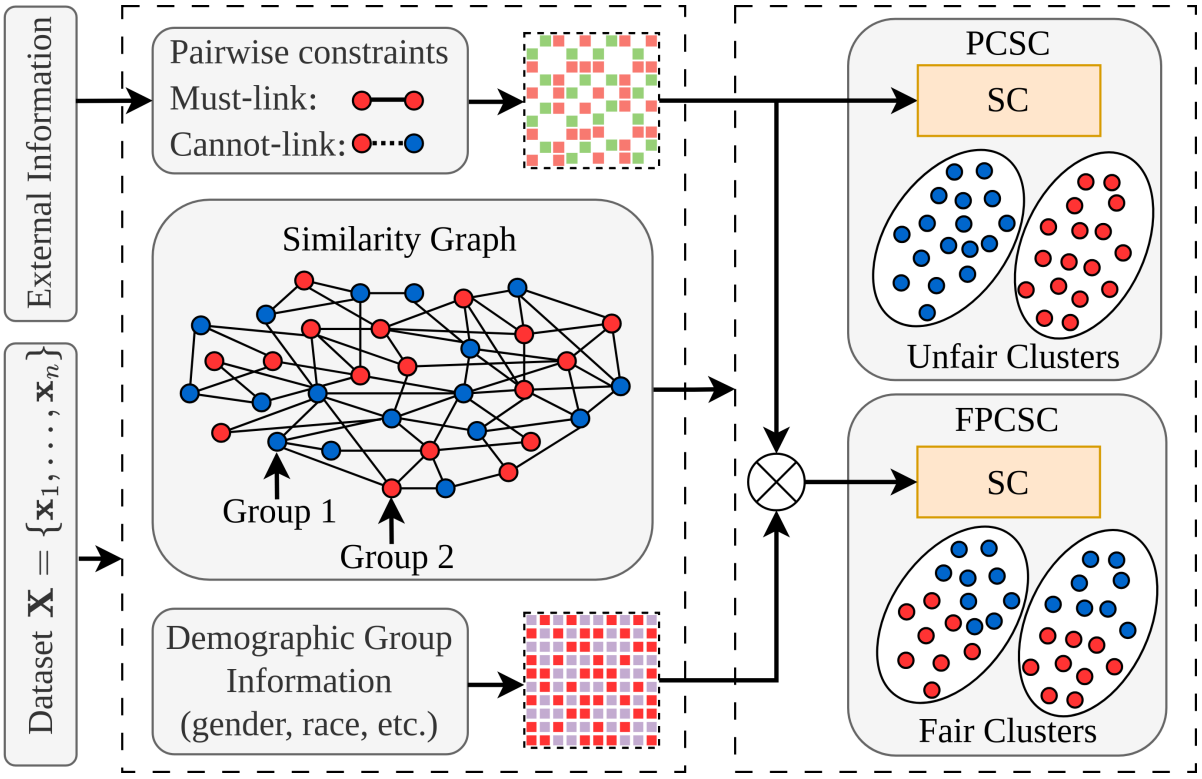


Figure 4.11: *Illustration of PCSC and FPCSC (proposed).*

4.2.3.1 Fair Constraint Term

To achieve fairness in pairwise constraints, the notion of balance in equation (2.2) is considered. The higher the balance value, the more fair the clustering is, as it indicates an adequate representation of demographic groups within each cluster.

Definition 5. *Given $\boldsymbol{s} \in \{1, -1\}^n$ be the demographic membership vector, where $\boldsymbol{s}_i = 1$ if $\mathbf{x}_i \in G_t$ and $\boldsymbol{s}_i = -1$ otherwise, a constraint matrix $\boldsymbol{Q}$ given in equation (2.11), and a cluster indicator matrix $\boldsymbol{H} \in \mathbb{R}^{n \times c}$, the following linear constraint ensures that every cluster has an equal representation of demographic groups.*

$$(\boldsymbol{s}\boldsymbol{s}^T \boldsymbol{Q})\boldsymbol{H} = 0. \tag{4.26}$$

Here, the pairwise constraints in $\mathbf{Q}$ use weighted demographic group information as linear constraint on cluster indicator $\mathbf{H}$. The resulting matrix $\mathbf{s}\mathbf{s}^T\mathbf{Q}$ influences the cluster assignments in $\mathbf{H}$. Therefore, The constraint in equation (4.26) ensures that the clustering

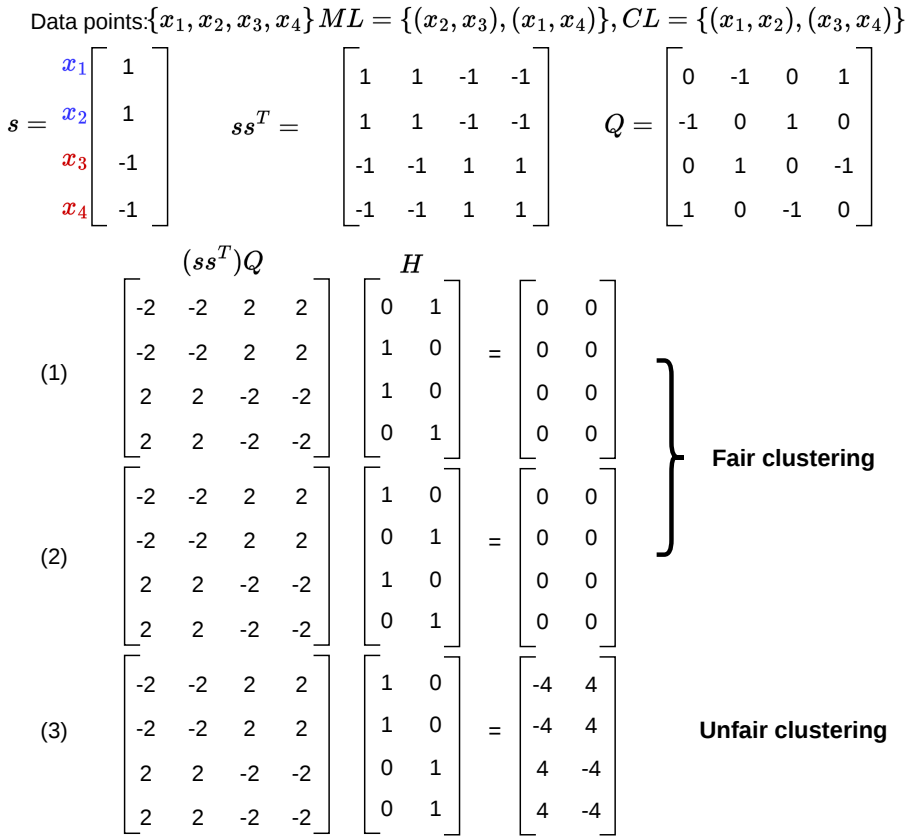


Figure 4.12: *Example illustrating the encoding of sensitive attributes, pairwise constraint matrix, and the resulting fair and unfair cluster assignments in the proposed framework.*

results respect the ML and CL requirements based on the sensitive attribute information encoded in **s**.

4.2.3.2 Illustrative Example of Fair Constraint Term

To provide further intuition on how sensitive attributes are input and utilized within the fairness-aware constrained spectral clustering framework, we consider an example consisting of four data points (x_1, x_2, x_3, x_4) as discussed in Figure 4.12.

- **Input Representation:**
  - **Sensitive Attributes (s):** The sensitive information is input as a group indicator vector **s**. In this example, x_1 and x_2 belong to one demographic group (assigned a value of 1), while x_3 and x_4 belong to a second group (assigned a value of -1) .

  - **Constraint Matrix (Q):** The domain knowledge is encoded in a matrix Q. Here, we define ML relationships for (x_2, x_3) and (x_1, x_4), and CL relationships for (x_1, x_2) and (x_3, x_4).

- **Evaluation of Clustering Fairness:**

  The algorithm evaluates cluster indicator matrices (H) against the fairness and constraint criteria. As shown in Figure 4.12:

  - **Fair Clustering:** In assignment cases (1) and (2), the clustering results in a balanced representation of both demographic groups across the clusters. Mathematically, the interaction between the indicator matrix and the fairness-constraint terms results in a zero-matrix $[0, \ldots, 0]$, indicating that the fairness requirements are satisfied.
  - **Unfair Clustering:** In case (3), the clustering assignment fails to maintain group balance. This misalignment is captured by the optimization term $(\mathbf{s}\mathbf{s}^T)Q$, where the resulting non-zero matrix values signify a violation of the fairness objectives.

4.2.3.3 Fairness in Pairwise Constrained Spectral Clustering

Given the objective of clustering data points, while simultaneously satisfying pairwise constraints and ensuring fairness concerning sensitive attribute information, the constraint defined in equation (4.26) guides the clustering process. It balances the ML and CL constraints, resulting in fair clusters. This constraint is incorporated into the SC problem of equation (2.4) by defining the constrained graph partitioning problem with fairness consideration. Therefore, the FPCSC method is formulated as follows:

$$\min_{\mathbf{H}\in\mathbb{R}^{n\times c}} \operatorname{Tr}(\mathbf{H}^T\mathbf{L}\mathbf{H}) \quad \text{s.t.} \quad \mathbf{H}^T\mathbf{D}\mathbf{H} = \mathbf{I}, \text{ and } (\mathbf{s}\mathbf{s}^T\mathbf{Q})\mathbf{H} = 0. \tag{4.27}$$

The solution approach is analogous to the work of fair SC [25]. The constraint $(\mathbf{s}\mathbf{s}^T\mathbf{Q})\mathbf{H} = 0$ implies that for any feasible matrix $\mathbf{H}$, its column must lie in the null space of $\mathbf{s}\mathbf{s}^T\mathbf{Q}$. Thus,

any feasible $\mathbf{H}$ can be represented as $\mathbf{H} = \mathbf{PY}$, where $\mathbf{Y} \in \mathbb{R}^{(n-r)\times c}$, and $\mathbf{P} \in \mathbb{R}^{n\times(n-r)}$ is an orthonormal basis whose columns span the null space of $\mathbf{ss}^T\mathbf{Q}$. Here, r is the rank of $\mathbf{ss}^T\mathbf{Q}$ and assuming that $c \leq n - r$ otherwise equation (4.27) does not have any solution. The problem in equation (4.27) will become:

$$\min_{\mathbf{Y}\in\mathbb{R}^{(n-r)\times c}} \text{Tr}(\mathbf{Y}^T\mathbf{P}^T\mathbf{LPY}) \quad \text{s.t.} \quad \mathbf{Y}^T\mathbf{P}^T\mathbf{DPY} = \mathbf{I}. \tag{4.28}$$

The square root of $\mathbf{P}^T\mathbf{DP}$ is positive definite, indicating that there exists a positive definite $\mathbf{K} \in \mathbb{R}^{(n-r)\times(n-r)}$ such that $\mathbf{P}^T\mathbf{DP} = \mathbf{K}^2$. By substituting $\mathbf{Y} = \mathbf{K}^{-1}\mathbf{B}$ for $\mathbf{B} \in \mathbb{R}^{(n-r)\times c}$, the following problem is equivalent to equation (4.28):

$$\min_{\mathbf{B}\in\mathbb{R}^{(n-r)\times c}} \text{Tr}(\mathbf{B}^T\mathbf{K}^{-1}\mathbf{P}^T\mathbf{LPK}^{-1}\mathbf{B}) \quad \text{s.t.} \quad \mathbf{B}^T\mathbf{B} = \mathbf{I}. \tag{4.29}$$

The solution to the above optimization problem involves finding the top c eigenvectors of the matrix $\mathbf{K}^{-1}\mathbf{P}^T\mathbf{LPK}^{-1}$, which form the matrix $\mathbf{B}$. The clusters are then identified by applying k-means clustering to the rows of $\mathbf{H} = \mathbf{PK}^{-1}\mathbf{B}$. The procedure is summarized in the Algorithm 3.

Algorithm 3: FPCSC Algorithm

Input : Similarity matrix $\mathbf{S} \in \mathbb{R}^{n\times n}$, clusters $c \in \mathbb{N}$, constraint matrix $\mathbf{Q} \in \mathbb{R}^{n\times n}$, demographic group information

Output: Clusters $\pi_1, \ldots, \pi_c$

1. Let $\mathbf{s}$ be a demographic group membership vector $\mathbf{s} \in \{1, -1\}^n$.
2. Compute the Laplacian matrix $\mathbf{L} = \mathbf{D} - \mathbf{S}$ using degree matrix $\mathbf{D}$.
3. Compute a matrix $\mathbf{P}$ containing orthonormal basis of null space of $(\mathbf{ss}^T\mathbf{Q})$.
4. Compute the square root of $\mathbf{P}^T\mathbf{DP}$ as $\mathbf{K}$.
5. Solve the equation (4.29) by computing c smallest eigenvalues of $\mathbf{K}^{-1}\mathbf{P}^T\mathbf{LPK}^{-1}$ and corresponding eigenvectors ($\mathbf{B}$ is the matrix containing these vectors as columns).
6. Apply k-means clustering to the rows of $\mathbf{H} = \mathbf{PK}^{-1}\mathbf{B}$ to obtain the clusters $\pi_1, \ldots, \pi_c$.

Computational Complexity: With the above implementations, the time complexity of both the methods PCSC and FPCSC is $O(n^3)$, and the space complexity is $O(n^2)$. This is comparable with the worst-case complexity of conventional SC, particularly in the case of

an arbitrary number of clusters. The computationally dominant steps are computing the null space of $\mathbf{s}\mathbf{s}^T\mathbf{Q}$ to calculate $\mathbf{P}$ (which involves singular value decomposition) and finding the eigenvectors.

Theoretical Analysis: The proposed approach ensures that the clustering outcomes adhere to fairness concerns in the presence of pairwise constraints. The product term $\mathbf{s}\mathbf{s}^T\mathbf{Q}$ in equation (4.26) integrates demographic information with the constraint matrix $\mathbf{Q}$, capturing the interaction between pairwise constraints under the influence of demographic groups. This strategy updates the constraint matrix with weighted terms based on the group membership information. When $(\mathbf{s}\mathbf{s}^T\mathbf{Q})\mathbf{H} = 0$ is satisfied, it implies that the interaction matrix $\mathbf{s}\mathbf{s}^T$, weighted by $\mathbf{Q}$, is orthogonal to the cluster indicator matrix $\mathbf{H}$. This ensures that the interaction patterns dictated by demographic group information are preserved within the clusters, leading to a distribution that aligns with the pairwise constraints in $\mathbf{Q}$. As a result, the clusters are more likely to respect ML and CL constraints, promoting equitable representation of demographic groups.

4.2.4 Experiments and Results

The experiments are performed on Hepatitis[1], Census[2], and FriendshipNet[3] datasets. The pairwise constraints are randomly generated using ground truth labels to form the matrix $\boldsymbol{Q}$, and the algorithm's performance is averaged over ten iterations of the constraint generation process. For the FriendshipNet dataset, pairwise constraints are generated randomly due to the absence of ground truth information. Consequently, the Rand Index is not evaluated for this dataset. All algorithms, including baselines, are implemented in Python3. The built-in k-means clustering method is utilized, keeping all parameters at their default values.

[1] https://archive.ics.uci.edu/dataset/46/hepatitis
[2] https://archive.ics.uci.edu/dataset/2/adult
[3] http://www.sociopatterns.org/datasets/high-school-contact-and-friendship-networks/

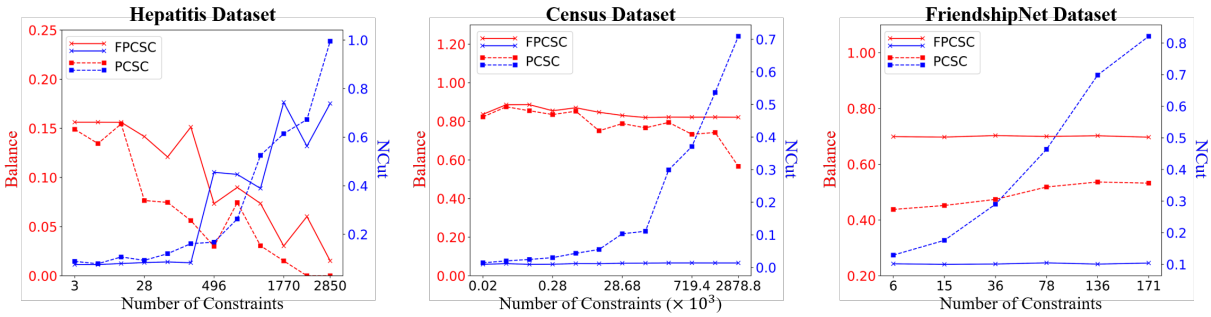


Figure 4.13: *Balance and NCut values for $c = 2$ obtained by varying the number of pairwise constraints using PCSC and FPCSC (proposed) for Hepatitis, Census, and FriendshipNet datasets.*

4.2.4.1 Parameters and Baselines

The proposed approach is examined using two parameters: the number of pairwise constraints and the number of clusters c. The impact of varying the number of pairwise constraints on fairness and clustering quality is analyzed by increasing these constraints to evaluate how effectively the proposed approach ensures fairness while respecting the pairwise constraints. In addition, the number of clusters c varied to analyze the impact of FPCSC.

To benchmark the performance, PCSC is used as the baseline model to evaluate how the FPCSC model performs, as it is the fundamental algorithm on which the fair version is built. To further evaluate the fairness aspect of the proposed approach, a comparison is made with two additional baselines: standard SC and fair SC [25].

4.2.4.2 Fairness and Clustering Performance

The performance of PCSC vs FPCSC is evaluated against the varying pairwise constraints for $c = 2$ Hepatitis, Census, and FriendshipNet datasets as illustrated in Figure 4.13. The fairness of the clustering is measured through balance (shown on the left axis of the plot), and the performance of the clustering is measured through the NCut value (shown on the right axis of the plot). The results demonstrate that the FPCSC performs better in terms of fairness and NCut than the PCSC. This enhancement is evident in the fact that even with the increasing number of pairwise constraints, FPCSC produces well-balanced clusters.

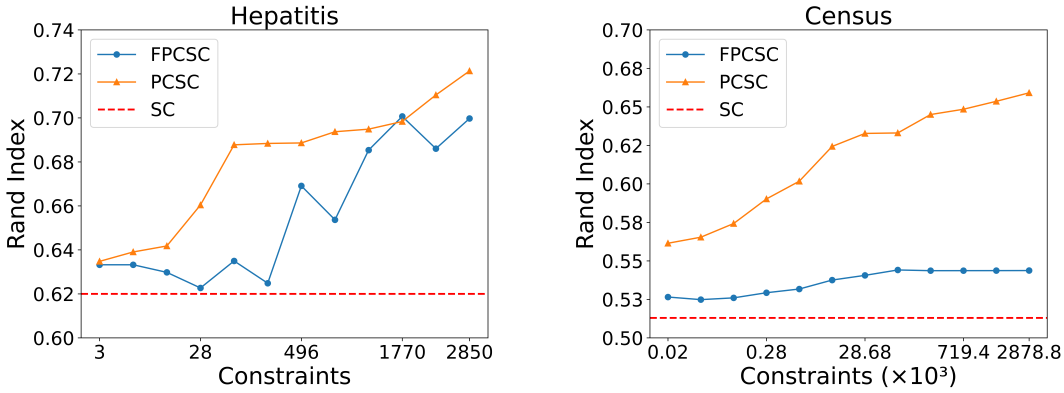


Figure 4.14: *Rand Index comparison for standard SC, PCSC, and FPCSC by varying the number of pairwise constraints.*

However, the PCSC struggles with balance due to its rigid focus on satisfying the constraints, which can come at the cost of fair clusters.

In contrast, the FPCSC method generates fair clusters and maintains minimal inter-cluster similarity. In the Hepatitis dataset, the relative improvement of 169.42% is observed with 30% pairwise constraints, the Census dataset shows a 44.92% improvement with 80% pairwise constraints, and the FriendshipNet dataset achieves an improvement of 59.66% with 3% pairwise constraints. Moreover, in terms of NCut, FPCSC maintains a well-controlled NCut value. As the number of pairwise constraints increases, The NCut value rises for PCSC. However, a low NCut value for FPCSC is observed, except for the Hepatitis dataset. The Hepatitis dataset has no clear increasing or decreasing trend, which can be attributed to the nature of pairwise constraints enforced during clustering. Thus, Figure 4.13 show that the FPCSC consistently produces clusters that achieve lower NCut values and maintain higher balance, even with increasing pairwise constraints.

4.2.4.3 Evaluation Against Clustering Quality

Figure 4.14 illustrates the clustering quality using the RI and compares the PCSC with SC. Since the results of SC are independent of pairwise constraints, it is drawn as a horizontal line in Figure 4.14. PCSC consistently improves the RI for every set of pairwise constraints, validating its ability to enhance clustering quality by satisfying the maximum possible constraints. However, it provides better clustering quality than SC; it does not in-

Table 4.3: *Comparison of balance metric for various datasets.*

Dataset→ Method ↓	Hepatitis ($c = 2$)	Census ($c = 6$)	FriendshipNet ($c = 9$)
SC	0.1	0.762	0.391
fair SC	**0.156**	0.777	0.485
PCSC	0.134	0.693	0.374
FPCSC (proposed)	**0.156**	**0.804**	**0.541**

[a]The best values in each column are denoted with boldface.

herently guarantee fairness. In contrast, the fair version of this algorithm, FPCSC, focuses on the equal representation of demographic groups, leading to a slight decrease in the RI. Therefore, the RI of FPCSC is lower than that of PCSC, which can be seen as the price of fairness. It reflects the trade-off between fairness and clustering quality. Despite this, FPCSC still achieves an improved clustering quality in terms of RI compared to standard SC while ensuring fairness.

4.2.4.4 Comparison Experiments

The experimental results in Table. 4.3 compare the balance value of PCSC and FPCSC with SC and fair SC [25], keeping a fixed set of pairwise constraints (5% of data). The number of clusters c is set to $2, 6$, and 9 for the Hepatitis, Census, and FriendshipNet datasets. Fair SC provides a fair variant of the SC algorithm, although both algorithms are independent of pairwise constraints. The balance value for PCSC reduces, indicating that incorporating pairwise constraints sometimes leads to a lower balance. However, FPCSC integrates fairness with pairwise constraints and outperforms all the other algorithms. FPCSC achieves a 16.41% relative improvement in balance over PCSC in the Hepatitis dataset, a 16.01% improvement in the Census dataset, and a 44.65% improvement in the FriendshipNet dataset.

4.2.4.5 Analysis

Varying Number of Clusters: To further study the effect of cluster choices on the proposed approach, the number of clusters c varies. The balance and NCut values are computed to observe the fairness and clustering performance while keeping a fixed number

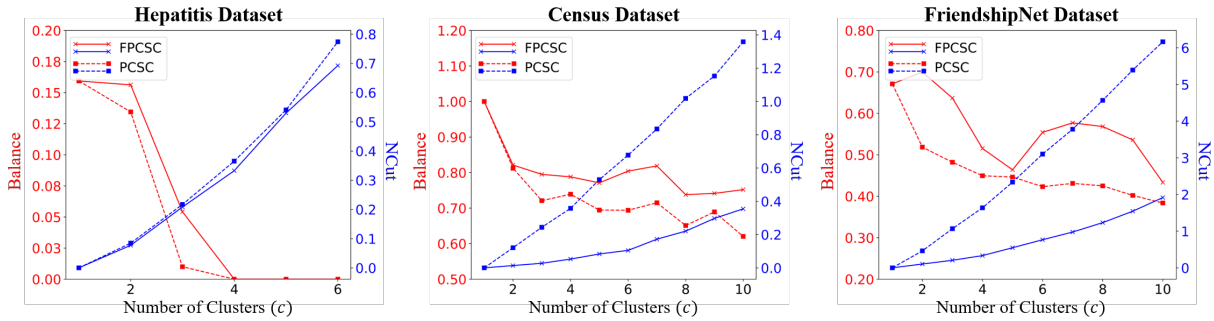


Figure 4.15: *Balance achieved by PCSC and the proposed FPCSC across varying numbers of clusters c , with a fixed number of pairwise constraints (10% of the data), evaluated on the Hepatitis, Census, and FriendshipNet datasets.*

of pairwise constraints (10% of data). Figure 4.15 shows how the variation of c affects the balance and NCut value. It is observed that with an increasing number of c, the balance of PCSC decreases, while the NCut value shows an upward trend.

In contrast, the FPCSC algorithm exhibits a different behavior. Although the NCut value also rises with c, the rate of increase is notably slower than that observed in PCSC. More importantly, the balance of FPCSC is improved with an increasing number of clusters compared to PCSC. This means FPCSC consistently adheres to balance criteria. This consistent achievement of balance highlights the FPCSC's ability to ensure equitable distribution of demographic groups across two or more clusters.

Quality Vs. Fairness: The trade-off achieved by the FPCSC algorithm with respect to RI and fairness is demonstrated. As shown in Figures 4.13 and 4.14, the plots highlight how the FPCSC method navigates these trade-offs. For PCSC, when pairwise constraints increase, RI increases, and the balance value decreases. In contrast, the FPCSC improves the balance by partitioning the clusters equitably and achieves a trade-off between clustering quality and fairness. Specifically, FPCSC may not always reach the same high RI level as PCSC, but it ensures that the obtained cluster adheres more closely to fairness criteria.

4.3 Summary

This chapter presented two contributions toward integrating fairness into constrained SC. In the first part of this chapter, a fair-cSC framework is proposed by incorporating fair ML and CL constraints through the construction of a fairness constraint matrix. This matrix is integrated into the clustering process by modifying the graph Laplacian of SC. This formulation ensures fairness both at the level of individual data points and within the constraint sets. Experimental evaluations on six benchmark datasets demonstrate that fair-cSC achieves substantially improved cluster balance while maintaining competitive clustering quality in terms of RI and reducing the objective function value. Overall, the method consistently outperforms its baseline cSC across fairness and quality metrics.

In the second part of this chapter, a novel FPCSC approach is proposed to consider fairness in constrained SC with pairwise constraints. Unlike the first contribution, which enforces fairness by modifying the graph Laplacian, this work integrates a linear constraint that considers demographic group information alongside the pairwise constraints into the SC objective. This formulation ensures equitable representation of demographic groups in each cluster while simultaneously satisfying pairwise constraints. The impact of pairwise constraints on FPCSC is analyzed by evaluating the RI, balance, and NCut. The experimental results demonstrate that the FPCSC generates balanced clusters with varying pairwise constraints, outperforming SC, PCSC, and fair SC methods.

Together, these two methods demonstrate that fairness can be successfully incorporated into the optimization stage of spectral clustering. However, both approaches adopt in-processing strategies, overlooking the fact that bias present in the data is transferred directly into the similarity graph before any optimization begins. This results in an inherently unfair graph structure and limits the effectiveness of downstream fairness mechanisms. This limitation highlights the need to integrate fairness considerations earlier in the graph-construction stage to mitigate biases before clustering. This raises the question of whether fairness and constraint compliance can be encouraged even earlier in the pipeline, specifically, at the graph construction stage, before the spectral embedding is computed. Therefore, the next

contributory chapter introduces a graph-level preprocessing technique that embeds fairness and constraint information directly into the similarity graph. This approach provides an alternative pathway for achieving fair clustering while preserving clustering quality, without modifying the optimization objective.

"We should not give up and we should not allow the problem to defeat us."

~A P J Abdul Kalam

5

Fairness and Constraint Integration at Graph-Level

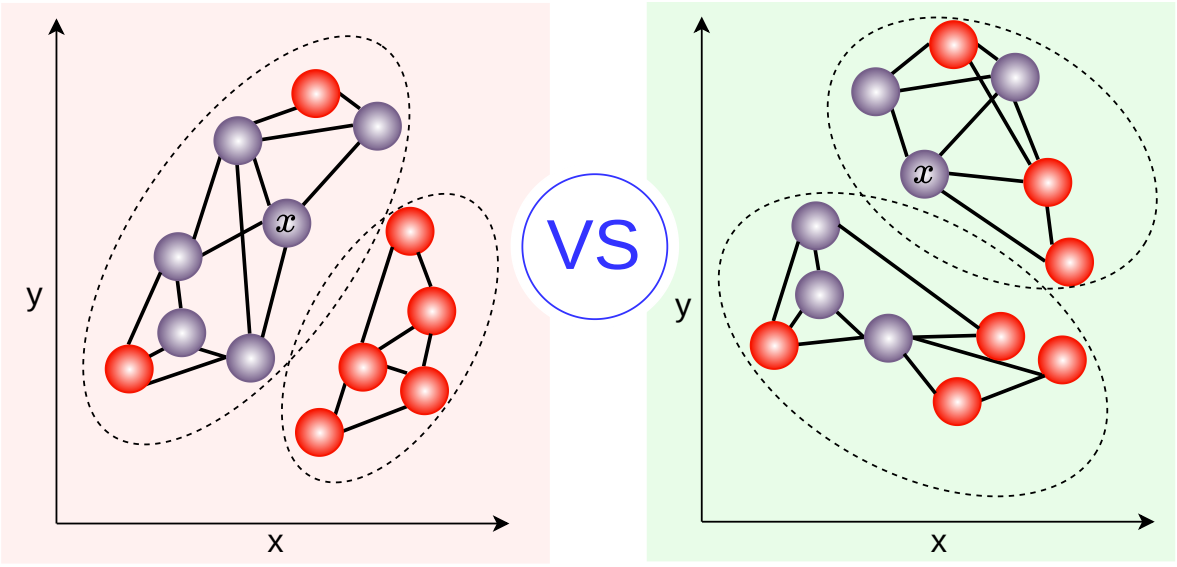
This chapter presents the second contribution, which focuses on a graph-level preprocessing technique that embeds fairness and pairwise constraints directly into the similarity graph before spectral decomposition. Instead of modifying the objective function, the proposed approach adjusts the graph structure itself by incorporating fairness-aware connections and reweighting edges to enforce constraint consistency. As a result, the constructed Laplacian inherently reflects fair and constraint-aligned relationships.

5.1 Introduction

While the previous contributions mark an important step toward incorporating fairness into constrained SC, they operate mainly at the constraint or objective function levels. Consequently, any biases already encoded in the similarity graph remain unaddressed, and the fairness–quality trade-off persists. This limitation highlights the need to integrate fairness considerations earlier in the graph construction process to mitigate biases before clustering commences. Embedding fairness at this stage, along with supervision from ML and CL constraints, enables the representation of more balanced and meaningful relationships among data points. This preprocessing step is therefore crucial to ensure that fairness is embedded in the data representation itself, enabling subsequent clustering to achieve equitable partitions while preserving clustering quality in a principled manner.

In this work, this challenge is addressed by proposing a novel pre-processing framework for fairness-aware constrained SC, in which fairness and pairwise constraints are embedded directly into the graph construction stage. The performance of SC depends heavily on the quality of the similarity graph. The k-Nearest Neighbor (kNN) graph is the most common choice because it effectively captures local neighborhood structures. In this work, we enhance this approach by building a fair kNN graph.

This graph enforces group balance at the node level by extending the disparate impact concept, formulated in supervised learning contexts [12]. In this work, this notion is adapted to the unsupervised setting of graph construction, ensuring that the neighborhood of each node includes a minimum proportion of nodes from groups different from its own. This adaptation promotes local fairness in the similarity structure, ensuring node neighborhoods are not biased toward any particular group. To illustrate the intuition behind our approach, Figure 5.1 presents how the local neighborhood composition of a node directly influences fairness in clustering: when a single sensitive group dominates the neighborhood of a node, the imbalance propagates into the cluster, whereas neighborhoods with diverse groups yield more balanced and fair clusters. At the same time, ML and CL constraints are integrated into the fair graph in a manner that preserves fairness. When adding an ML constraint to



(a) Unfair Clusters (b) Fair Clusters

Figure 5.1: *Impact of node x's neighborhood on clustering fairness. A homogeneous neighborhood dominated by one group reinforces unfairness, whereas a heterogeneous neighborhood leads to more balanced clusters.*

a node, the farthest node belonging to the same sensitive group as the node being linked through the ML constraint, is dropped to maintain k-regularity and neighborhood balance. Conversely, for a CL constraint on a node, the existing connection between the two nodes is removed, and the next-nearest neighbor belonging to the same sensitive group as that node is added to restore diversity. This adaptive integration leverages domain knowledge while upholding fairness guarantees and mitigating bias early in the graph construction stage before clustering occurs. The resulting framework, termed Fair Constrained k-Nearest Neighbor (FC-kNN) for SC, produces fair and structurally coherent clusters. The main contributions of this chapter are summarized below:

- This is the first study on SC that includes a preprocessing method to incorporate fairness and pairwise constraints at the same time.
- A novel method is proposed to construct a fair kNN graph to ensure group balance for each node's neighborhood.
- This graph is then extended to incorporate ML and CL constraints, ensuring both

fairness and domain knowledge are respected simultaneously.

- Developed a SC framework using FC-kNN, that jointly applied fairness and pairwise constraints at the graph construction stage to obtain fair and quality clusters.
- Empirical evaluation demonstrates that the proposed method enhances fairness and clustering quality, effectively overcoming the trade-off observed in prior works.

The rest of the chapter is structured as follows. Section 5.2 and Section 5.3 describe the construction of the standard kNN graph and the Constrained k-Nearest Neighbor (C-kNN) graph, respectively. Section 5.4.1 presents the proposed FC-kNN approach. Section 5.4.2 evaluates the effectiveness of the proposed method in comparison with its unfair baseline using real-world datasets. Finally, Section 5.5 summarizes the chapter.

5.2 Standard kNN Graph

A kNN graph is a building block in clustering algorithms. Let $X = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\}$ be a dataset of n real-valued vectors, where each $\mathbf{x}_i \in \mathbb{R}^d$. For a given query point $\mathbf{x}_i$, the kNN search is performed to find k points closest to $\mathbf{x}_i$ with respect to a distance metric $d(\mathbf{x}_i, \mathbf{x}_j)$ that measures the distance between $\mathbf{x}_j \in X$ and the query point $\mathbf{x}_i$. Formally, the kNN set of $\mathbf{x}_i$ is $\mathcal{N}_k(\mathbf{x}_i) = \arg\min_{R \subseteq X, |R|=k} \sum_{\mathbf{x}_j \in X} d(\mathbf{x}_i, \mathbf{x}_j)$.

The kNN graph $G(V, E)$ is constructed using the neighborhoods obtained from the kNN, search. Each vertex $v_i \in V$ corresponds to a data point $\mathbf{x}_i \in X$. An edge $(v_i, v_j) \in E$ is added if $\mathbf{x}_j$ belongs to the k-nearest neighbor set of $\mathbf{x}_i$ as obtained from kNN search. Formally, $(v_i, v_j) \in E \Leftrightarrow \mathbf{x}_j \in \mathcal{N}_k(\mathbf{x}_i)$. Edges are also assigned weights to reflect similarity between connected points. A common choice is the Gaussian kernel:

$$w_{ij} = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right), \tag{5.1}$$

where $\sigma > 0$ is a scale parameter. The resulting adjacency matrix S then encodes these neighborhood relationships. The kNN graph is widely used in clustering and graph-based

learning methods, such as SC, since it preserves local neighborhood structure while remaining computationally efficient and sparse. Once the similarity graph is constructed, the SC framework involves computing the graph Laplacian, deriving its eigenvectors to embed the data into a low-dimensional space, which is then partitioned into clusters.

5.3 Constrained kNN Graph

Several studies [35–37] have developed methods to directly integrate ML and CL constraints during the graph construction phase. However, in conventional kNN graphs, the edge connections are determined solely based on pairwise distance, without considering any prior knowledge about which samples should or should not be connected. The proposed C-kNN graph construction explicitly honors these ML and CL constraints during the kNN formation stage. Each node's neighborhood is adaptively modified such that the resulting graph simultaneously respects domain knowledge and preserves the k-regularity structure.

Definition 6 (k-Regularity)**.** *A C-kNN graph G satisfies the* k-regularity *if the degree of every vertex $v_i \in V$ remains equal or close to k, i.e.,*

$$|\mathcal{N}_k(v_i)| = k \pm \epsilon, \qquad \forall v_i \in V,$$

where $\epsilon \ll k$ is a small deviation introduced by the enforcement of ML and CL constraints. These constraints affect the neighborhood structure according to:

$$\forall (v_i, v_j) \in ML \implies v_j \in \mathcal{N}_k(v_i), \quad \forall (v_i, v_j) \in CL \implies v_j \notin \mathcal{N}_k(v_i).$$

Thus, while ML constraints require certain vertices to be included in the neighborhood and CL constraints prohibit specific vertices from being included, the overall degree of each vertex remains close to k, yielding a near-uniform local neighborhood structure.

Maintaining k-regularity prevents severe degree imbalance and preserves the spectral characteristics of the graph, ensuring that pairwise constraint integration does not distort

its underlying structure. The key innovation is to reshape each local neighborhood in a constraint-aware manner rather than merely adding or removing edges. This preserves the k-regularity while embedding the required pairwise constraints. The process is executed in three main stages:

1. **Initial neighborhood identification:** A baseline graph structure is first established by performing a kNN search under a similarity metric $d(.,.)$. for each data point. In this initial step, every node is connected to its k closest points in the feature space, capturing the local geometry that serves as the foundation for later constraint refinement. For each node $\mathbf{x}_i$, its neighborhood is:

$$\mathcal{N}_k(\mathbf{x}_i) = \{\mathbf{x}_{i_1}, \ldots, \mathbf{x}_{i_k} \mid d(\mathbf{x}_i, \mathbf{x}_{i_1}) \leq \cdots \leq d(\mathbf{x}_i, \mathbf{x}_{i_k})\}.$$

2. **Constraint-Driven Neighborhood Correction:** Once the baseline graph is built, it is systematically refined to respect the ML and CL constraints:

   - **ML Constraint Enforcement:** For each $(\mathbf{x}_i, \mathbf{x}_j) \in \text{ML}$, the algorithm ensures a strong link exists between them. If $\mathbf{x}_j \notin \mathcal{N}_k(\mathbf{x}_i))$, the neighborhood of $\mathbf{x}_i$ is updated by adding $\mathbf{x}_j$ to $\mathcal{N}_k(\mathbf{x}_i)$ and removing the farthest existing neighbor to preserve the k-regular structure. The new constrained neighborhood $\mathcal{N}_k^C(\mathbf{x}_i)$ is formed by:

$$\begin{aligned} \mathcal{N}_k^C(\mathbf{x}_i) &= \big(\mathcal{N}_k(\mathbf{x}_i) \setminus \{\mathbf{x}_p\}\big) \cup \{\mathbf{x}_j\}, \\ \text{where } \mathbf{x}_p &= \underset{\mathbf{x}_m \in \mathcal{N}_k(\mathbf{x}_i)}{\operatorname{argmax}} \; d(\mathbf{x}_i, \mathbf{x}_m). \end{aligned} \tag{5.2}$$

     This operation replaces the farthest neighbor, $\mathbf{x}_p$, with the required neighbor $\mathbf{x}_j$, ensuring that the ML relationship is explicitly embedded in the local neighborhood structure.

   - **CL Constraint Enforcement:** For each $(\mathbf{x}_i, \mathbf{x}_j) \in \text{CL}$, the algorithm enforces the constraint by pruning any direct connection between $\mathbf{x}_i$ and $\mathbf{x}_j$. If such an edge exists ($i.e., \mathbf{x}_j \in \mathcal{N}_k(\mathbf{x}_i)$), it is removed. To preserve the k-regular structure

of the graph, the removed edge is immediately replaced. This is achieved by performing a secondary search to find the new neighbor $\mathbf{x}_{new}$, the next closest point to $\mathbf{x}_i$ that is not already in its neighborhood. The updated neighborhood $\mathcal{N}_k^C(\mathbf{x}_i)$ is formed by:

$$\begin{aligned} \mathcal{N}_k^C(\mathbf{x}_i) &= \big(\mathcal{N}_k(\mathbf{x}_i) \setminus \{\mathbf{x}_j\}\big) \cup \{\mathbf{x}_{\text{new}}\}, \\ \text{where } \mathbf{x}_{\text{new}} &= \underset{\mathbf{x}_m \in X \setminus \left(\mathcal{N}_k(\mathbf{x}_i) \cup \{\mathbf{x}_i, \mathbf{x}_j\}\right)}{\operatorname{argmin}} d(\mathbf{x}_i, \mathbf{x}_m) \end{aligned} \tag{5.3}$$

This process removes the CL connection to $\mathbf{x}_j$ and repairs the structure with the next-best alternative $\mathbf{x}_{new}$ from the ordered list of neighbors. This maintains consistency with the kNN property and ensures that the local structure reflects both geometric proximity and the imposed constraint.

3. **Final Affinity Matrix Formulation:** Upon completion of all neighborhood corrections, the final neighborhoods represent a structure that encodes both the local manifold of the data and the pairwise constraints. From this, the final constrained affinity matrix $\mathbf{S}_C$ is build as given in following equation 5.4:

$$\mathbf{S}_C(i,j) = \begin{cases} 1, & (\mathbf{x}_i, \mathbf{x}_j) \in \text{ML}, \\ 0, & (\mathbf{x}_i, \mathbf{x}_j) \in \text{CL}, \\ w_{ij}, & \text{as in equation (5.1)}. \end{cases} \tag{5.4}$$

The matrix $\mathbf{S}_C$ is subsequently used to compute the graph Laplacian, and the rest of the SC algorithm proceeds as standard.

5.4 Fair Constrained kNN

This section presents the proposed method for fair neighborhood rewiring and constraint integration, along with the corresponding algorithms.

5.4.1 Proposed Methodology

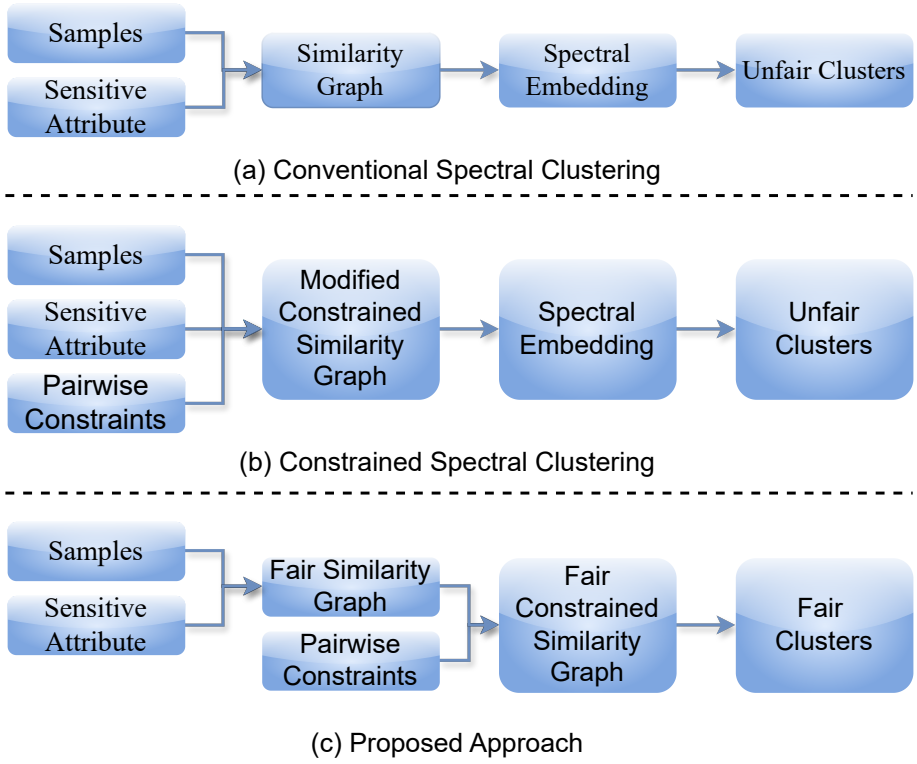


Figure 5.2: *Comparative diagram showing the progression from conventional to constrained and fair constrained SC. The proposed approach integrates a fair graph construction with pairwise constraints to ensure balanced and consistent clustering.*

The work introduces a novel FC-kNN, a graph construction strategy that unifies fairness and pairwise constraints. It is specifically designed as a pre-processing step for SC. The method addresses the limitations of the standard kNN graph, where neighborhood formation can disadvantage sensitive groups, and imposing pairwise constraints can further disrupt group balance. Unlike prior studies that prioritize one aspect at the expense of the other, FC-kNN addresses both issues in one pipeline. To provide an overview of how the proposed framework differs from conventional approaches, Figure 5.2 presents a comparative diagram of conventional SC, constrained SC, and the proposed fair constrained SC. This highlights how the proposed FC-kNN extends existing methods by embedding fairness into the constraint-based clustering paradigm.

The process begins with constructing a standard kNN graph G from the dataset X. Next, it enforces a neighborhood-level disparate-impact requirement, ensuring that each

node's neighbors include at least a minimum proportion of nodes from groups different from its own. This adjustment produces a fair-kNN graph with demographically balanced local neighborhoods. Thereafter, the fair-kNN graph is expanded using the given ML and CL constraints: ML edges are added to strengthen similarity, while CL edges are removed to reflect dissimilarity, ensuring that domain knowledge is respected without violating the fairness condition. The outcome is the final affinity matrix S_{FC}, which simultaneously encodes demographic fairness and domain constraints. This matrix serves as the input to SC.

Figure 5.3 illustrates an example of the proposed FC-kNN framework, showing how fairness-aware neighborhood construction and constraint propagation are carried out to form the balanced neighborhood structure. Part (I) demonstrates the construction of a fair neighborhood in a step-wise manner. Initially, the top-k neighbors of a node x are selected based on distance ranking. In the first step of initial kNN selection, all four neighbors (k=4) belong to the same sensitive group (Group 1) i.e., $\mathcal{N}_4$(x)={a,b,c,d}, resulting in an unfair neighborhood. In the second step, a fair kNN adjustment is performed to improve group balance. The farthest neighbor d from Group 1 is replaced with the next-neraest candidate f from the different group (Group 2), yielding $\mathcal{N}_4$(x)={a,b,c,f}. However, the neighborhood still lacks fairness. Therefore, a further fair kNN adjustment is applied by replacing the farthest remaining neighbor c from Group 1 with the next-nearest node g from Group 2. This results in the updated neighborhood $\mathcal{N}_4$(x)={a,b,f,g}, which achieves a fair representation of the sensitive groups. Part (II) shows the pairwise constraint propagation step, where ML and CL constraints are integrated into the fair neighborhood. For an ML constraint between (x,p), node p (belonging to Group 2) is added as a neighbor of x. To maintain both k-regularity and neighborhood balance, the farthest neighbor from the same group as p (Group 2), namely g is dropped from the neighborhood. This adjustment results in the updated neighborhood $\mathcal{N}_4$(x)={a,b,f,p}. Similarly, for a CL constraint between (x,b), the connection to node b (belonging to Group 1) is dropped and replaced with the next-nearest candidate c from the same group as b (Group 1). This adjustment results in

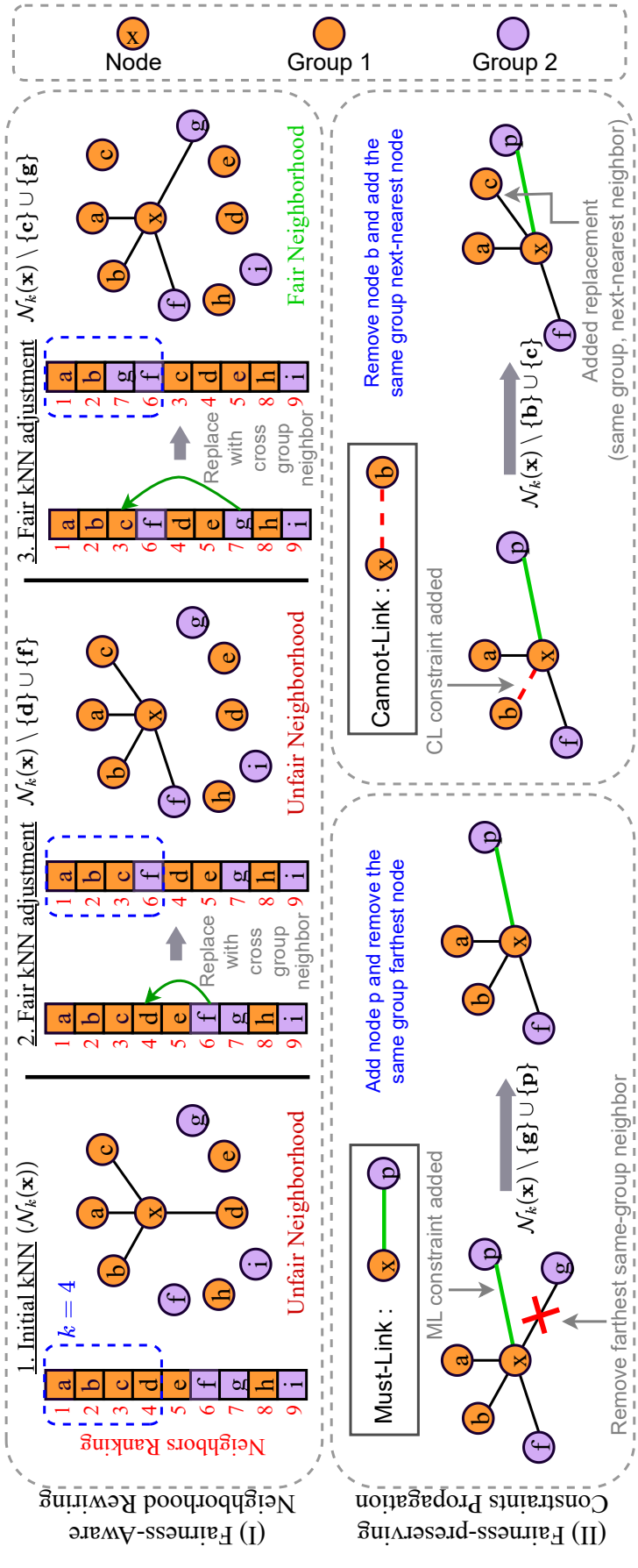


Figure 5.3: *Illustration of the proposed Fair Constrained neighborhood construction. (a) Fairness-aware neighborhood rewiring: the top-k nearest neighbors are selected and iteratively updated by replacing the farthest same-group nodes with the nearest nodes from different groups. (b) Fairness-preserving constraint propagation: ML and glsxtrshortCL constraints are integrated, and adjustments are made to preserve both fairness in neighborhood composition and k-regularity.*

the updated neighborhood $\mathcal{N}_4$(x)={a,f,p,c}. These updates ensure that both fairness and constraint satisfaction are preserved, producing a refined neighborhood.

The following subsections describe this process in detail through Phase I: Fairness-Aware Neighborhood Rewiring and Phase II: Fairness-preserving Constraints Propagation.

5.4.1.1 Fairness-Aware Neighborhood Rewiring

This phase begins with building a standard *k*NN graph from the dataset and then refining local neighborhoods to satisfy a fairness criterion. The goal is to mitigate disparate impact by ensuring that each node's neighborhood reflects a balanced representation of demographic groups. Specifically, the neighborhood of each node is examined for the distribution of sensitive groups. If an imbalance is detected, the farthest neighbor from an over-represented group is replaced with the next closest node from an underrepresented group, while maintaining k neighbors.

Fairness is controlled by a parameter α, which specifies the minimum required proportion of cross-group neighbors. The *k*NN graph is iteratively refined until every node's neighborhood satisfies this fairness condition, yielding the fair-*k*NN graph.

Definition 7 (Neighborhood Fairness)**.** *Let G be a kNN graph and consider a node $\mathbf{x}_i$ with neighborhood $\mathcal{N}_k(\mathbf{x}_i)$ of size k. Each node is associated with a sensitive group label $g(\mathbf{x}_i)$, where $g(\mathbf{x}_i) = t$ if $\boldsymbol{x}_i \in G_t$. Define*

$$n_i^{\text{diff}} = \left|\{\mathbf{x}_j \in \mathcal{N}_k(\mathbf{x}_i) : g(\mathbf{x}_j) \neq g(\mathbf{x}_i)\}\right| \tag{5.5}$$

as the number of neighbors of $\mathbf{x}_i$ that belong to groups different from $g(\mathbf{x}_i)$, and

$$n_i^{\text{same}} = \left|\{\mathbf{x}_j \in \mathcal{N}_k(\mathbf{x}_i) : g(\mathbf{x}_j) = g(\mathbf{x}_i)\}\right| \tag{5.6}$$

as the number of neighbors that share the same group as $\mathbf{x}_i$. The neighborhood $\mathcal{N}(\mathbf{x}_i)$ is deemed fair *if*

$$n_i^{\text{diff}} \;\geq\; \alpha \cdot n_i^{\text{same}}, \quad \alpha \in [0, 1]. \tag{5.7}$$

Where α is a parameter that controls the minimum required neighbors from different sensitive groups. Since $n_i^{\text{diff}} + n_i^{\text{same}} = k$, this condition can be equivalently written as

$$n_i^{\text{diff}} \geq \frac{\alpha}{1+\alpha} k. \tag{5.8}$$

Remark 2. *The parameter α controls degree of cross-group representation required in each neighborhood. When $\alpha = 0$, the definition reduces to the standard kNN neighborhood with no fairness requirement, while larger values of α enforce greater diversity. In practice, the choice of α is motivated by the* disparate impact criterion *commonly expressed as the* 80% *rule [12]. This provides a principled way to calibrate α so that local neighborhoods respect the disparate impact threshold.*

If a neighborhood $\mathcal{N}_k(\mathbf{x}_i)$ does not satisfy the fairness condition, its neighbors are adjusted through an iterative rewiring procedure to enforce the criterion consistently across the graph (see Algorithm 4). This criterion guarantees that each neighborhood contains a sufficient share of cross-group neighbors, preventing dominance of a single group and promoting balanced local structures. In particular, it avoids situations where a node is exclusively surrounded by its own group, thereby mitigating homophily.

5.4.1.2 Fairness-preserving Constraints Propagation

This phase is about Propagating ML and CL into the fair$k$NN graph to guide the graph construction further. A straightforward approach, such as directly adding or removing edges, may compromise the fairness guarantees established in the previous stage. For instance, adding an ML edge might displace a cross-group neighbor, potentially dropping a node's neighborhood below the fairness threshold. To address this challenge, we introduce a novel fairness-preserving constraint propagation framework. When a constraint forces a structural change (an edge addition or removal), our method performs a compensatory modification that is specifically designed to preserve the pre-existing count of neighbors from each sensitive group. This ensures the local fairness condition remains stable and robust throughout

Algorithm 4: Fair$k$NN

Input : Initial neighborhoods $\{\mathcal{N}_k(\mathbf{x}_i)\}_{i=1}^n$, group labels $g(\mathbf{x}_i)$, fairness threshold α
Output: Fair neighborhoods $\{\mathcal{N}_k^F(\mathbf{x}_i)\}_{i=1}^n$

for *each node* $\mathbf{x}_i \in X$ **do**
  Compute n_i^{diff} and n_i^{same} using equation (5.5) and (5.6).
  while $n_i^{\text{diff}} < \alpha \cdot n_i^{\text{same}}$ **do**
    Identify farthest same-group neighbor:
    $\mathbf{x}_r = \text{argmax}_{\mathbf{x}_p \in \mathcal{N}_k(\mathbf{x}_i),\, g(\mathbf{x}_p)=g(\mathbf{x}_i)}\, d(\mathbf{x}_i, \mathbf{x}_p)$
    Identify nearest cross-group node not already in $\mathcal{N}_k(\mathbf{x}_i)$:
    $\mathbf{x}_a = \text{argmin}_{\mathbf{x}_p \in \mathcal{X} \setminus \mathcal{N}_k(\mathbf{x}_i),\, g(\mathbf{x}_p) \neq g(\mathbf{x}_i)}\, d(\mathbf{x}_i, \mathbf{x}_p)$
    if $\mathbf{x}_a$ *exists* **then**
      Update neighborhood: $\mathcal{N}_k(\mathbf{x}_i) \leftarrow \big(\mathcal{N}_k(\mathbf{x}_i) \setminus \{\mathbf{x}_r\}\big) \cup \{\mathbf{x}_a\}$
      Recompute n_i^{diff}, n_i^{same}
    end
    else
      break
    end
  end
  $\mathcal{N}_k^F(\mathbf{x}_i) \leftarrow \mathcal{N}_k(\mathbf{x}_i)$
end
return $\{\mathcal{N}_k^F(\mathbf{x}_i)\}_{i=1}^n$

the constraint integration process.

For each $(\mathbf{x}_i, \mathbf{x}_j) \in \text{ML}$, a direct edge must connect the corresponding nodes in the graph. If $\mathbf{x}_j \in \mathcal{N}_k^F(\mathbf{x}_i)$, no modification is needed. Otherwise, $\mathbf{x}_j$ is added to the neighborhood of $\mathbf{x}_i$, which temporarily increases its size beyond k and violates the kNN structure. To restore the neighborhood size, one existing neighbor must be removed. An arbitrary removal of a neighbor could disrupt the neighborhood's sensitive group distribution, reintroducing bias. To prevent this, the fairness-preserving mechanism of Fair$k$NN is extended. Let $g(\mathbf{x}_j)$ denote the sensitive attribute of the newly added node. The neighborhood cardinality is restored by removing the node from the same group as $\mathbf{x}_j$ that is farthest from $\mathbf{x}_i$:

$$\mathbf{x}_{\text{rem}} = \underset{\mathbf{x}_p \in \mathcal{N}_k(\mathbf{x}_i) \cup \{\mathbf{x}_j\},\, g(\mathbf{x}_p)=g(\mathbf{x}_j)}{\text{argmax}} d(\mathbf{x}_i, \mathbf{x}_p).$$

In this way, the addition of $\mathbf{x}_j$ is exactly counterbalanced by the removal of the most distant node from the same group. This preserves the group distribution within $\mathcal{N}(\mathbf{x}_i)$ and ensures

that fairness is maintained while all ML constraints are satisfied (Algorithm 5).

Algorithm 5: ML Constraint Enforcement

procedure EnforceML$\big(\{\mathcal{N}_k(\mathbf{x}_i)\}_{i=1}^n, \text{ML}, g(\mathbf{x})\big)$
  forall $(\boldsymbol{x}_i, \boldsymbol{x}_j) \in \mathit{ML}$ **do**
    if $\mathbf{x}_j \notin \mathcal{N}_k(\mathbf{x}_i)$ **then**
      $X_{rem} \leftarrow \{\mathbf{x}_p \in \mathcal{N}_k(\mathbf{x}_i) \mid g(\mathbf{x}_p) = g(\mathbf{x}_j), (\mathbf{x}_i, \mathbf{x}_p) \notin \text{ML}\}$
      $\mathbf{x}_{rem} \leftarrow \arg\max_{\mathbf{x}_p \in X_{rem}} d(\mathbf{x}_i, \mathbf{x}_p)$
      $\mathcal{N}_k(\mathbf{x}_i) \leftarrow \big(\mathcal{N}_k(\mathbf{x}_i) \setminus \{\mathbf{x}_{rem}\}\big) \cup \{\mathbf{x}_j\}$
    end
  end
  return $\{\mathcal{N}_k(\mathbf{x}_i)\}_{i=1}^n$
end procedure

For each $(\mathbf{x}_i, \mathbf{x}_j) \in \text{CL}$, it must be ensured that no direct edge connects them. If an edge $(\mathbf{x}_i, \mathbf{x}_j)$ exists in the fair graph, it is removed. This action reduces the cardinality of $\mathcal{N}(\mathbf{x}_i)$ and introduces an unbalanced neighborhood corresponding to the group of the removed node. To restore both structural balance and fairness, the removed neighbor is replaced with a new candidate node. Specifically, a closest available node will be searched such that $\mathbf{x}_{\text{add}}$ belongs to the same sensitive group as $\mathbf{x}_j$. such that $\mathbf{x}_{\text{add}}$ belongs to the same sensitive group as $\mathbf{x}_j$.

$$\mathbf{x}_{\text{add}} = \underset{\mathbf{x}_p \in X \setminus \mathcal{N}_k(\mathbf{x}_i),\, g(\mathbf{x}_p) = g(\mathbf{x}_j)}{\operatorname{argmin}} d(\mathbf{x}_i, \mathbf{x}_p),$$

By replacing the removed neighbor with the nearest feasible candidate from the identical sensitive group, this method ensures that satisfying CL constraints does not degrade the fairness guarantees of the graph structure (Algorithm 6).

The resulting neighborhoods $\{\mathcal{N}_k^{FC}(\mathbf{x}_i)\}_{i=1}^n$ are simultaneously fair and constraint-compliant, and are used to construct the affinity matrix $\mathbf{S}_{FC}$ via a Gaussian kernel given in equation (5.1). This modified affinity matrix encodes both fairness and pairwise constraints while preserving the local *k*NN structure and similarity information. Normalized SC is then performed on the adjusted graph: the Laplacian $\mathbf{L}_{FC}$ is computed from $\mathbf{S}_{FC}$, followed by spectral embedding using its eigenvectors, and finally standard clustering methods (e.g., k-means) are applied to obtain the partitioning. The final clusters thus inherit the fairness guarantees and constraint satisfaction embedded in $\mathbf{S}_{FC}$, achieving a principled balance be-

Algorithm 6: CL Constraint Enforcement

procedure EnforceCL$\big(\{\mathcal{N}_k(\mathbf{x}_i)\}_{i=1}^n, \text{CL}, g(\mathbf{x})\big)$
  forall $(\boldsymbol{x}_i, \boldsymbol{x}_j) \in CL$ **do**
    if $\mathbf{x}_j \in \mathcal{N}_k(\mathbf{x}_i)$ **then**
      $X_{add} \leftarrow \{\mathbf{x}_p \in X \setminus (\mathcal{N}_k(\mathbf{x}_i)) \mid g(\mathbf{x}_p) = g(\mathbf{x}_j), (\mathbf{x}_i, \mathbf{x}_p) \notin \text{CL}\}$
      $\mathbf{x}_{add} \leftarrow \arg\min_{\mathbf{x}_p \in X_{add}} d(\mathbf{x}_i, \mathbf{x}_p)$
      $\mathcal{N}_k(\mathbf{x}_i) \leftarrow \mathcal{N}_k(\mathbf{x}_i) \cup \{\mathbf{x}_{add}\}$
      $\mathcal{N}_k(\mathbf{x}_i) \leftarrow \mathcal{N}_k(\mathbf{x}_i) \setminus \{\mathbf{x}_j\}$
    end
  end
  return return $\{\mathcal{N}_k(\mathbf{x}_i)\}_{i=1}^n$
end procedure

tween structural integrity, demographic representation, and domain-specific requirements (Algorithm 7)

5.4.2 Experiments and Results

This section presents the experiments designed to evaluate the performance of the proposed method on real-world datasets, with a particular focus on the role of pairwise constraints in SC. Pairwise constraints were randomly generated using class labels and varied to examine their influence on clustering quality and fairness. To assess the effectiveness of fairness considerations, both C-kNN and FC-kNN graph construction methods were implemented in Python, and their outcomes were compared. This evaluation demonstrates the contribution of the proposed fair constrained graph construction to improving fairness while ensuring clustering performance. The results and their corresponding discussions are presented in the following subsections.

5.4.2.1 Experimental Parameters

In the kNN graph construction, the number of nearest neighbors k is set to $k = \sqrt{n}$. For FC-kNN, the disparate impact parameter α was fixed at 0.8. The number of clusters c was set to the actual number of classes in each dataset. This is a principled choice because the pairwise constraints are generated based on the ground-truth labels. Maintaining c equal

Algorithm 7: FC-kNN Spectral Clustering

Input : Dataset $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$; sensitive group labels $g(\mathbf{x}_i)$; number of neighbors k; fairness threshold α; ML constraints; CL constraints; desired number of clusters c

Output: Partition of X into c clusters $\{\pi_1, \ldots, \pi_c\}$

// Step 1: Initial kNN Search

Compute initial neighborhoods $\{\mathcal{N}_k(\mathbf{x}_i)\}_{i=1}^n$

// Step 2: Fair Neighborhood Rewiring

$\{\mathcal{N}_k^F(\mathbf{x}_i)\}_{i=1}^n \leftarrow \textsc{FairkNN}(\{\mathcal{N}_k(\mathbf{x}_i)\}_{i=1}^n, g(x), \alpha)$ ▷ Algorithm 4

// Step 3: Pairwise Constraints Integration

$\{\mathcal{N}_k^{FC}(\mathbf{x}_i)\}_{i=1}^n \leftarrow \textsc{EnforceML}(\{\mathcal{N}_k^F(\mathbf{x}_i)\}_{i=1}^n, \text{ML}, g(x))$ ▷ Algorithm 5

$\{\mathcal{N}_k^{FC}(\mathbf{x}_i)\}_{i=1}^n \leftarrow \textsc{EnforceCL}(\{\mathcal{N}_k^F(\mathbf{x}_i)\}_{i=1}^n, \text{ML}, g(x))$ ▷ Algorithm 6

// Step 4: Affinity Matrix Construction

Build the affinity matrix: $\mathbf{S}_{FC}(i,j) = \begin{cases} \text{as in equation (5.1)}, & \mathbf{x}_j \in \mathcal{N}_k^{FC}(\mathbf{x}_i), \\ 1, & (\mathbf{x}_i, \mathbf{x}_j) \in \text{ML}, \\ 0, & (\mathbf{x}_i, \mathbf{x}_j) \in \text{CL}. \end{cases}$

// Step 5: Spectral Clustering

Compute degree matrix $\mathbf{D} = \text{diag}(\mathbf{S}_{FC})$

Form normalized Laplacian $\mathbf{L}_{FC} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}}\mathbf{S}_{FC}\mathbf{D}^{-\frac{1}{2}}$

Compute the c eigenvectors $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_c]$ of $\mathbf{L}_{FC}$ corresponding to the c smallest eigenvalues

Apply k-means to the rows of $\mathbf{U}$ to obtain clusters $\{\pi_1, \ldots, \pi_c\}$

return $\{\pi_1, \ldots, \pi_c\}$

to the number of true classes ensures that the evaluation of clustering quality and constraint satisfaction remains consistent and interpretable. Moreover, the number of pairwise constraints was systematically varied to examine its influence on clustering results.

5.4.2.2 Evaluation Against Fairness

Fairness is measured by how well the clustering maintains the ratio of sensitive groups. The results of constrained and fair constrained graph construction are reported for SC in the Table 5.1. It shows that FC-kNN outperforms the C-kNN on all the datasets w.r.t fairness (i.e., balance), demonstrating the method's motivation. FC-kNN consistently achieves

Table 5.1: *Balance and NCut comparison between C-kNN and FC-kNN across datasets under varying pairwise constraints.*

Dataset	Constraints ($\times 10^3$)	Balance		NCut		Dataset	Constraints ($\times 10^3$)	Balance		NCut	
		C-kNN	FC-kNN	C-kNN	FC-kNN			C-kNN	FC-kNN	C-kNN	FC-kNN
Bank	5	0.4123	**0.4428**	0.0432	**0.0256**	Crime	10	0.0379	**0.1897**	**0.2184**	0.2810
	10	0.3122	**0.4322**	**0.0632**	0.0723		20	0.0388	**0.1899**	**0.3025**	0.3290
	15	0.0884	**0.2216**	**0.0698**	0.0931		30	0.0570	**0.1770**	0.3733	**0.3564**
	20	0.1319	**0.1931**	**0.0705**	0.1106		50	0.0685	**0.1921**	**0.3945**	0.4042
	25	0.0534	**0.0638**	**0.0628**	0.1410		100	0.0798	**0.1450**	0.5187	**0.4979**
	30	0.2062	**0.4571**	**0.0756**	0.1585		120	0.0850	**0.1448**	**0.5526**	0.5605
	35	0.0592	**0.4485**	**0.0809**	0.1375		140	0.0889	**0.1436**	**0.5862**	0.6003
Catalonia	5	0.0546	**0.1407**	**0.0842**	0.2074	Student	1	0.0000	**0.2143**	**0.2026**	0.6876
	10	0.0306	**0.1415**	**0.1506**	0.3018		6	0.0893	**0.5217**	**0.2017**	0.7043
	15	0.0318	**0.1371**	**0.2269**	0.3668		9	0.3529	**0.5238**	**0.2021**	0.7301
	20	0.0248	**0.0909**	**0.2683**	0.4249		20	0.1000	**0.5000**	**0.2697**	0.7767
	25	0.0324	**0.0705**	**0.2131**	0.5389		40	0.0769	**0.1429**	**0.3256**	0.7622
	30	0.0576	**0.0736**	**0.1775**	0.5296		50	0.0000	**0.4828**	**0.4811**	0.7470
German	5	0.3121	**0.3152**	**0.0294**	0.0828	ILPD	5	0.0289	**0.0895**	0.1617	**0.0978**
	10	0.2804	**0.4310**	**0.0367**	0.0895		10	0.0417	**0.0739**	0.2852	**0.1823**
	15	0.3118	**0.4240**	**0.0420**	0.0892		15	0.0508	**0.1224**	0.3341	**0.2964**
	20	0.3122	**0.4077**	**0.0492**	0.0951		20	0.0875	**0.3162**	0.3885	**0.3682**
	25	0.3303	**0.4032**	**0.0592**	0.0979		25	0.1569	**0.2518**	**0.4326**	0.4467
	30	0.3280	**0.4069**	**0.0774**	0.1054		30	0.3051	**0.3129**	**0.4722**	0.4958
	35	0.3245	**0.4044**	**0.0979**	0.1108		35	0.2892	**0.3117**	**0.4643**	0.4652
Census	50	0.2931	**0.4263**	**0.1784**	0.3538	Law	20	**0.0596**	0.0537	**0.1981**	0.3337
	100	0.2194	**0.4215**	**0.1725**	0.4673		40	0.0584	**0.1374**	**0.2267**	0.3898
	150	0.1934	**0.4332**	**0.1670**	0.4287		60	0.0817	**0.1393**	**0.2343**	0.4210
	200	0.1869	**0.4189**	**0.1618**	0.3775		80	0.0677	**0.0963**	**0.2614**	0.4720
	250	0.1826	**0.4297**	**0.1572**	0.3462		100	0.0838	**0.1256**	**0.2905**	0.4979
	300	0.1973	**0.4238**	**0.1530**	0.3362		120	0.0893	**0.1792**	**0.3215**	0.5605

higher balance scores compared to C-kNN (without fairness). Even under strict pairwise constraints, the proposed FC-kNN method maintains a balanced proportion of sensitive groups in each cluster, whereas the baseline often exhibits significant deviations. As shown in Table 5.1, FC-kNN achieves notable absolute improvements in balance across all datasets, ranging from 0.01 – 0.39 (avg. 0.14) for Bank, 0.02 – 0.11 (avg. 0.07) for Catalonia, 0.003 – 0.15 (avg. 0.09) for German, 0.13 – 0.25 (avg. 0.21) for Census, 0.05 – 0.15 (avg. 0.10) for Crime, 0.07 – 0.48 (avg. 0.29) for Student, 0.01 - 0.23 (avg. 0.07) for ILPD, and -0.01 - 0.09 (avg. 0.04) for Law datasets across different constraint levels, demonstrating that the proposed fairness-aware graph construction enables more balanced neighborhood formation. Since the focus is on constructing the fair graph using the disparate impact rule, it maintains the balance on each node for each group so that the obtained clusters will reflect the balanced group representation. Thus, the proposed method consistently achieves better balance even under varying constraint levels. Importantly, integrating ML and CL constraints alongside fairness requirements does not compromise fairness performance.

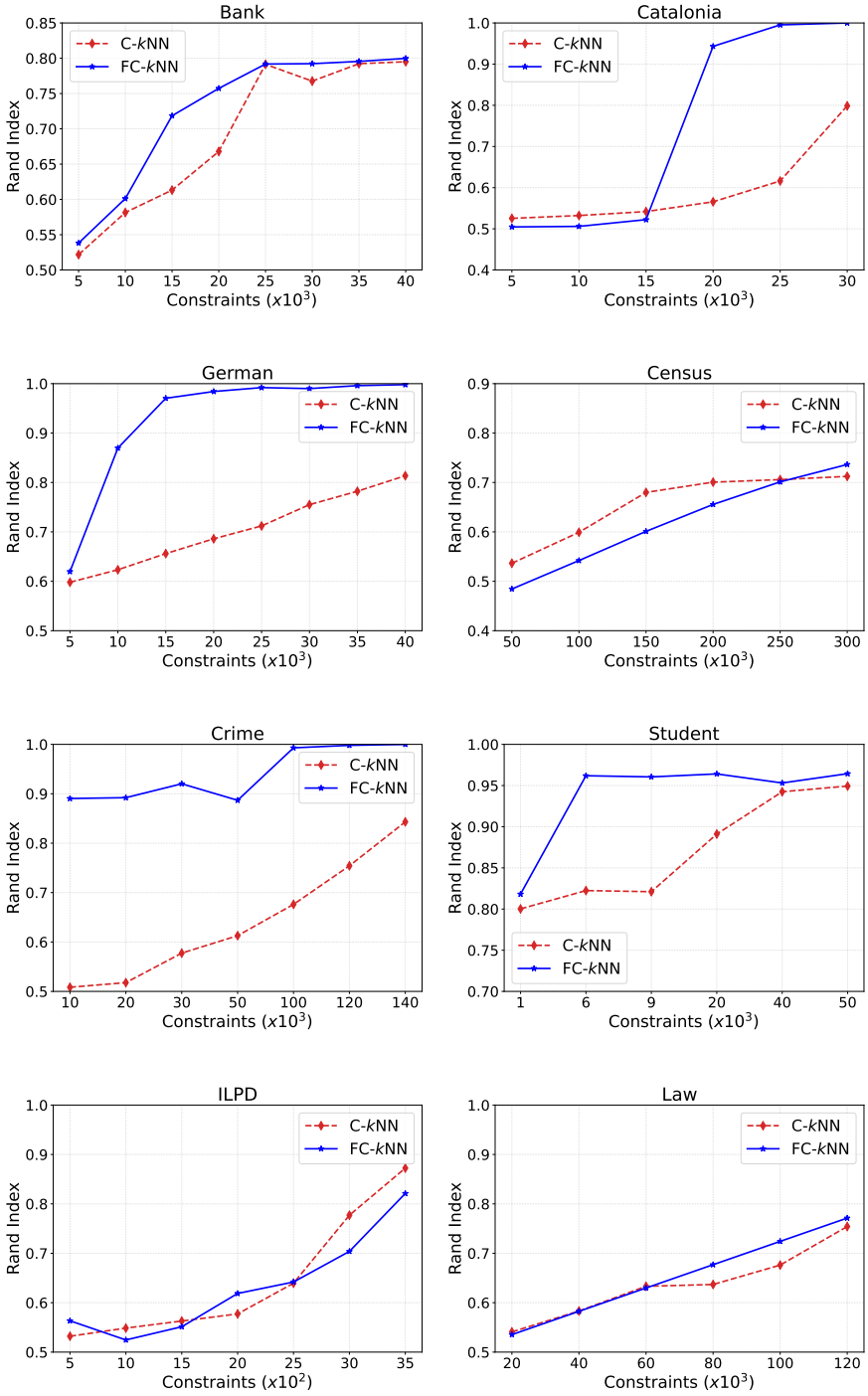


Figure 5.4: *Clustering performance of C-kNN and FC-kNN across eight datasets under varying numbers of pairwise constraints, with the number of clusters c fixed to the ground-truth labels. The plots report the RI, showing that the proposed FC-kNN method consistently achieves competitive clustering quality.*

5.4.2.3 Evaluation Against Clustering Quality

The clustering quality is evaluated using two widely adopted measures: NCut and RI. Table 5.1 reports the NCut values for both C-kNN and FC-kNN SC methods. The NCut values generally show an upward trend for both C-kNN and FC-kNN methods as the number of pairwise constraints increases. While the initial NCut values are low, indicating good separation with fewer pairwise constraints, they rise significantly with the addition of further pairwise constraints. This implies that the graph structure becomes less stable under heavier supervision. Moreover, the proposed FC-kNN method exhibits higher NCut values than C-kNN in most cases, reflecting the influence of fairness-aware neighborhood modifications. These adjustments not only enforce demographic balance but also integrate constraint satisfaction simultaneously, resulting in a slight increase in the cut cost. However, for a small number of pairwise constraints, the difference between C-kNN and FC-kNN remains minimal, suggesting that the fairness component becomes more prominent as constraint density grows.

To further assess clustering quality, the RI is compared between C-kNN and FC-kNN SC across a varying number of pairwise constraints. As shown in Figure 5.4, the RI values generally increase with the number of pairwise constraints, indicating improved alignment with the true cluster structure. In most cases, FC-kNN achieves comparable or higher RI scores than C-kNN, particularly at moderate and higher constraint levels. This improvement can be attributed to the fairness-aware neighborhood construction, which ensures cross-group representation and prevents fairness degradation while leveraging the added pairwise constraints more effectively. Overall, the RI results highlight that both methods benefit from pairwise constraints, with FC-kNN providing a more balanced clustering outcome.

To further evaluate the clustering quality obtained by the proposed approaches, an internal cluster validation metric, namely the Silhouette Score, was computed for all datasets to assess the compactness and separation of the clusters generated by C-kNN and FC-kNN. Table 5.2 presents a comparison of Silhouette scores between the C-kNN and FC-kNN across different datasets and varying numbers of constraints. From the obtained results, it can be

Table 5.2: *Silhouette score comparison between C-kNN and FC-kNN across datasets under varying pairwise constraints.*

Dataset	Constraints ($\times 10^3$)	Silhouette Score		Dataset	Constraints ($\times 10^3$)	Silhouette Score	
		C-kNN	FC-kNN			C-kNN	FC-kNN
Bank	5	0.4521	0.4186	Crime	10	0.4027	0.3894
	10	0.4413	0.4097		20	0.3928	0.3811
	15	0.4302	0.3984		30	0.3836	0.3725
	20	0.4215	0.3892		50	0.3719	0.3612
	25	0.4001	0.3712		100	0.3604	0.3485
	30	0.3914	0.3648		120	0.3498	0.3407
	35	0.3827	0.3585		140	0.3411	0.3325
Catalonia	5	0.5114	0.4981	Student	1	0.5314	0.5288
	10	0.5032	0.4876		6	0.5247	0.5293
	15	0.4918	0.4789		9	0.5183	0.5242
	20	0.4801	0.4664		20	0.5071	0.4996
	25	0.4695	0.4541		40	0.4916	0.4824
	30	0.4583	0.4435		50	0.4809	0.4713
German	5	0.4732	0.4618	ILPD	5	0.4216	0.4185
	10	0.4645	0.4524		10	0.4138	0.4102
	15	0.4538	0.4417		15	0.4025	0.3989
	20	0.4426	0.4313		20	0.3917	0.3894
	25	0.4314	0.4202		25	0.3812	0.3798
	30	0.4218	0.4109		30	0.3725	0.3696
	35	0.4105	0.4016		35	0.3631	0.3592
Census	50	0.3817	0.3698	Law	20	0.3924	0.3896
	100	0.3726	0.3605		40	0.3838	0.3807
	150	0.3618	0.3517		60	0.3745	0.3714
	200	0.3521	0.3429		80	0.3632	0.3609
	250	0.3414	0.3326		100	0.3541	0.3518
	300	0.3322	0.3238		120	0.3436	0.3417

observed that both methods achieve positive Silhouette Scores across all datasets, indicating meaningful cluster structures. In most cases, the C-kNN method achieves slightly higher Silhouette Scores than the FC-kNN method. This behavior is expected because incorporating fairness constraints into FC-kNN imposes additional restrictions during clustering, which may slightly reduce cluster compactness to maintain fairness among sensitive groups. The Silhouette Score decreases gradually as the number of constraints increases across all datasets. A similar trend is observed for both C-kNN and FC-kNN methods, indicating that increasing constraints can affect cluster separability. Nevertheless, FC-kNN maintains competitive clustering quality while enforcing fairness constraints.

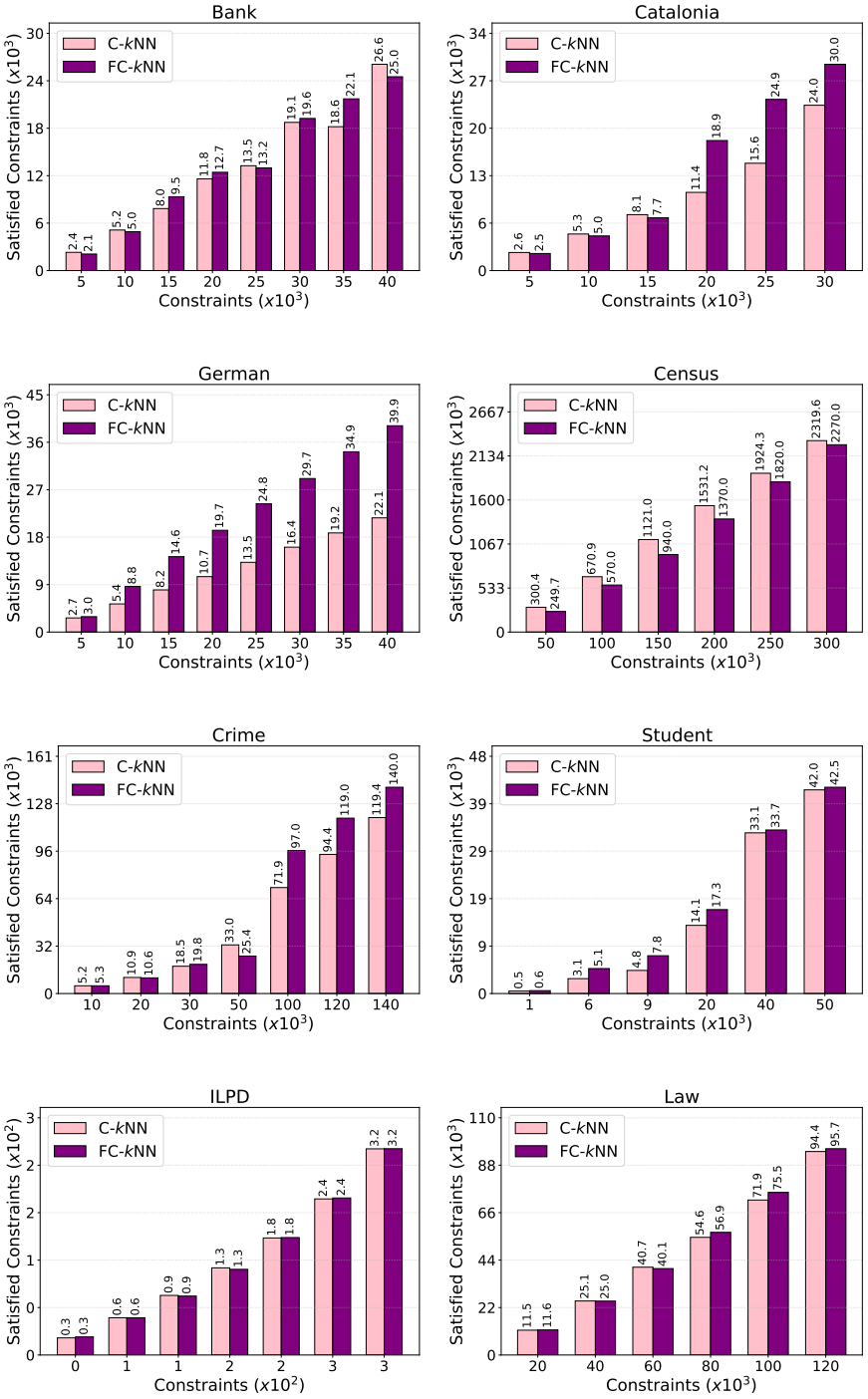


Figure 5.5: *Comparison of satisfied pairwise constraints between C-kNN and FC-kNN across eight datasets under varying numbers of pairwise constraints, with clusters set to the ground-truth labels.*

5.4.2.4 Constraints Satisfaction

The constraint satisfaction metric evaluates how effectively each method fulfills the given pairwise relationships. As the number of pairwise constraints increases, the satisfaction ratio in FC-kNN generally shows a slight upward trend, indicating that more pairwise constraints are being respected during clustering, as shown in Figure 5.5. When only a small number of pairwise constraints are provided, both methods perform comparably. However, occasional minor decreases are observed, likely due to conflicts among pairwise constraints as their number increases. For the Census, Law, and Bank datasets, FC-kNN exhibits nearly similar or slightly reduced satisfaction at higher constraint levels, which can be attributed to the denser and larger constraint sets. For example, on the Bank dataset, both methods perform comparably for fewer pairwise constraints, but at 35×10^3 constraints, FC-kNN satisfies about 19% more constraints than C-kNN. On the Catalonia dataset, the gap widens with more supervision: FC-kNN achieves a 25% gain at 30×10^3, where it fully satisfies all the given pairwise constraints. These results highlight a consistent pattern across datasets: constraint satisfaction is comparable under limited supervision, but with more constraints, FC-kNN not only preserves fairness but also adheres to a larger number of pairwise relationships. This trend aligns with the higher RI observed earlier, since improved constraint satisfaction directly contributes to better clustering quality.

5.4.2.5 Sensitivity Analysis of FC-kNN Parameters

To assess the robustness of the proposed method, experiments were conducted with different values of k. Furthermore, the disparate impact parameter α varied to examine its influence on balance.

Impact of k: This analysis focuses on the effect of varying the neighborhood size k on the balance of the proposed FC-kNN method. Fig. 5.6 reports the results for eight datasets under a fixed number of pairwise constraints. At smaller neighborhood sizes, the balance score of FC-kNN is relatively high, indicating that fairness is preserved when local neigh-

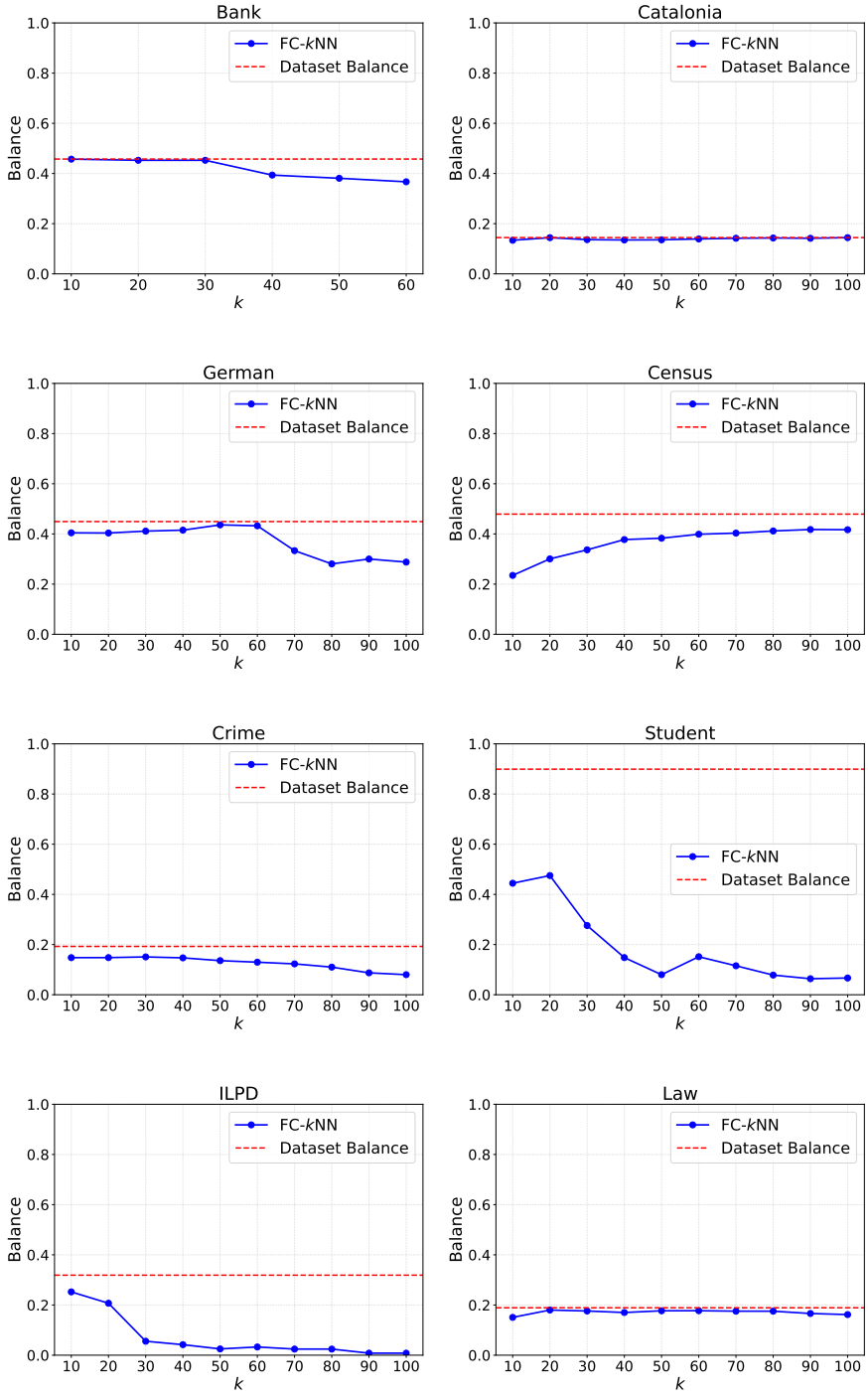


Figure 5.6: *Analysis of the effect of the number of nearest neighbors k on cluster balance under fixed pairwise constraints. The figure compares FC-kNN with the dataset balance across four datasets: Bank (50k constraints), Catalonia (2k constraints), German (25k constraints), Census (50k constraints), Crime (50k constraints), Student (8k constraints), ILPD (2k constraints), and Law (50k constraints).*

borhoods remain compact. As k increases, a slight decline in balance is observed. The results suggest that while small to moderate k values maintain stable fairness levels, larger k values lead to imbalanced neighborhoods and reduced fairness in clustering. An important observation is that the effectiveness of pairwise constraints depends on k. With increasing k, the influence of pairwise constraints tends to weaken, as more edges dilute the effect of specific constraint-based modifications. Consequently, fairness deteriorates as k grows with pairwise constraints. Thus, compact neighborhoods not only allow pairwise constraints to operate with greater impact but also preserve cross-group diversity more effectively, whereas large neighborhoods introduce structural bias and reduce fairness.

Moreover, it is not always possible to maintain exactly k neighbors in the modified graph since the inclusion of ML and CL constraints may require adding or removing edges, thereby slightly altering the neighborhood structure.

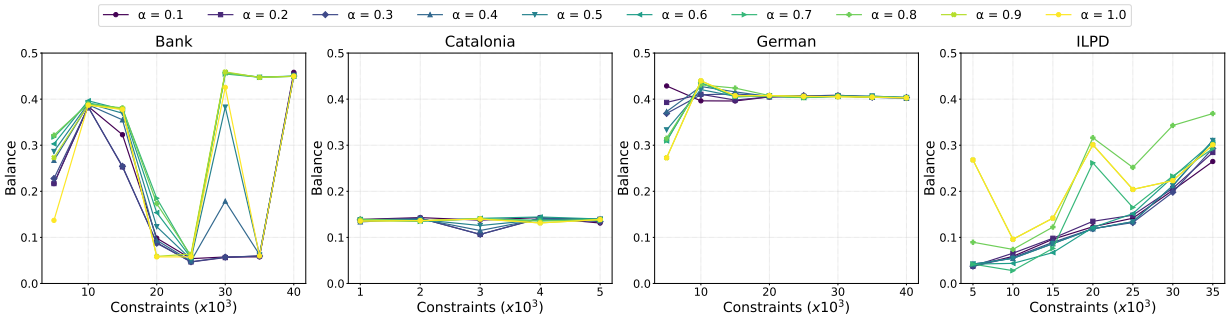


Figure 5.7: *Effect of the fairness parameter α on cluster balance across varying numbers of pairwise constraints, demonstrating how FC-kNN maintains fairness.*

Impact of parameter α: The experiments shown in Fig. 5.7 demonstrate how cross-group representation in FC-kNN evolves under varying fairness parameter $\alpha \in \{0.1, 0.2, \cdots, 1\}$ in equation (5.7). Balance is measured under different levels of pairwise constraints on Bank, Catalonia, German, and ILPD datasets. As α increases, the level of cross-group representation in the kNN graph systematically improves, which directly affects the balance values of the clustering. At very low values of ($\alpha = 0.1$–0.2), the fairness condition is weak, since only a small fraction of cross-group neighbors is required. For example, when $\alpha = 0.1$, a node with 10 same-group neighbors only needs one different-group neighbor, leaving neigh-

borhoods dominated by the same group. As a result, balance values remain low. As α increases to moderate levels (0.3–0.6), more cross-group connections are enforced, leading to noticeable improvements in neighborhood diversity. At higher values ($\alpha = 0.7$–0.8), the graphs achieve their strongest fairness levels. For instance, when $\alpha = 0.8$, the condition requires at least 8 different-group neighbors per 10 same-group neighbors, yielding strongly mixed neighborhoods. This corresponds to the disparate impact rule from fairness literature [12], a common criterion for reducing disparate impact. Beyond this, values of $\alpha \geq 0.9$ approach infeasibility, since not every node can achieve such a high proportion of cross-group neighbors. At very high α, equation (5.8) may require n_i^{diff} values that are locally impossible; many nodes cannot find enough different-group neighbors within a reasonable distance, making $\alpha \to 1$ infeasible.

The integration of ML and CL constraints may counteract or reinforce the fairness adjustments. A well-chosen α thus facilitates constraint integration by maintaining structural diversity in neighborhoods. Across all datasets, $\alpha = 0.8$ consistently achieves the best trade-off between feasibility and fairness. It enforces strong cross-group representation and preserves balance even under the influence of ML and CL constraints. This demonstrates that fairness in neighborhood graph construction is highly sensitive to the choice of α: small values leave the graph biased, excessively large values become infeasible, and the intermediate choice of $\alpha = 0.8$ achieves the most effective balance.

5.5 Summary

In the second contributory chapter, the FC-kNN graph is introduced, a novel approach that integrates fairness considerations and pairwise constraints directly into the similarity graph construction stage for SC. By enforcing group balance conditions within each node's neighborhood, FC-kNN mitigates disparate impact and ensures that sensitive groups are adequately represented. At the same time, the incorporation of ML and CL constraints preserves meaningful cluster structure. Experimental results across multiple real-world datasets demonstrate that FC-kNN achieves substantial improvements in fairness compared

to C-kNN, while maintaining competitive clustering quality and high constraint satisfaction. These findings confirm the effectiveness of embedding fairness and pairwise constraints jointly into the neighborhood graph, rather than treating them as separate objectives. Additionally, the proposed FC-kNN framework is designed with scalability in mind; by operating on sparse neighborhood graphs, it significantly reduces computational and memory complexity compared to fully connected graph representations. Future improvements can further enhance scalability for large-scale clustering applications through techniques such as approximate nearest-neighbor search, NN-Descent-based graph construction, and sparse eigensolvers.

Having incorporated fairness into constrained SC, the dissertation has addressed one major branch of advanced SC methods that improve clustering quality by leveraging pairwise constraints. The contributions presented in Chapters 4 and 5 demonstrate how fairness can be embedded in both the optimization formulation and the graph construction process. This approach ensures that different demographic groups receive fair treatment while leveraging ML and CL relationships. Although these methods enhance the fairness and performance of the traditional two-stage SC pipeline, they inherently work within separate frameworks: first, constructing the graph, then performing the clustering. This separation can lead to information loss and sub-optimal results.

To overcome these limitations, it is essential to consider a framework that integrate all SC steps. One such improvement is one-step spectral rotation clustering. This method brings together graph construction and cluster assignment into a single optimization framework, providing greater flexibility in incorporating fairness directly into the optimization criterion, thereby avoiding the limitations of the traditional two-step approach.

To address these limitations, the next chapter explores a one-step approach for SC. This is a unified framework that learns a fair spectral representation and cluster assignment within a single optimization procedure.

"The biggest risk is not taking any risk... In a world that's changing really quickly, the only strategy that is guaranteed to fail is not taking risks."

~ Mark Zuckerberg

6

Fairness in Unified Spectral Clustering

This chapter presents a fairness-aware extension of the OSSRC framework. Unlike the conventional two-step process, the one-step formulation jointly learns the graph structure and the cluster assignments together in a single optimization step. Building on this advantage, group fairness is integrated directly into the objective function so that fairness influences the clustering mechanism with each iteration. The proposed method adaptively updates both the similarity graph and the cluster indicators, enabling the graph to evolve in response to fairness considerations. Through the integration of fairness, spectral embedding, and adaptive graph learning, the framework aims to produce clusters that preserve clustering quality while remaining demographically balanced.

6.1 Introduction

One Step Spectral Rotation Clustering (OSSRC) methods were introduced to address the drawbacks of traditional multi-stage SC methods. By jointly optimizing graph learning, spectral embedding, and cluster assignment together in a single objective, one-step formulations avoid the performance degradation that typically arises when these components are optimized separately. This unified structure enables the graph and clustering to influence each other more effectively, often resulting in improved clustering quality. However, despite these advantages, existing one-step SC techniques do not consider fairness with respect to sensitive attributes.

To address this gap, a few recent studies have attempted to bring fairness into such unified models. For example, Zhang *et al.* [33] introduced a unified fair SC approach that employs a node-adaptive graph filter for graph construction, effectively learning graphs from noisy data. Furthermore, they integrated the balance constraint proposed in [25] into their framework as a constrained optimization problem. However, a major drawback of this method is that it considers the balance constraint as a separate component within the optimization criterion, which may not fully capture or directly influence the fairness requirements across clusters. To alleviate these limitations, a Fair One-Step Spectral Rotation clustering (FOSSRC) framework is proposed, which is directly embedded into the optimization objective. The framework includes a novel Group Fairness Term (GFT) to obtain an equitable representation of sensitive groups across all clusters.

The contribution of this chapter is described as follows:

- GFT is introduced on the cluster indicator matrix to ensure equal representation of sensitive groups within clusters.
- The effect of the GFT term is theoretically analyzed. Motivated by the analysis, GFT is integarted with the OSSRC method.
- The balance of clustering is adaptively adjusted through iterative updates within a one-step learning framework until convergence.

- The proposed fair model is evaluated on real-world datasets, demonstrating its effectiveness in improving fairness and clustering quality.

In this chapter, the OSSRC method is first described, and its limitations with respect to fairness are highlighted in Section 6.2. GFT is then introduced in Section 6.3.1, forming the basis for achieving fairness. Next, the fair version of OSSRC is presented, demonstrating how GFT is integrated to ensure fairness in Section 6.3.2. The method is subsequently evaluated through experiments in Section 6.3.3, followed by a summary in Section 6.4.

6.2 One-Step Spectral Rotation Clustering via Adaptive Graph Learning

In this section, a model for OSSRC is provided. Unlike the two-stage SC approach, The OSSRC method learns the similarity matrix and performs SC simultaneously. It works directly in the low-dimensional feature space of the original data, reducing noise and dimensionality issues. Building on [108], which integrates adaptive graph learning into unsupervised feature selection, this work incorporates spectral rotation for clustering, inspired by one-step spectral rotation approaches [44, 100, 102, 109, 110]. The OSSRC model integrates affinity graph learning, spectral embedding, and spectral rotation into a one-step framework to produce final clustering results.

Given a data set $\mathbf{X} \in \mathbb{R}^{n\times d}$ with n samples and d dimension. Assume $\mathbf{M} \in \{0,1\}^{n\times c}$ is the binary index matrix. We aim to find an orthogonal projection matrix $\mathbf{W} \in \mathbb{R}^{d\times c}$ that finds a low-dimension subspace when linearly integrated with the data matrix $\mathbf{X}$. We follow the literature [108] with spectral rotation term and have the objective function given below:

$$\begin{aligned}\min_{\mathbf{W},\mathbf{S},\gamma,\mathbf{U},\mathbf{R},\mathbf{Y}} \ & \operatorname{Tr}(\mathbf{W}^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{W}) + \gamma\|\mathbf{S}\|_F^2 - \rho\operatorname{Tr}(\mathbf{W}^T\mathbf{X}\mathbf{U}\mathbf{U}^T\mathbf{X}^T\mathbf{W}) \\ & + \|\mathbf{Y} - \mathbf{X}\mathbf{W}\mathbf{R}\|_F^2 + \tau\|\mathbf{W}\|_{2,1}, \\ \text{s.t. } & \mathbf{W}^T\mathbf{W} = \mathbf{I},\ \mathbf{U}^T\mathbf{U} = \mathbf{I},\ \mathbf{R}^T\mathbf{R} = \mathbf{I},\ \mathbf{S}^T\mathbf{1} = \rho\mathbf{1}.\end{aligned} \tag{6.1}$$

The first term captures the important correlation of the data as explained in equation

(2.14), where the matrix $\mathbf{L}$ represents the Laplacian, defined as $\mathbf{L} = \mathbf{D} - \mathbf{S}$, where $\mathbf{S} \in \mathbb{R}^{n \times n}$ represents the similarity matrix and $\mathbf{D}$ is the degree matrix with diagonal entries given by $D_{ii} = \sum_j (s_{ij})$. The second term contributes towards the iterative update of matrix $\mathbf{S}$ and refines the structure of the subspace graph. The third term is to maximize the between-class scatter matrix in the subspace where indicator matrix $\mathbf{U} = \mathbf{M}(\mathbf{M}^T\mathbf{M})^{-1/2}$ is relaxed from a discrete form to an orthogonal one that $\mathbf{U}^T\mathbf{U} = \mathbf{I}$. The fourth term is the objective of spectral rotation to obtain the clusters as given in equation (2.12), where $\mathbf{R}$ and $\mathbf{Y}$ are the rotation and cluster indicator matrices, respectively. Due to the sensitivity of k-means to the initial selection of cluster centers, the resulting solution may differ from true discrete clustering. To address this, spectral rotation is used as proposed in [49]. The fifth term is the $\ell_{2,1}-$norm on $\mathbf{W}$ for feature selection to reduce redundancy and outliers. The regularization parameter τ adjusts the influence of feature selection. We symmetrize the similarity matrix as $\mathbf{S} = (\mathbf{S}^T + \mathbf{S})/2$ to enhance the connection between closer data points in the subspace and impose a constraint $\mathbf{S}^T\mathbf{1} = \rho\mathbf{1}$, where ρ is the controlling parameter. The coefficient γ is distinct from other parameters (i.e., τ, ρ) and is adaptively updated throughout the optimization process.

6.3 Fair One-Step Spectral Rotation Clustering via Adaptive Graph Learning

6.3.1 Group Fairness Term

A new perspective is provided on the definition of balance. According to Definition 1, balance is defined as the equitable distribution of sensitive groups across all clusters. To understand the distribution of sensitive groups, we must first examine the imbalance within clusters, as outlined in the following definition:

Definition 8. *(Imbalance) Given a dataset* $\mathbf{X} \in \mathbb{R}^{n \times d}$ *with* n *data points and* d *features. The dataset contains* m *sensitive groups* $\{G_1, \ldots, G_m\}$ *and partitioned into* c *clusters* $\pi =$

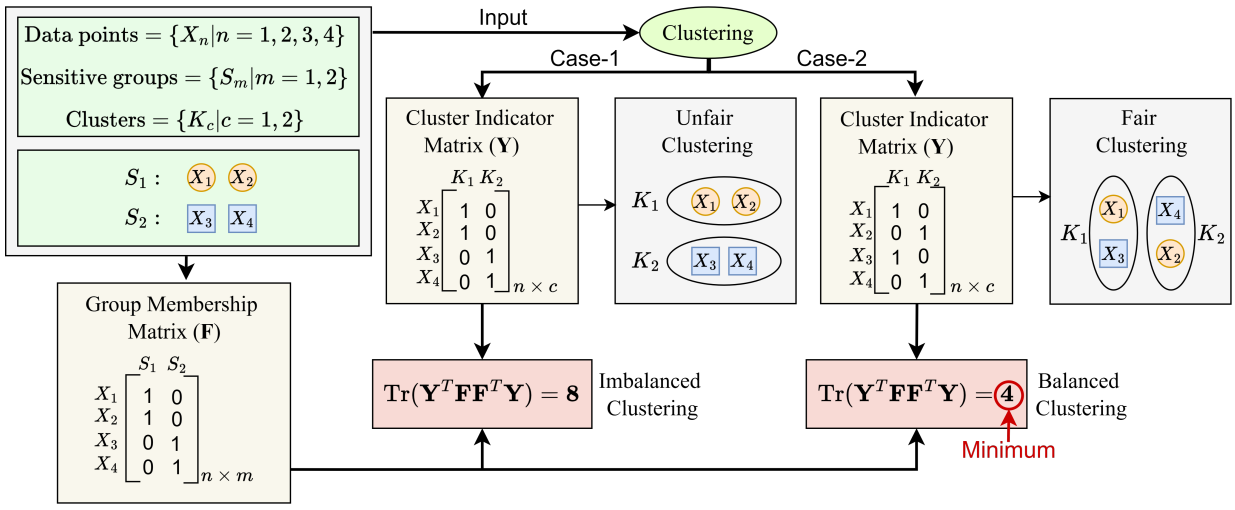


Figure 6.1: *An example to demonstrate the effectiveness of the GFT. We considered $n = 4$ data points having two sensitive groups, G_1 and G_2, with two clusters: π_1 and π_2.*

$\{\pi_1, \ldots, \pi_c\}$. *$N_t(\ell)$ represents total number of data points belonging to sensitive group G_t, where G_t represents t-th group ($t \in \{1, \ldots, m\}$) in cluster $\ell \in \{1, 2, \ldots, c\}$ (i.e.,$|\pi_\ell \cap G_t|$). The ideal case is that each cluster ℓ should have $\frac{|G_t|}{c}$ data points to maintain balance. Thus, the objective of achieving fair clusters is to minimize the imbalance, as defined below:*

$$Imbalance = \frac{1}{c}\sum_{\ell=1}^{c}\left(N_t(\ell) - \frac{|G_t|}{c}\right)^2, \ \forall t \in \{1, \ldots, m\} \tag{6.2}$$

From the perspective of Definition 8, balance is ensured if the number of data points belonging to the G_t in each cluster is as close as possible to the proportion of G_t in the entire dataset (i.e., $\frac{|G_t|}{c}$). Our aim here is to minimize the imbalance to obtain the equal distribution of sensitive groups ($\frac{|G_t|}{c}$) across all the clusters. In order to achieve this, a GFT is introduced that penalizes deviations from the ideal distribution $N_t(\ell) = \frac{|G_t|}{c}$, reducing imbalance for all the groups.

Theorem 6.3.1. *Let $\mathbf{F} \in \{0, 1\}^{n\times m}$ be a sensitive group membership matrix where $F_{it} = 1$ indicates that data point i belongs to sensitive group t and $F_{it} = 0$ otherwise, and $\mathbf{Y} \in \{0, 1\}^{n\times c}$ be a cluster indicator matrix where $Y_{i\ell} = 1$ if data point i assigned to ℓ^{th} cluster and $Y_{i\ell} = 0$ otherwise. The GFT is expressed as:*

$$T(Y) = \text{Tr}(\mathbf{Y}^T\mathbf{F}\mathbf{F}^T\mathbf{Y}) \tag{6.3}$$

Minimizing T(Y) ensures an even distribution of sensitive groups across clusters, reducing the concentration of the same sensitive groups in a single cluster.

Proof. Let $(\mathbf{F}\mathbf{F}^T)_{ij} = \sum_{t=1}^{m} F_{it}F_{jt}$ denote the $(i,j) - th$ entry of $\mathbf{F}\mathbf{F}^T$. This entry equals 1 if data points i and j belong to the same sensitive group and 0 otherwise. The term $\mathbf{Y}^T\mathbf{F}\mathbf{F}^T\mathbf{Y}$ measures how data points from the same sensitive group are distributed across clusters. The trace $\text{Tr}(\mathbf{Y}^T\mathbf{F}\mathbf{F}^T\mathbf{Y}) = \sum_{\ell=1}^{c}(\mathbf{Y}^T\mathbf{F}\mathbf{F}^T\mathbf{Y})_{\ell\ell}$ sums the diagonal values, each of which represents the total number of pairs of data points from the same sensitive group that are assigned to the same cluster. Specifically, for each sensitive group t, the diagonal value counts all pairs of data points (both self-pair and cross-pair) that are assigned to the same cluster, which is given as $T(Y) = \sum_{t=1}^{m}\sum_{i,j\in G_t} 1[i \in \pi_\ell, j \in \pi_\ell]$. A high value of $T(Y)$ indicates a large number of data point pairs within the same sensitive group are assigned to the same cluster, leading to an imbalance, as shown in Figure 6.1. Therefore, minimizing the $T(Y)$ promotes the distribution of members from the same sensitive group across different clusters, reducing the imbalance. The ideal case for minimizing the $T(Y)$ occurs when no two data points from the same sensitive group are assigned to the same cluster, leading to $Y_{i\ell}Y_{j\ell} = 0$ for all i and j such that $F_{it}F_{jt} = 1$. However, this ideal condition is not achievable practically, as the number of clusters is small relative to the number of data points in the sensitive groups. Based on the Definition 8, we aim each cluster ℓ to contain approximately $\frac{|G_t|}{c}$ data points from group G_t. If $N_t(\ell) = \frac{|G_t|}{c}$ for all clusters, then the sensitive group G_t is evenly distributed among the clusters. In this scenario, The value of $T(Y)$ will be minimum as demonstrated in the fair clustering case of Figure 6.1. Conversely, if most members of group G_t are concentrated in a single cluster ℓ, then $N_t(\ell) \gg \frac{|G_t|}{c}$, leads to a larger $T(Y)$. Thus, minimizing $T(Y)$ in equation (6.3) indirectly reduces the imbalance defined in equation (6.2), pushing $N_t(\ell)$ closer to $\frac{|G_t|}{c}$ for all clusters. □

6.3.2 FOSSRC

The model in equation (6.1) is for OSSRC without consideration of fairness. To achieve fairness in OSSRC , we proposed a novel fair model that integrates the GFT of equation (6.3) in the objective function of OSSRC in equation (6.1) and penalizes imbalance in which members of the same sensitive group are overrepresented in a cluster. The proposed optimization problem of FOSSRC is represented as:

$$\begin{aligned} \min_{\mathbf{W},\mathbf{S},\gamma,\mathbf{U},\mathbf{R},\mathbf{Y}} \; & \operatorname{Tr}(\mathbf{W}^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{W}) + \gamma\|\mathbf{S}\|_F^2 - \rho\operatorname{Tr}(\mathbf{W}^T\mathbf{X}\mathbf{U}\mathbf{U}^T\mathbf{X}^T\mathbf{W}) \\ & + \|\mathbf{Y} - \mathbf{X}\mathbf{W}\mathbf{R}\|_F^2 + \tau\operatorname{Tr}(\mathbf{W}^T\mathbf{O}\mathbf{W}) + \delta\operatorname{Tr}(\mathbf{Y}^T\mathbf{F}\mathbf{F}^T\mathbf{Y}), \\ & \text{s.t. } \mathbf{W}^T\mathbf{W} = \mathbf{I},\ \mathbf{U}^T\mathbf{U} = \mathbf{I},\ \mathbf{R}^T\mathbf{R} = \mathbf{I},\ \mathbf{S}^T\mathbf{1} = \rho\mathbf{1}. \end{aligned} \tag{6.4}$$

To simplify the calculation, the regularization term $\|\mathbf{W}\|_{2,1}$ of equation (6.1) is converted into a matrix form $\operatorname{Tr}(\mathbf{W}^T\mathbf{O}\mathbf{W})$ in equation (6.4) by taking advantage of Iteratively Reweighted Least Square (IRLS) [111] where $\mathbf{O}$ is a $d \times d$ diagonal matrix with $o_{ii} = \frac{1}{2\|\mathbf{w}_i\|_2}$. Here, δ is a balance (i.e., group fairness) parameter that adjusts the extent of the balancing effect during the learning process. Thus, similarity graph learning, dimensionality reduction, and group fairness are jointly optimized, ensuring fairness while preserving clustering quality within a unified framework.

Therefore, we obtain the group fairness regularized term $\operatorname{Tr}(\mathbf{Y}^T\mathbf{F}\mathbf{F}^T\mathbf{Y})$. According to Theorem 6.3.1, by minimizing the regularized term, we can achieve fair clusters.To understand the idea of our proposed group fairness term, we have taken a small example for $n = 4$ data points $X = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4\}$ with $m = 2$ sensitive groups and $c = 2$ clusters.The working of the GFT is shown in Figure 6.1.

6.3.2.1 Model Optimization

The problem in equation (6.4) is not jointly convex with respect to all variables $\mathbf{W}, \mathbf{S}, \gamma, \mathbf{U}, \mathbf{R}$, and $\mathbf{Y}$. However, It is convex with respect to each variable when the remaining variables are fixed. We utilized an alternative optimization strategy to optimize each variable it-

eratively by keeping the others fixed, continuing this process until the convergence. The corresponding pseudo is summarized in Algorithm 9, and the details are explained below:

Algorithm 8: Steps for the optimization in equation (6.6)

Input: Original data $\mathbf{X}$, tuning parameters τ, ρ, and cluster number c;
Output: $\mathbf{W}$;
Initialize $\mathbf{V}$ and $\mathbf{Z}$ as random matrix;
repeat
  Update $\mathbf{W}$ using equation (6.8)
  Update $\mathbf{Z}$ following the GPI framework [112]
  Update $\mathbf{V}$ using equation (6.10);
until *convergence of equation* (6.6);

A. **Update $\mathbf{W}$ by fixing $\mathbf{U}, \mathbf{S}, \gamma, \mathbf{Y}, \mathbf{R}$:**

When the other variables are fixed, the objective function in equation (6.4) simplifies to:

$$\begin{aligned}\min_{\mathbf{W}} \operatorname{Tr}(\mathbf{W}^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{W}) - \rho \operatorname{Tr}(\mathbf{W}^T\mathbf{X}\mathbf{U}\mathbf{U}^T\mathbf{X}^T\mathbf{W}) \\ + \tau \operatorname{Tr}(\mathbf{W}^T\mathbf{O}\mathbf{W}) + \|\mathbf{Y} - \mathbf{X}\mathbf{W}\mathbf{R}\|_F^2 \quad \text{s.t. } \mathbf{W}^T\mathbf{W} = \mathbf{I}\end{aligned} \tag{6.5}$$

We employ the framework of Alternating Direction Method of Multipliers (ADMM) [113] to divide the problem of equation (6.5) into sub-problems by setting $\mathbf{X}\mathbf{W} = \mathbf{Z}$. The corresponding pseudo is summarized in Algorithm 8.

$$\begin{aligned}\min_{\mathbf{Z},\mathbf{W},\mathbf{V}} \operatorname{Tr}(\mathbf{Z}^T\mathbf{L}\mathbf{Z}) - \rho \operatorname{Tr}(\mathbf{Z}^T\mathbf{U}\mathbf{U}^T\mathbf{Z}) + \tau \operatorname{Tr}(\mathbf{W}^T\mathbf{O}\mathbf{W}) \\ + \|\mathbf{Y} - \mathbf{Z}\mathbf{R}\|_F^2 + \beta\|\mathbf{Z} - \mathbf{X}\mathbf{W} + \mathbf{V}\|_F^2 \quad \text{s.t. } \mathbf{Z}^T\mathbf{Z} = \mathbf{I}\end{aligned} \tag{6.6}$$

Equation (6.6) can be interpreted as iteratively solving the following sub-problems:

$$\begin{cases}\mathbf{W}^{(t+1)} : \min_{\mathbf{W}^{(t)}} \tau \operatorname{Tr}(\mathbf{W}^{(t)^T}\mathbf{O}\mathbf{W}^{(t)}) + \beta\|\mathbf{Z}^{(t)} - \mathbf{X}\mathbf{W}^{(t)} + \mathbf{V}^{(t)}\|_F^2 \\ \mathbf{Z}^{(t+1)} : \min_{\mathbf{Z}^{(t)}} \operatorname{Tr}(\mathbf{Z}^{(t)^T}\mathbf{L}\mathbf{Z}^{(t)}) - \rho \operatorname{Tr}(\mathbf{Z}^{(t)^T}\mathbf{U}\mathbf{U}^T\mathbf{Z}^{(t)}) \\ \quad + \|\mathbf{Y} - \mathbf{Z}^{(t)}\mathbf{R}\|_F^2 + \beta\|\mathbf{Z}^{(t)} - \mathbf{X}\mathbf{W}^{(t+1)} + \mathbf{V}^{(t)}\|_F^2, \quad \text{s.t.} \quad \mathbf{Z}^{(t)^T}\mathbf{Z}^{(t)} = \mathbf{I}, \\ \mathbf{V}^{(t+1)} = \mathbf{V}^{(t)} + (\mathbf{Z}^{(t+1)} - \mathbf{X}\mathbf{W}^{(t+1)})\end{cases} \tag{6.7}$$

– **Update $W^{(t+1)}$:** By fixing $\mathbf{Z}^{(t)}$ and $\mathbf{V}^{(t)}$, and setting the derivative with respect to $\mathbf{W}$ equal to zero, we obtain the closed-form solution for $\mathbf{W}$ as follows:

$$\mathbf{W}^{(t+1)} = \beta(\tau\mathbf{O} + \beta\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T(\mathbf{Z}^{(t)} + \mathbf{V}^{(t)}) \tag{6.8}$$

– **Update $\mathbf{Z}^{(t+1)}$:** By fixing $\mathbf{V}^{(t)}$ and $\mathbf{W}^{(t+1)}$, we utilize the Generalized Power Iteration (GPI) method [112] with the orthogonal constraint, we have:

$$\min_{\mathbf{Z}^T\mathbf{Z}=\mathbf{I}} \mathrm{Tr}({\mathbf{Z}^{(t)}}^T\mathbf{A}\mathbf{Z}^{(t)} - 2{\mathbf{Z}^{(t)}}^T\mathbf{B}) \tag{6.9}$$

Where $\mathbf{A} = (\mathbf{L} - \rho\mathbf{U}\mathbf{U}^T + \beta\mathbf{I})$ and $\mathbf{B} = (\beta\mathbf{X}\mathbf{W}^{(t+1)} + \mathbf{Y}\mathbf{R}^T - \beta\mathbf{V}^{(t)})$.

– **Update $\mathbf{V}^{(t+1)}$:** By fixing $\mathbf{Z}^{(t+1)}$ and $\mathbf{W}^{(t+1)}$, obtaining $\mathbf{V}$ becomes straightforward:

$$\mathbf{V}^{(t+1)} = \mathbf{V}^{(t)} + (\mathbf{Z}^{(t+1)} + \mathbf{X}\mathbf{W}^{(t+1)}) \tag{6.10}$$

B. **Update U by fixing W, S, γ, Y and R:**

When the other variables are fixed, the optimization of $\mathbf{U}$ becomes straightforward as:

$$\mathbf{U} = \underset{\mathbf{U}^T\mathbf{U}=\mathbf{I}}{\mathrm{argmin}} \; -\mathrm{Tr}(\mathbf{W}^T\mathbf{X}\mathbf{U}\mathbf{U}^T\mathbf{X}^T\mathbf{W}) = \underset{\mathbf{U}^T\mathbf{U}=\mathbf{I}}{\mathrm{argmax}} \; \mathrm{Tr}(\mathbf{U}^T\mathbf{X}^T\mathbf{W}\mathbf{W}^T\mathbf{X}\mathbf{U}). \tag{6.11}$$

The optimal solution for $\mathbf{U}$ is given by the eigenvectors of $(\mathbf{X}^T\mathbf{W}\mathbf{W}^T\mathbf{X})$ associated with the c largest eigenvalues.

C. **Update S by fixing W, U, γ, Y and R:**

To guarantee that $\mathbf{S}$ is semi-positive definite, we impose the constraint $0 \leq s_i \leq 1$. The objective function with respect to $\mathbf{S}$ is expressed in vector form as:

$$\min_{\mathbf{S}} \sum_{i,j=1}^{n} \left(\|\mathbf{W}^T x_i - \mathbf{W}^T x_j\|_2^2 s_{ij} + \gamma s_{ij}^2\right) \quad \text{s.t. } \forall i, \quad s_i^T 1 = \rho, \; 0 \leq s_i \leq 1. \tag{6.12}$$

Since the problem in equation (6.12) is independent for each i, it can be rewritten in

vector form as:

$$\min_{s_i^T 1=\rho, 0\leq s_i\leq 1} \left\| s_i + \frac{1}{2\gamma} d_i \right\|_2^2 \tag{6.13}$$

Where $d_{i,j} = \|\mathbf{W}^T x_i$ - $\mathbf{W}^T x_j\|_2^2$. The Lagrangian function of equation (6.13) as:

$$\mathcal{L}(s_i, \sigma, \zeta_i) = \frac{1}{2} \left\| s_i + \frac{d_i}{2\gamma} \right\|_2^2 - \sigma(s_i^T 1 - \rho) - \zeta_i^T s_i \tag{6.14}$$

where σ and $\zeta_i \geq 0$ are the Lagrangian multipliers. According to the KKT conditions, the optimal solution s_{ij} is:

$$s_{ij} = \left(-\frac{d_{ij}}{2\gamma_i} + \sigma \right)_+ \tag{6.15}$$

The regularization parameter γ is obtained as follows:

$$\gamma = \frac{1}{n} \sum_{i=1}^{n} \left(\frac{k}{2\rho} d_{i,k+1} - \frac{1}{2\rho} \sum_{j=1}^{k} d_{ij} \right) \tag{6.16}$$

Where k denotes the number of nearest neighbors to x_i that are considered for constructing the sparse similarity matrix $\mathbf{S}$. Refer to [108] for more detail. Thus, the optimal s_{ij} will be:

$$s_{ij} = \left(\frac{\rho d_{i,k+1} - \rho d_{ij}}{k d_{i,k+1} - \sum_{j=1}^{k} d_{ij}} \right)_+ \tag{6.17}$$

D. **Update R by fixing W, S, U, γ and Y:**

When the other variables are fixed, the optimization problem with respect to $\mathbf{R}$ is:

$$\min_{\mathbf{R}^T\mathbf{R}=\mathbf{I}} \|\mathbf{Y} - \mathbf{XWR}\|_F^2 \tag{6.18}$$

According to the literature [43], $\mathbf{R}$ has the following closed-form solution:

$$\mathbf{R} = \mathbf{AB}^T \tag{6.19}$$

Where $\mathbf{A}$ and $\mathbf{B}$ are left and right singular vectors of the Singular Value Decomposition

(SVD) with respect to $\mathbf{W}^T\mathbf{X}^T\mathbf{Y}$, i.e., $\mathbf{W}^T\mathbf{X}^T\mathbf{Y} = \mathbf{A}\Sigma\mathbf{B}^T$

E. **Update Y by fixing W, S, γ, U, and R:**

When the other variables are fixed, the objective function for $\mathbf{Y}$ becomes:

$$\min_{\mathbf{Y}} \|\mathbf{XWR} - \mathbf{Y}\|_F^2 + \delta \mathrm{Tr}(\mathbf{Y}^T\mathbf{F}\mathbf{F}^T\mathbf{Y}) \tag{6.20}$$

The problem is non-smooth and optimizing the discrete matrix $\mathbf{Y}$ is difficult. Using the Augmented Lagrange Method (ALM) [114], we replace $\mathbf{Y}$ with a continuous matrix $\mathbf{H}$ in the GFT. We then apply ADMM to convert the problem into an equality-constrained optimization, solving iteratively by alternating between updating $\mathbf{Y}$ and $\mathbf{H}$. Thus, equation (6.20) is reformulated as:

$$\min_{\mathbf{Y},\,\mathbf{H}} \|\mathbf{XWR} - \mathbf{Y}\|_F^2 + \delta \mathrm{Tr}(\mathbf{H}^T\mathbf{F}\mathbf{F}^T\mathbf{H}) + \frac{\xi}{2}\|\mathbf{Y} - \mathbf{H} + \frac{1}{\xi}\mu\|_F^2 \tag{6.21}$$

Here, ξ is a positive scalar that gradually increases, and μ is the Lagrange multiplier estimate, improving accuracy with each iteration. While $\mathbf{Y}$ is fixed, the derivative of the Lagrangian function of equation (6.21) with respect to $\mathbf{H}$ is set to zero. Thus, we have:

$$\mathbf{H} = (\xi\mathbf{I}_n + 2\delta\mathbf{F}\mathbf{F}^T)^{-1}(\xi\mathbf{Y} + \mu) \tag{6.22}$$

While $\mathbf{H}$ is fixed, the equation (6.21) will become:

$$\min_{\mathbf{Y}} \|\mathbf{Y} - \mathbf{K}\|_F^2 + \omega, \tag{6.23}$$

where $\mathbf{K} = \frac{2}{2+\xi}(\mathbf{XWR}) + \frac{1}{2+\xi}(\xi\mathbf{H} - \mu)$, and ω is a constant. Equation (6.23) is independent for each i, Thus, we can solve it separately for each $\mathbf{y}_i$.

$$\min_{\mathbf{y}_i} \sum_{i=1}^{n}\sum_{\ell=1}^{c}(y_{i\ell} - v_{i\ell})^2, \quad \text{s.t.} \quad y_{i\ell} \in \{0,1\}, \quad \sum_{\ell=1}^{c} y_{i\ell} = 1. \tag{6.24}$$

Thus, optimal solution of y_i is as follows:

$$y_{i\ell} = \begin{cases} 1, & \text{if } \ell = \arg\max_{\ell' \in [1,c]} v_{i\ell'} \\ 0, & \text{otherwise} \end{cases} \tag{6.25}$$

Algorithm 9: Steps for the optimization in equation (6.4)

Input: Original data $\mathbf{X} \in \mathbb{R}^{n \times d}$, cluster number c, hyper-parameters τ, ρ, δ, and $\theta \in (1, 2)$;

Output: Clustering result $\mathbf{Y}$;

Randomly initialize $\mathbf{R}$, $\mathbf{W}$, $\mathbf{Y}$, $\mathbf{S}$ matrices and $\mathbf{O} = \mathbf{I}$;
Initialize random binary index matrix $\mathbf{M} \in \{0, 1\}^{n \times c}$;
Initialize $\xi > 0$ and $\mu = 0 \in \mathbb{R}^{n \times c}$;
repeat
  Update $\mathbf{W}$ via Algorithm 8;
  Update the diagonal matrix $\mathbf{O}$ such as $o_{ii} = \frac{1}{2\|\mathbf{w}_i\|_2}$;
  Update $\mathbf{U}$ using equation (6.11);
  Update $\mathbf{S}$ using equation (6.17);
  Determine γ using equation (6.16);
  Update $\mathbf{R}$ via equation (6.19);
  Update $\mathbf{H}$ via $\mathbf{H}^{t+1} = (\xi^t \mathbf{I}_n + 2\delta \mathbf{F}\mathbf{F}^T)^{-1}(\xi^t \mathbf{Y}^t + \mu^t)$;
  Update $\mathbf{Y}$ via equation (6.25);
  Update μ via $\mu^{t+1} = \mu^t + \xi^t(\mathbf{Y} - \mathbf{H})$;
  Update ξ via $\xi^{t+1} = \theta \xi^t$;
until *convergence or the number of iterations* $= T_1$ *with respect to equation* (6.4);

6.3.3 Experiments and Results

We set the number of clusters c corresponding to the actual number of classes in each dataset. The Lagrange multiplier coefficient ξ is often set to 0.1 or 1, whereas the θ is chosen somewhat more than 1 for ALM optimization [114]. We used $\xi = 0.1$ and $\theta = 1.005$ for all datasets in our experiments. k is set to $\sqrt{n}$ for k-nearest neighbors in corresponding methods. We tune the parameters and present the results based on the optimal parameter setting of $\tau, \rho, \delta \in \{10^{-3}, 10^{-2}, \ldots, 10^{3}\}$. In experiments, we first evaluated the OSSRC method on all possible combinations of τ, ρ. We selected the best combination of τ and ρ and used the same combination with all the values of δ in the FOSSRC model.

Table 6.1: *Balance improvement(%) of the FOSSRC method over baseline methods*

Datasets	Existing Methods			
	SC	FSC	UFSC	OSSRC
FacebookNet ($c = 2$)	100	95.12	35.59	100
DrugNet ($c = 6$)	156.52	118.51	59.45	136

The performance of the FOSSRC model is evaluated using two widely used benchmark datasets from [25, 33]: FacebookNet[1] and DrugNet[2]. Since these datasets include ground-truth graphs and the FOSSRC algorithm relies on adaptive graph learning, the data are generated according to the underlying ground-truth graph, following the procedure described in [33]. Furthermore, the performance of the FOSSRC model is assessed on real-world datasets collected from the UCI Machine Learning repository[3], including Bank, ILPD, German Credit, and Hepatitis.

6.3.3.1 Baselines and Evaluation Metrics:

The baseline methods for comparison include SC, Fair Spectral Clustering (FSC) [25], Unified Fair Spectral Clustering (UFSC) [33], and OSSRC (i.e., Eq. (6.1)) without fairness. First, we compare the FOSSRC model with the above methods on two benchmark datasets. Additionally, we explored other real-world datasets to test the effectiveness of the FOSSRC method by comparing it with SC and OSSRC without fairness.

Based on the literature, we use standard metrics, including Acc, NMI, to assess clustering performance and use balance to assess the fairness of clustering as in equation (2.2) (averaged over all clusters).

6.3.3.2 Results

Since we do not have class labels in the benchmark data sets (i.e., FacebookNet and DrugNet), we evaluate the FOSSRC method using the balance metric. Figure 6.2 demonstrates that the FOSSRC method significantly improves balance by 35.59% and 59.45% on

[1] http://www.sociopatterns.org/datasets/high-school-contact-and-friendship-networks/
[2] https://sites.google.com/site/ucinetsoftware/datasets/covert-networks/drugnet
[3] https://archive.ics.uci.edu/datasets

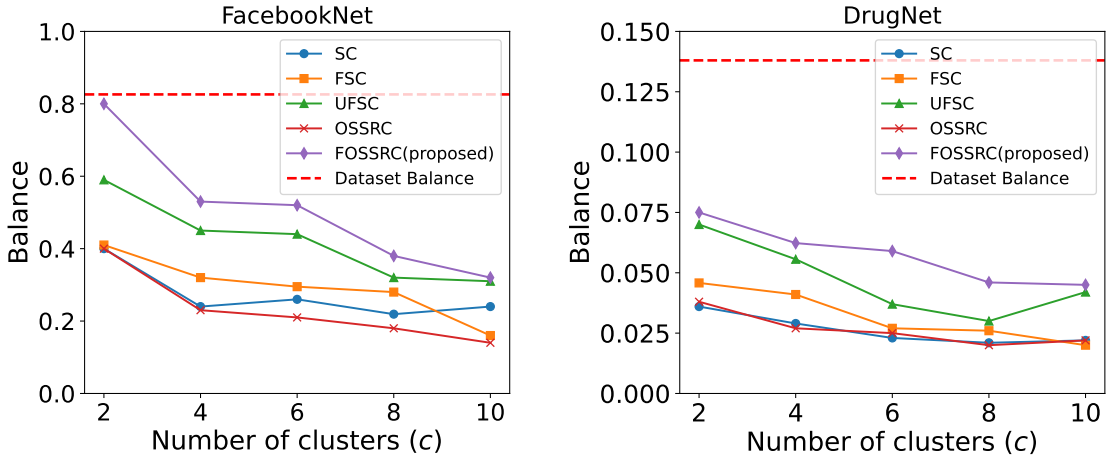


Figure 6.2: *Balance comparison of the FOSSRC(proposed) method against baseline methods with varying numbers of clusters on two benchmark datasets.*

Table 6.2: *Clustering results on additional real-world datasets*

Bank				**ILPD**			
Method	Acc	NMI	Balance	Method	Acc	NMI	Balance
SC	0.613	0.0025	0.38	SC	0.594	**0.031**	0.180
OSSRC	**0.678**	**0.0031**	0.403	OSSRC	**0.620**	0.001	0.259
FOSSRC	0.633	0.0017	**0.426**	FOSSRC	0.597	0.002	**0.315**
German				**Hepatitis**			
Method	Acc	NMI	Balance	Method	Acc	NMI	Balance
SC	0.54	0.0052	0.21	SC	0.726	**0.095**	0.10
OSSRC	**0.598**	**0.0054**	0.417	OSSRC	**0.819**	0.013	0.096
FOSSRC	0.580	0.0046	**0.443**	FOSSRC	0.764	0.003	**0.142**

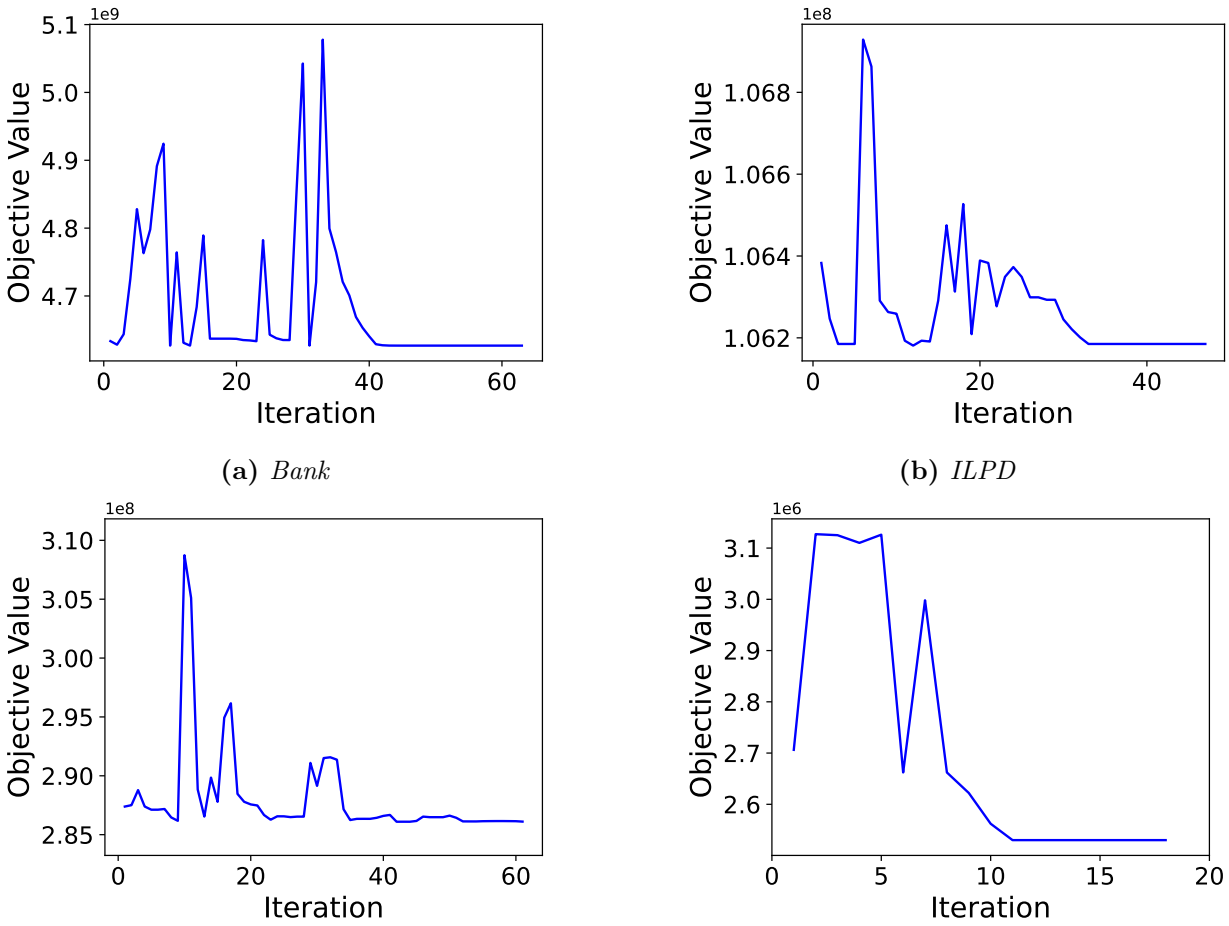
Dataset balance: Bank (0.427), ILPD (0.318), German (0.449), Hepatitis (0.159).

FacebookNet ($c = 2$) and DrugNet($c = 6$) datasets compared to the UFSC model. In addition to the results in Figure 6.2, Table 6.1 presents the percentage improvement of the proposed FOSSRC method compared to the baseline methods. Thus, these results demonstrate the superiority of the FOSSRC method in achieving a higher balance.

To further validate the generalizability of the proposed FOSSRC approach, we evaluate it on four additional real-world datasets summarized in Table 6.2. Acc, NMI, and balance metrics are calculated for SC, OSSRC, and FOSSRC, with the best results highlighted in bold. The proposed FOSSRC model takes advantage of the GFT integrated with adaptive graph learning, achieving superior clustering performance while maintaining fairness. Notably, FOSSRC outperforms the other methods with respect to fairness metrics(i.e., balance) across all the datasets, improving balance by 5.7% in the Bank, 21.62% in ILPD,

6.23% in German, and 57.77% in the Hepatitis dataset. Moreover, FOSSRC achieves higher Acc than SC across all datasets due to adaptive graph learning, though a slight trade-off in NMI is observed while enhancing fairness. OSSRC outperforms SC in terms of balance and Acc because it employs adaptive graph learning in a unified framework rather than a fixed graph. We observed increased NMI in OSSRC on the Bank and German datasets, but there is a trade-off in NMI in the ILPD and hepatitis datasets compared to SC. Overall, the proposed FOSSRC method achieves fair clusters and simultaneously improves accuracy compared to the standard SC.

6.3.3.3 Convergence Analysis

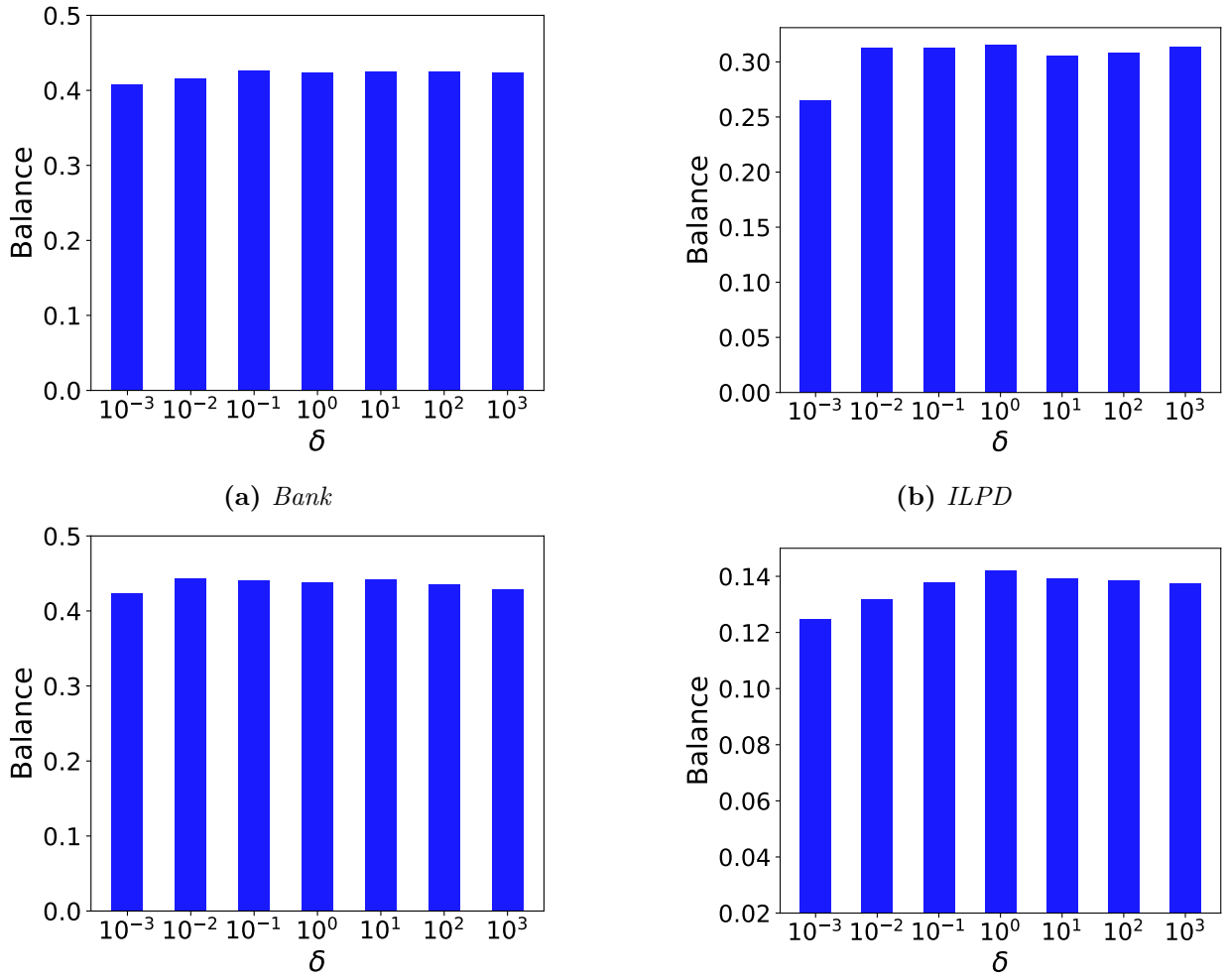


(a) *Bank* **(b)** *ILPD*

(c) *German* **(d)** *Hepatitis*

Figure 6.3: *Convergence analysis of the proposed Algorithm 9.*

Figure 6.3 shows the changes in objective function values obtained by the proposed



(a) *Bank* **(b)** *ILPD*

(c) *German* **(d)** *Hepatitis*

Figure 6.4: *Parameter sensitivity analysis illustrating the effect of varying δ.*

Algorithm 9 for each iteration. The stop condition of iterations is set as $|obj(t)-obj(t-1)| \leq 10^{-3}$, where $obj(t)$ is the objective function value at the $t-th$ iteration. The convergence plot in Figure 6.3 shows that some parts of the convergence curve exhibit fluctuations and lack smoothness. This is due to the discrete nature of the cluster indicator matrix $\mathbf{Y}$.

6.3.3.4 Parameter Sensitivity

Eq. (6.4) has four tuning parameters, that is, τ, γ, ρ and δ. We directly compute the value of γ as given in [108]. To analyze the influence of other parameters (τ, ρ and δ) we evaluate over the range: $\{10^{-3}, 10^{-2}, \ldots, 10^{3}\}$. We perform clustering in the OSSRC model with varying parameters τ and ρ, selecting their optimal combination. This optimal setup is then used

with δ in the FOSSRC model for further analysis of fairness. Figure 6.4 illustrates the balance in all datasets for different values of δ in the FOSSRC model. The best balance value for the Bank dataset is 0.426, corresponding to $\tau = 0.001, \rho = 10, \delta = 0.1$. For the ILPD dataset, the balance value is 0.315, corresponding to $\tau = 0.01, \rho = 0.001, \delta = 1$. For the German dataset, the balance value is 0.443, which corresponds to $\tau = 1000, \rho = 100, \delta = 0.01$. For the Hepatitis dataset, the balance value is 0.142, corresponding to $\tau = 100, \rho = 0.01, \delta = 1$.

6.4 Summary

This final contributory chapter introduced a novel GFT designed to incorporate fairness directly into the OSSRC framework. The chapter presented a new perspective and explanation of fairness in one-step clustering criteria, providing a theoretical analysis that demonstrates the role and impact of the proposed GFT. Based on these insights, the GFT was integrated with the objective function of the OSSRC, leading to the FOSSRC model. An effective optimization algorithm is formulated to solve the FOSSRC model by alternately updating the variables corresponding to each sub-task. Comprehensive experiments show that FOSSRC achieves fair clusters and outperforms state-of-the-art models (i.e., FSC and UFSC). In addition, FOSSRC achieves better clustering accuracy than SC, confirming its effectiveness in simultaneously ensuring fairness and maintaining clustering quality.

Furthermore, while the fairness-aware SC frameworks presented in this dissertation primarily demonstrate effectiveness using single sensitive attributes, they provide a flexible foundation for addressing more complex demographic structures. Specifically, the proposed models can be extended to achieve intersectional fairness by adopting a composite attribute approach. By defining protected groups as the Cartesian product of multiple sensitive dimensions (e.g., the intersection of race and gender). Furthermore, the frameworks can support multi-attribute objectives by reformulating the optimization criterion to include independent GFT penalties for each sensitive attribute. In this multi-penalty structure, distinct fairness constraints are summed and weighted individually within the objective

function, allowing practitioners to prioritize different demographic dimensions according to specific legal or ethical requirements. This pathway extensibility ensures that the proposed spectral optimization remains a robust and scalable solution for the multi-dimensional demographic landscapes encountered in real-world, high-impact decision systems.

The next chapter concludes the dissertation by summarizing the key contributions and outlining potential directions for future research.

"Obstacles are those frightful things you see when you take your eyes off your goal."

~Henry Ford

7

Conclusions and Future Perspectives

This chapter concludes the dissertation by summarizing the key contributions made throughout the research and discussing potential directions for future research. It highlights how the proposed fairness-aware clustering methods advance both fairness and clustering performance. The chapter is organized into two sections: a summary of contributions and a discussion of future work.

7.1 Summary of Contributions

This thesis addressed the problem of fairness in SC, with a particular focus on integrating fairness into advanced and improved variants of SC, such as constrained SC and OSSRC. Existing fair SC methods often compromise clustering quality when fairness is imposed; in

contrast, advanced SC methods, such as constrained SC and OSSRC, essentially prioritize structural performance over fairness. Motivated by these shortcomings, this dissertation developed fairness-aware variants of advanced SC methods that simultaneously improve fairness and preserve clustering quality. The research focused on two primary directions: (i) enhancing fairness within constrained SC, and (ii) integrating fairness into one-step SC. All the proposed works are extensively validated theoretically and experimentally. A consolidated summary of the contributions is provided in Table 7.1, which summarizes how fairness has been incorporated at different levels: objective-level fairness, graph-level preprocessing, and unified one-step fairness modeling.

The dissertation contains four main contributions:

7.1.1 Fairness-integrated Constrained Spectral Clustering via Laplacian Adjustment

The first contribution of this dissertation introduced a fairness-aware extension of cSC by modifying the Laplacian formulation to incorporate demographic group information with ML and CL constraints. Fairness is defined in ML and CL constraints through a specially designed fair constraint matrix. A fair constraint matrix is designed to introduce fairness in the constraint set, which encodes group-level balance requirements. This matrix, together with a balance constraint, is integrated into the SC objective by adjusting the graph Laplacian. Experimental results and analysis demonstrated the effectiveness of the method, ensuring fair representation of demographic groups while preserving clustering quality.

7.1.2 Constrained Spectral Clustering with Pairwise Constraints

The second contribution extended fairness into the pairwise constraint representation itself. A refined PCSC framework was developed in which demographic information is embedded within the structure of ML and CL constraints. This was achieved by formulating a fairness-driven linear constraint that aligns the SC objective. As a result, the obtained clusters reflect a distribution of sensitive groups that aligns with the given pairwise con-

straints. Experimental evaluations demonstrated substantial improvements in fairness while preserving clustering quality.

7.1.3 Fair and Constraints-respecting Graph Construction for Spectral Clustering

The third contribution proposed a novel preprocessing method that integrates fairness and pairwise constraints at the graph construction stage to address the limitations of fairness integration at the optimization level. A fair constrained kNN graph is constructed, ensuring neighborhood fairness, where each group is proportionally represented within local neighborhoods, while simultaneously enforcing ML and CL constraints at the graph level. By embedding fairness and pairwise constraints directly into the similarity graph itself, the method produces an unbiased structural representation prior to clustering. SC applied to this fair constrained graph subsequently yields fair clusters that preserve clustering quality while satisfying the specified pairwise constraints.

7.1.4 Fairness-aware One-Step Spectral Rotation Clustering

This fourth contribution integrates fairness within the one-step SC framework. A novel group fairness term is introduced and integrated with the OSSRC formulation to reduce demographic imbalances in the resulting clusters while preserving accuracy. By combining the three stages of traditional SC, the proposed FOSSRC model jointly performs adaptive graph learning, fairness enforcement, and clustering in a unified framework. An efficient alternating optimization algorithm is used to solve the model. Experimental results showed that FOSSRC produces fair clusters and outperforms fairness-driven baselines such as FSC and UFSC.

Across all four contributions, this dissertation demonstrates that fairness can be effectively integrated into advanced SC methods, enabling the generation of clusters that are fair while preserving clustering quality.

Table 7.1: *Summary of the main contributions presented in this dissertation*

Method	Fairness Integration Level	Objective	Outcome
Fair Constrained SC (Contribution 1 & 2)	In-processing (objective modification)	Adds fairness + pairwise constraints	Maintains fairness and clustering quality
Graph-Level Fair Constrained SC (Contribution 3)	Preprocessing (graph-level)	Enforces fairness + pairwise constraints in kNN graph	Improves fairness consistency before clustering
Fair One-Step Spectral Rotation SC (Contribution 4)	Unified model (one-step)	Adds fairness to one-step objective	Ensures fairness, efficiency, and improved performance

7.2 Practical Deployment and Applicability

Transitioning the fairness-aware SC frameworks developed in this dissertation from theoretical models to practical, real-world deployment requires a thorough consideration of system applicability and production constraints. The utility of these frameworks spans across several critical, high-impact domains, such as healthcare analytics for equitable resource distribution, financial decision systems for unbiased credit profiling, educational data analysis, personalized recommendation engines, and social network analysis, where group-level bias mitigation is increasingly mandatory. From a deployment perspective, the proposed methods can be integrated into existing graph-based clustering pipelines with relatively limited modifications, since fairness interventions are incorporated through graph construction, fairness regularization, or constrained optimization. The experimental results presented in this dissertation demonstrate that fairness can be effectively integrated into advanced SC frameworks while preserving competitive clustering quality, thereby supporting the practical applicability of the proposed methods in real-world systems.

7.3 Scope for Future Work

While this dissertation makes significant contributions to the development of fair and improved SC methods, there are still several future directions that could be explored to further

enhance fairness and clustering performance through more advanced and optimized methods. This initiative marks the beginning of such advancements. This can be further extended to more advanced versions.

7.3.1 Scalability Considerations

Although the proposed fairness-aware SC frameworks demonstrate strong clustering and fairness performance on benchmark datasets, scalability remains an important challenge for very large-scale data. Similar to traditional SC methods, the computational bottleneck primarily arises from similarity graph construction and eigen-decomposition of the graph Laplacian. Nevertheless, the proposed methods preserve the sparse graph structure commonly used in spectral clustering, particularly through the use of k-nearest neighbor graphs and sparse constraint representations. This reduces computational and memory requirements compared to fully connected graph formulations. For large-scale applications involving millions of samples, future research may explore scalable approximations such as landmark-based spectral clustering, sparse eigensolvers, distributed graph processing, and GPU-accelerated optimization frameworks. These extensions would further improve the practical applicability of fairness-aware SC in large real-world systems.

7.3.2 Extension to Multiple Sensitive Attributes

The methods introduced in this dissertation primarily focus on fairness defined over a single sensitive attribute. Extending the core frameworks to handle multiple sensitive attributes simultaneously represents an important and necessary direction for future research. Generally, two strategies can be applied across all the proposed methods: First, the intersectional Approach defines protected groups as the Cartesian product of multiple attributes (e.g., combining 'Age' and 'Race' into a single multi-valued variable). This ensures that fairness interventions protect individuals at the intersection of multiple identities, preventing bias against specific combined subgroups. Second, the multi-objective approach formulates independent fairness terms for each distinct attribute, integrating them into a single objective

function via a weighted summation. This allows for the simultaneous enforcement of distinct fairness constraints.

7.3.3 Fairness in Online and Incremental Spectral Clustering

Many real-world systems operate in dynamic environments where data evolves over time. Incorporating fairness into online or incremental SC could significantly broaden the practical applicability of fairness-aware clustering, particularly in domains such as evolving social networks, recommendation systems, and real-time decision-making tools.

7.3.4 Unified Framework for Constraints, Fairness, and Graph Learning

Each contribution in this dissertation addressed fairness at a specific stage. The next step is to develop a fully unified model that simultaneously incorporates pairwise constraints, fairness considerations, graph learning, and clustering within a joint optimization framework. Such a comprehensive approach would further reduce information loss between stages, mitigate sub-optimal intermediate results, and potentially produce more accurate and robust clustering outcomes.

7.3.5 On the Availability of Sensitive Attributes

The proposed fairness-aware SC frameworks assume the availability of sensitive attributes during clustering. This assumption is a well-established paradigm in fairness-aware machine learning literature because fairness objectives, constraints, and evaluation metrics necessitate access to demographic or protected-group information. Furthermore, the use of such attributes is standard practice across foundational benchmarks, such as Adult, COMPAS, and German Credit datasets, which explicitly provide sensitive attributes for evaluating algorithmic bias and fairness performance.

In many practical domains, including healthcare analytics, educational systems, and

census-based studies, sensitive information is often accessible under strict ethical and privacy guidelines. Indeed, the availability of such data is essential for the initial auditing phase of any fairness intervention, as the absence of protected-group information makes the measurement and mitigation of disparities significantly more challenging. This requirement highlights a fairness-privacy paradox: while privacy regulations may restrict the collection of sensitive data, the absence of such data prevents the empirical verification of fairness.

Therefore, while the current assumption provides a necessary baseline for developing and benchmarking fairness-aware clustering methods, extending these approaches to settings with missing or unavailable sensitive attributes remains an important direction for future research. Potential extensions include fairness under unawareness [115] and the use of proxy variables correlated with protected groups. Such advancements would ensure that fairness-aware spectral clustering remains robust even in environments where explicit demographic information is restricted.

7.3.6 Privacy Considerations

The integration of sensitive attributes into clustering algorithms necessitates a robust privacy framework, particularly when data is handled in distributed or single-node environments. In distributed settings, directly sharing sensitive demographic information may violate privacy regulations. Future extensions of the proposed frameworks may therefore incorporate privacy-preserving mechanisms, such as federated learning, differential privacy, or encrypted graph learning. These approaches would allow fairness constraints to be optimized collaboratively while keeping sensitive data localized. However, this introduces an important fairness–privacy trade-off. Fairness evaluation and mitigation often require access to sensitive attribute information, whereas privacy-preserving systems seek to minimize exposure of such information. Designing fairness-aware spectral clustering frameworks that effectively balance demographic fairness, clustering quality, and privacy preservation remains an important direction for future research.

References

[1] A. Chouldechova, "Fair prediction with disparate impact: A study of bias in recidivism prediction instruments," *Big data*, vol. 5, no. 2, pp. 153–163, 2017.

[2] J. Angwin, J. Larson, S. Mattu, and L. Kirchner, "Machine bias," in *Ethics of data and analytics*. Auerbach Publications, 2022, pp. 254–264. [Online]. Available: https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing

[3] J. Dastin, "Amazon scraps secret ai recruiting tool that showed bias against women," in *Ethics of data and analytics*. Auerbach Publications, 2022, pp. 296–299.

[4] A. Datta, M. C. Tschantz, and A. Datta, "Automated experiments on ad privacy settings," *Proceedings on Privacy Enhancing Technologies*, vol. 2015, no. 1, pp. 92–112, 2015.

[5] T. Simonite, "Probing the dark side of google's ad-targeting system," *MIT Technology Review*, July 2015, accessed: 2019-07-31. [Online]. Available: https://www.technologyreview.com/s/539021/probing-the-dark-side-of-googles-ad-targeting-system/

[6] M. B. Zafar, I. Valera, M. Gomez Rodriguez, and K. P. Gummadi, "Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment," in *Proceedings of the 26th international conference on world wide web*, 2017, pp. 1171–1180.

[7] R. Zemel, Y. Wu, K. Swersky, T. Pitassi, and C. Dwork, "Learning fair representations," in *International conference on machine learning*. PMLR, 2013, pp. 325–333.

[8] R. Berk, H. Heidari, S. Jabbari, M. Kearns, and A. Roth, "Fairness in criminal justice risk assessments: The state of the art," *Sociological Methods & Research*, vol. 50, no. 1, pp. 3–44, 2021.

[9] S. A. Friedler, C. Scheidegger, S. Venkatasubramanian, S. Choudhary, E. P. Hamilton, and D. Roth, "A comparative study of fairness-enhancing interventions in machine learning," in *Proceedings of the conference on fairness, accountability, and transparency*, 2019, pp. 329–338.

[10] K. Holstein, J. Wortman Vaughan, H. Daumé III, M. Dudik, and H. Wallach, "Improving fairness in machine learning systems: What do industry practitioners need?" in *Proceedings of the 2019 CHI conference on human factors in computing systems*, 2019, pp. 1–16.

[11] S. Barocas, M. Hardt, and A. Narayanan, *Fairness and machine learning: Limitations and opportunities*. MIT press, 2023.

[12] M. Feldman, S. A. Friedler, J. Moeller, C. Scheidegger, and S. Venkatasubramanian, "Certifying and removing disparate impact," in *proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, 2015, pp. 259–268.

[13] A. Agarwal, A. Beygelzimer, M. Dudík, J. Langford, and H. Wallach, "A reductions approach to fair classification," in *International conference on machine learning*. PMLR, 2018, pp. 60–69.

[14] B. Woodworth, S. Gunasekar, M. I. Ohannessian, and N. Srebro, "Learning non-discriminatory predictors," in *Conference on learning theory*. PMLR, 2017, pp. 1920–1953.

[15] S. Corbett-Davies, E. Pierson, A. Feller, S. Goel, and A. Huq, "Algorithmic decision making and the cost of fairness," in *Proceedings of the 23rd acm sigkdd international conference on knowledge discovery and data mining*, 2017, pp. 797–806.

[16] A. K. Menon and R. C. Williamson, "The cost of fairness in binary classification," in *Conference on Fairness, accountability and transparency*. PMLR, 2018, pp. 107–118.

[17] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel, "Fairness through awareness," in *Proceedings of the 3rd innovations in theoretical computer science conference*, 2012, pp. 214–226.

[18] D. Pessach and E. Shmueli, "A review on fairness in machine learning," *ACM Computing Surveys (CSUR)*, vol. 55, no. 3, pp. 1–44, 2022.

[19] R. T. Rabonato and L. Berton, "A systematic review of fairness in machine learning," *AI and Ethics*, vol. 5, no. 3, pp. 1943–1954, 2025.

[20] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan, "A survey on bias and fairness in machine learning," *ACM computing surveys (CSUR)*, vol. 54, no. 6, pp. 1–35, 2021.

[21] A. K. Jain and R. C. Dubes, *Algorithms for clustering data*. Prentice-Hall, Inc., 1988.

[22] A. Chhabra, K. Masalkovaitė, and P. Mohapatra, "An overview of fairness in clustering," *IEEE Access*, vol. 9, pp. 130 698–130 720, 2021.

[23] S. Shaham, A. Hajisafi, M. K. Quan, D. C. Nguyen, B. Krishnamachari, C. Peris, G. Ghinita, C. Shahabi, and P. N. Pathirana, "Privacy and fairness in machine learning: A survey," *IEEE Transactions on Artificial Intelligence*, 2025.

[24] F. Chierichetti, R. Kumar, S. Lattanzi, and S. Vassilvitskii, "Fair clustering through fairlets," *Advances in neural information processing systems*, vol. 30, 2017.

[25] M. Kleindessner, S. Samadi, P. Awasthi, and J. Morgenstern, "Guarantees for spectral clustering with fairness constraints," in *International conference on machine learning*. PMLR, 2019, pp. 3458–3467.

[26] J. Wang, D. Lu, I. Davidson, and Z. Bai, "Scalable spectral clustering with group fairness constraints," in *International conference on artificial intelligence and statistics*. PMLR, 2023, pp. 6613–6629.

[27] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.

[28] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[29] R. Couillet and F. Benaych-Georges, "Kernel spectral clustering of large dimensional data," 2016.

[30] S. Gupta and A. Dukkipati, "Consistency of constrained spectral clustering under graph induced fair planted partitions," *Advances in Neural Information Processing Systems*, vol. 35, pp. 13 527–13 540, 2022.

[31] D. Fu, D. Zhou, R. Maciejewski, A. Croitoru, M. Boyd, and J. He, "Fairness-aware clique-preserving spectral clustering of temporal graphs," in *Proceedings of the ACM Web Conference 2023*, 2023, pp. 3755–3765.

[32] J. Li, Y. Wang, and A. Merchant, *Spectral Normalized-Cut Graph Partitioning with Fairness Constraints*, 09 2023.

[33] X. Zhang and Q. Wang, "A dual laplacian framework with effective graph learning for unified fair spectral clustering," *Neurocomputing*, vol. 601, p. 128210, 2024.

[34] J. Kleinberg, S. Mullainathan, and M. Raghavan, "Inherent trade-offs in the fair determination of risk scores," *arXiv preprint arXiv:1609.05807*, 2016.

[35] Q. Xu, M. desJardins, and K. L. Wagstaff, "Active constrained clustering by examining spectral eigenvectors," in *International Conference on Discovery Science*. Springer, 2005, pp. 294–307.

[36] Z. Lu and M. A. Carreira-Perpinan, "Constrained spectral clustering through affinity propagation," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2008, pp. 1–8.

[37] S. D. Kamvar, D. Klein, and C. D. Manning, "Spectral learning," in *IJCAI*, vol. 3, 2003, pp. 561–566.

[38] X. Wang and I. Davidson, "Flexible constrained spectral clustering," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2010, pp. 563–572.

[39] P. Qian, Y. Jiang, S. Wang, K.-H. Su, J. Wang, L. Hu, and R. F. Muzic, "Affinity and penalty jointly constrained spectral clustering with all-compatibility, flexibility, and robustness," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 5, pp. 1123–1138, 2016.

[40] G. Wacquet, É. P. Caillault, D. Hamad, and P.-A. Hébert, "Constrained spectral embedding for k-way data clustering," *Pattern Recognition Letters*, vol. 34, no. 9, pp. 1009–1017, 2013.

[41] F. Daneshfar, S. Soleymanbaigi, P. Yamini, and M. S. Amini, "A survey on semi-supervised graph clustering," *Engineering Applications of Artificial Intelligence*, vol. 133, p. 108215, 2024.

[42] Z. Kang, C. Peng, Q. Cheng, and Z. Xu, "Unified spectral clustering with optimal graph," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[43] Y. Pang, J. Xie, F. Nie, and X. Li, "Spectral clustering by joint spectral embedding and spectral rotation," *IEEE transactions on cybernetics*, vol. 50, no. 1, pp. 247–258, 2018.

[44] G. Wen, X. Li, Y. Zhu, L. Chen, Q. Luo, and M. Tan, "One-step spectral rotation clustering for imbalanced high-dimensional data," *Information Processing & Management*, vol. 58, no. 1, p. 102388, 2021.

[45] Y. Yang, S. Fumin, H. Zi, and H. T. Shen, "A unified framework for discrete spectral clustering," in *IJCAI International Joint Conference on Artificial Intelligence*, vol. 2016. International Joint Conferences on Artificial Intelligence, 2016, pp. 2273–2279.

[46] X. Zhu, W. He, Y. Li, Y. Yang, S. Zhang, R. Hu, and Y. Zhu, "One-step spectral clustering via dynamically learning affinity matrix and subspace," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31, no. 1, 2017.

[47] P. Zhou, R. Li, Z. Ling, L. Du, and X. Liu, "Fair clustering ensemble with equal cluster capacity," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[48] K. Wagstaff, C. Cardie, S. Rogers, S. Schrödl *et al.*, "Constrained k-means clustering with background knowledge," in *Icml*, vol. 1, 2001, pp. 577–584.

[49] J. Huang, F. Nie, and H. Huang, "Spectral rotation versus k-means in spectral clustering," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 27, no. 1, 2013, pp. 431–437.

[50] X. Zhu, S. Zhang, R. Hu, Y. Zhu *et al.*, "Local and global structure preservation for robust unsupervised spectral feature selection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 3, pp. 517–529, 2017.

[51] X. Zhu, S. Zhang, Y. Li, J. Zhang, L. Yang, and Y. Fang, "Low-rank sparse subspace for spectral clustering," *IEEE Transactions on knowledge and data engineering*, vol. 31, no. 8, pp. 1532–1543, 2018.

[52] A. Fabris, S. Messina, G. Silvello, and G. A. Susto, "Algorithmic fairness datasets: the story so far," *Data Mining and Knowledge Discovery*, vol. 36, no. 6, pp. 2074–2152, 2022.

[53] S. Bera, D. Chakrabarty, N. Flores, and M. Negahbani, "Fair algorithms for clustering," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[54] R. Pan and C. Zhong, "Fairness first clustering: A multi-stage approach for mitigating bias," *Electronics*, vol. 12, no. 13, p. 2969, 2023.

[55] D. Dua and C. Graff, "Uci machine learning repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml

[56] B. Becker and R. Kohavi, "Adult," UCI Machine Learning Repository, 1996, DOI: https://doi.org/10.24432/C5XW20.

[57] R. P. Moro S. and C. P., "Bank Marketing," UCI Machine Learning Repository, 2012, DOI: https://doi.org/10.24432/C5K306.

[58] M. Redmond, "Communities and Crime," UCI Machine Learning Repository, 2002, DOI: https://doi.org/10.24432/C53W3X.

[59] B. Ramana and N. Venkateswarlu, "ILPD (Indian Liver Patient Dataset)," UCI Machine Learning Repository, 2022, DOI: https://doi.org/10.24432/C5D02C.

[60] "Hepatitis," UCI Machine Learning Repository, 1988, DOI: https://doi.org/10.24432/C5Q59J.

[61] I.-C. Yeh, "default of credit card clients," UCI Machine Learning Repository, 2016, DOI: https://doi.org/10.24432/C55S3H.

[62] R. Mastrandrea, J. Fournet, and A. Barrat, "Contact patterns in a high school: a comparison between data collected using wearable sensors, contact diaries and friendship surveys," *PloS one*, vol. 10, no. 9, p. e0136497, 2015.

[63] M. R. Weeks, S. Clair, S. P. Borgatti, K. Radda, and J. J. Schensul, "Social networks of drug users in high-risk sites: Finding the connections," *AIDS and Behaviour*, vol. 6, pp. 193–206, 2002. [Online]. Available: https://sites.google.com/site/sfeverton18/research/cohesion-and-clustering

[64] H. Hofmann, "Statlog (German Credit Data)," UCI Machine Learning Repository, 1994, DOI: https://doi.org/10.24432/C5NC77.

[65] M. Miron, S. Tolan, E. Gómez, and C. Castillo, "Evaluating causes of algorithmic bias in juvenile criminal recidivism," *Artificial intelligence and law*, vol. 29, no. 2, pp. 111–147, 2021.

[66] P. Cortez, "Student Performance," UCI Machine Learning Repository, 2008, DOI: https://doi.org/10.24432/C5TG7T.

[67] L. F. Wightman, "Lsac national longitudinal bar passage study. lsac research report series." 1998.

[68] G. González-Almagro, D. Peralta, E. De Poorter, J.-R. Cano, and S. García, "Semi-supervised constrained clustering: An in-depth overview, ranked taxonomy and future research directions," *arXiv preprint arXiv:2303.00522*, 2023.

[69] T. Le Quy, G. Friege, and E. Ntoutsi, "A review of clustering models in educational data science toward fairness-aware learning," *Educational data science: Essentials, approaches, and tendencies: Proactive education based on empirical big data evidence*, pp. 43–94, 2023.

[70] A. Backurs, P. Indyk, K. Onak, B. Schieber, A. Vakilian, and T. Wagner, "Scalable fair clustering," in *International Conference on Machine Learning*. PMLR, 2019, pp. 405–413.

[71] C. Rösner and M. Schmidt, "Privacy preserving clustering with constraints," *arXiv preprint arXiv:1802.02497*, 2018.

[72] I. M. Ziko, J. Yuan, E. Granger, and I. B. Ayed, "Variational fair clustering," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, 2021, pp. 11 202–11 209.

[73] L. Zheng, Y. Zhu, and J. He, "Fairness-aware multi-view clustering," in *Proceedings of the 2023 SIAM International Conference on Data Mining (SDM)*. SIAM, 2023, pp. 856–864.

[74] A. Anagnostopoulos, L. Becchetti, A. Fazzone, C. Menghini, and C. Schwiegelshohn, "Spectral relaxations and fair densest subgraphs," in *29th ACM Int. Conf. on Inform. & Knowledge Manag.*, 2020, pp. 35–44.

[75] J. Li, Y. Wang, and A. Merchant, "Spectral normalized-cut graph partitioning with fairness constraints," *arXiv preprint arXiv:2307.12065*, 2023.

[76] Z. Yang, H. Zhang, C. Yang, B. Li, X. Zhao, and Y. Long, "Scale fairness on spectral clustering," in *Proceedings of the 36th international conference on scientific and statistical database management*, 2024, pp. 1–9.

[77] ——, "Fair laplace: A unified framework for fair spectral clustering," *Information Processing & Management*, vol. 62, no. 4, p. 104124, 2025.

[78] C. Laclau, C. Largeron, and M. Choudhary, "A survey on fairness for machine learning on graphs," *arXiv preprint arXiv:2205.05396*, 2022.

[79] Y. Dong, J. Ma, S. Wang, C. Chen, and J. Li, "Fairness in graph mining: A survey," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 10, pp. 10 583–10 602, 2023.

[80] K. Wagstaff and C. Cardie, "Clustering with instance-level constraints," *AAAI/IAAI*, vol. 1097, pp. 577–584, 2000.

[81] B. Kulis, S. Basu, I. Dhillon, and R. Mooney, "Semi-supervised graph clustering: a kernel approach," in *Proceedings of the 22nd International Conference on Machine Learning*, 2005, pp. 457–464.

[82] Z. Lu and H. H. Ip, "Constrained spectral clustering via exhaustive and efficient constraint propagation," in *European Conference on Computer Vision*. Springer, 2010, pp. 1–14.

[83] Z. Lu and Y. Peng, "Exhaustive and efficient constraint propagation: A graph-based learning approach and its applications," *International journal of computer vision*, vol. 103, no. 3, pp. 306–325, 2013.

[84] Y. Jia, H. Liu, J. Hou, and S. Kwong, "Semisupervised adaptive symmetric non-negative matrix factorization," *IEEE transactions on cybernetics*, vol. 51, no. 5, pp. 2550–2562, 2020.

[85] Y. Jia, W. Wu, R. Wang, J. Hou, and S. Kwong, "Joint optimization for pairwise constraint propagation," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 7, pp. 3168–3180, 2020.

[86] F. Nie, H. Zhang, R. Wang, and X. Li, "Semi-supervised clustering via pairwise constrained optimal graph," in *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, 2021, pp. 3160–3166.

[87] Y. Li and H. Lu, "Multi-modal constraint propagation via compatible conditional distribution reconstruction," *Neurocomputing*, vol. 426, pp. 185–194, 2021.

[88] S. X. Yu and J. Shi, "Segmentation given partial grouping constraints," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 2, pp. 173–183, 2004.

[89] Z. Li, J. Liu, and X. Tang, "Constrained clustering via spectral regularization," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 421–428.

[90] X. Wang, B. Qian, and I. Davidson, "On constrained spectral clustering and its applications," *Data Mining and knowledge discovery*, vol. 28, pp. 1–30, 2014.

[91] W. Chen and G. Feng, "Spectral clustering: a semi-supervised approach," *Neurocomputing*, vol. 77, no. 1, pp. 229–242, 2012.

[92] M. Cucuringu, I. Koutis, S. Chawla, G. Miller, and R. Peng, "Simple and scalable constrained clustering: a generalized spectral method," in *AI and Statistics*. PMLR, 2016, pp. 445–454.

[93] W. Liu, M. Ye, J. Wei, and X. Hu, "Compressed constrained spectral clustering framework for large-scale data sets," *Knowledge-Based Systems*, vol. 135, pp. 77–88, 2017.

[94] P. Zhao, Y. Jiang, and Z.-H. Zhou, "Multi-view matrix completion for clustering with side information," in *Advances in Knowledge Discovery and Data Mining: 21st Pacific-Asia Conference, PAKDD 2017, Jeju, South Korea, May 23-26, 2017, Proceedings, Part II 21*. Springer, 2017, pp. 403–415.

[95] S. Ding, H. Jia, M. Du, and Y. Xue, "A semi-supervised approximate spectral clustering algorithm based on hmrf model," *Information Sciences*, vol. 429, pp. 215–228, 2018.

[96] C. Chen, H. Qian, W. Chen, Z. Zheng, and H. Zhu, "Auto-weighted multi-view constrained spectral clustering," *Neurocomputing*, vol. 366, pp. 1–11, 2019.

[97] H. Zhang, M. Gong, Y. Gu, F. Nie, and X. Li, "Side-constrained graph fusion for semi-supervised multi-view clustering," *Neurocomputing*, vol. 570, p. 127102, 2024.

[98] W. Yu, L. Xing, F. Nie, and X. Li, "An efficient semi-supervised balanced cut with hard pairwise constraints and partial labels," *Knowledge-Based Systems*, vol. 276, p. 110747, 2023.

[99] J. Zhang, R. Fan, H. Tao, J. Jiang, and C. Hou, "Constrained clustering with weak label prior," *Frontiers of Computer Science*, vol. 18, no. 3, p. 183338, 2024.

[100] T. Tong, J. Gan, G. Wen, and Y. Li, "One-step spectral clustering based on self-paced learning," *Pattern Recognition Letters*, vol. 135, pp. 8–14, 2020.

[101] X. Zhu, Y. Zhu, and W. Zheng, "Spectral rotation for deep one-step clustering," *Pattern Recognition*, vol. 105, p. 107175, 2020.

[102] G. Wen, Y. Zhu, L. Chen, and S. Zhang, "One-step spectral rotation clustering with balanced constrains," *World Wide Web*, vol. 25, no. 1, pp. 259–280, 2022.

[103] Y. Zhu, W. Zhu, and W. Wei, "Subspace guided spectral embedding learning for one-step spectral clustering," in *International Conference on Intelligent Computing*. Springer, 2024, pp. 274–285.

[104] J. Kunegis, S. Schmidt, A. Lommatzsch, J. Lerner, E. W. De Luca, and S. Albayrak, "Spectral analysis of signed graphs for clustering, prediction and visualization," in *Proceedings of the 2010 SIAM International Conference on Data Mining*. SIAM, 2010, pp. 559–570.

[105] Y. Qin, S. Ding, L. Wang, and Y. Wang, "Research progress on semi-supervised clustering," *Cognitive Computation*, vol. 11, no. 5, pp. 599–612, 2019.

[106] L. Xu, W. Li, and D. Schuurmans, "Fast normalized cut with linear constraints," in *2009 IEEE Conf. on Comput. Vision Patt. Recognit.* IEEE, 2009, pp. 2866–2873.

[107] A. Eriksson, C. Olsson, and F. Kahl, "Normalized cuts revisited: A reformulation for segmentation with linear grouping constraints," *Journal of Mathematical Imaging and Vision*, vol. 39, pp. 45–61, 2011.

[108] R. Zhang, Y. Zhang, and X. Li, "Unsupervised feature selection via adaptive graph learning and constraint," *IEEE Trans. on Neural Networks and Learning Systems*, vol. 33, no. 3, pp. 1355–1362, 2020.

[109] C. Tang, M. Wang, and K. Sun, "One-step multiview clustering via adaptive graph learning and spectral rotation," *IEEE Trans. on Neural Networks and Learning Systems*, 2024.

[110] G. Wen, Y. Zhu, and W. Zheng, "Spectral representation learning for one-step spectral rotation clustering," *Neurocomputing*, vol. 406, pp. 361–370, 2020.

[111] I. Daubechies, R. DeVore, M. Fornasier, and C. S. Güntürk, "Iteratively reweighted least squares minimization for sparse recovery," *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, vol. 63, no. 1, pp. 1–38, 2010.

[112] F. Nie, R. Zhang, and X. Li, "A generalized power iteration method for solving quadratic problem on the stiefel manifold," *Science China Information Sciences*, vol. 60, pp. 1–10, 2017.

[113] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein *et al.*, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine learning*, vol. 3, no. 1, pp. 1–122, 2011.

[114] D. P. Bertsekas, *Constrained optimization and Lagrange multiplier methods*. Academic press, 2014.

[115] J. Chen, N. Kallus, X. Mao, G. Svacha, and M. Udell, "Fairness under unawareness: Assessing disparity when protected class is unobserved," in *Proceedings of the conference on fairness, accountability, and transparency*, 2019, pp. 339–348.

Publications Related to Thesis

Journals

1. **Laxita Agrawal**, V. Vijaya Saradhi, and Teena Sharma, "**Fairness in constrained spectral clustering**", Neurocomputing, Elsevier 634 (2025): 129815. [Chapter 4]

Conferences

1. **Laxita Agrawal**, V. Vijaya Saradhi, and Teena Sharma, "**Equitable Representation of Demographic Groups in Constrained Spectral Clustering**", 2025 IEEE Conference on Artificial Intelligence **(IEEE CAI)**. IEEE, 2025. [Chapter 4]

2. **Laxita Agrawal**, V. Vijaya Saradhi, and Teena Sharma, "**Fair One-Step Spectral Rotation Clustering via Adaptive Graph Learning**", Pacific-Asia Conference on Knowledge Discovery and Data Mining **(PAKDD)**, 2025. [Chapter 6]

Under Review

Journals

1. **Laxita Agrawal**, Ahir Barman Maji, V. Vijaya Saradhi, and Teena Sharma, "**Fair and Constrained k-Nearest Neighbor Graph Construction for Spectral Clustering**", In IEEE Transaction on Artificial Intelligence **(IEEE TAI)** [Chapter 5]

Department of Computer Science and Engineering

Indian Institute of Technology Guwahati

Guwahati 781039, India