DOCTORAL THESIS

Generalized Association Rule Mining – Dimensional Unsupervised Learning

Minakshi Kaushik

TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY
TALLINN 2024

TALLINN UNIVERSITY OF TECHNOLOGY
DOCTORAL THESIS
6/2024

Generalized Association Rule Mining – Dimensional Unsupervised Learning

MINAKSHI KAUSHIK

TAL TECH PRESS

TALLINN UNIVERSITY OF TECHNOLOGY
School of Information Technologies
Department of Software Science

The dissertation was accepted for the defence of the degree of Doctor of Philosophy (Computer Science) on 5 January 2024

Supervisor:	Prof. Dr.rer.nat.habil. Dirk Draheim, Information Systems Group Department of Software Science School of Information Technologies Tallinn University of Technology Tallinn, Estonia
Opponents:	Professor Gillian Dobbie, Ph.D. University of Auckland Auckland, New Zealand
	Em.O.Univ.Prof. Dipl.-Ing. Dr.techn. A Min Tjoa, Vienna University of Technology Vienna, Austria

Defence of the thesis: 15 February 2024, Tallinn

Declaration:
Hereby, I declare that this doctoral thesis, my original investigation and achievement, submitted for the doctoral degree at Tallinn University of Technology, has not been submitted for any academic degree elsewhere.

Minakshi Kaushik ______________________________
signature

ISSN 2585-6898 (publication)
ISBN 978-9916-80-106-2 (publication)
ISSN 2585-6901 (PDF)
ISBN 978-9916-80-107-9 (PDF)
DOI https://doi.org/10.23658/taltech.6/2024
Printed by Koopia Niini & Rauam

Kaushik, M. (2024). *Generalized Association Rule Mining – Dimensional Unsupervised Learning* [TalTech Press]. https://doi.org/10.23658/taltech.6/2024

TALLINNA TEHNIKAÜLIKOOL
DOKTORITÖÖ
6/2024

Üldistatud assotsiatsioonireeglite kaevandamine – dimensiooniline juhendamata õpe

MINAKSHI KAUSHIK

TALTECH KIRJASTUS

Dedicated to My Beloved Father

Contents

List of Publications

The present Ph.D. thesis is based on the following publications that are referred to in the text by Roman numbers.

I M. Kaushik, R. Sharma, S. A. Peious, M. Shahin, S. B. Yahia, and D. Draheim. On the potential of numerical association rule mining. In *Proceedings of FDSE: 7th International Conference on Future Data and Security Engineering*, pages 3–20, Vietnam, 2020. Springer

II M. Kaushik, R. Sharma, S. A. Peious, M. Shahin, S. B. Yahia, and D. Draheim. A systematic assessment of numerical association rule mining methods. *SN Computer Science*, 2(5):1–13, 2021

III M. Kaushik, R. Sharma, S. A. Peious, and D. Draheim. Impact-driven discretization of numerical factors: Case of two- and three-partitioning. In *Proceedings of BDA: 9th International Conference on Big Data Analytics*, pages 244–260, Cham, 2021. Springer International Publishing

IV M. Kaushik, R. Sharma, M. Shahin, S. A. Peious, and D. Draheim. An analysis of human perception of partitions of numerical factor domains. In *Proceedings of iiWAS 2022 – the 24th International Conference on Information Integration and Web Intelligence*, pages 137–144, Cham, 2022. Springer Nature Switzerland

V M. Kaushik, R. Sharma, A. Vidyarthi, and D. Draheim. Discretizing numerical attributes: An analysis of human perceptions. In *Proceedings of ADBIS 2022 – the 26th International Conference on New Trends in Database and Information Systems*, pages 188–197, Cham, 2022. Springer International Publishing

VI M. Kaushik, R. Sharma, I. Fister Jr.2, and D. Draheim. Numerical association rule mining: A systematic literature review, arxiv, 2307.00662, 2023

VII M. Kaushik. Swarm-intelligence algorithms for mining numerical association rules: An exhaustive multi-aspect analysis of performance assessment data. *SSRN Electronic Journal*, 2023

VIII M. Kaushik, R. Sharma, and D. Draheim. Discretizing numerical attributes: An analysis of human perceptions (extended version). arxiv:2311.03278, 2023

Author’s Contributions to the Publications

I **First Author:** The author explored the methods and algorithms in numerical association rule mining, analyzed the results, prepared the figures and tables, and wrote the manuscript.

II **First Author:** The author conducted a literature review, analyzed the result, prepared the figures, and wrote the manuscript.

III **First Author:** The author identified the research problem, designed a mathematical formulation, proposed two novel measures, and executed the experiment. The author also prepared the figures and tables and wrote the manuscript.

IV **First Author:** The author underscored the research problem, designed and conducted the survey, experimented, and analyzed the data. The author wrote the manuscript and prepared all the figures and tables.

V **First Author:**The author conducted the experiments, analyzed the data, and compared the results of the proposed measures with human-perceived outcomes. The author reported the manuscript and prepared all the figures and tables.

VI **First Author:** The author defined research questions, conducted a systematic literature review (SLR) of 1,140 research articles, outlined the findings of the review from 68 selected articles, prepared the figures, and authored the manuscript.

VII **First Author:** The author identified the research problem, conducted a literature review, and performed a multi-aspect analysis of the swarm intelligence algorithms for numerical association rule mining. The author prepared the manuscript.

VIII **First Author:** The author designed and carried out the survey, conducted the experiment and analyzed the data, and compared the results of the proposed measures with human-perceived outcomes. The author wrote the manuscript and prepared all the figures and tables.

Abbreviations

ARM	Association Rule Mining
NARM	Numerical Association Rule Mining
QARM	Quantitative Association Rule Mining
GARM	Generalized Association Rule Mining
EA	Evolutionary Algorithm
SI	Swarm Intelligence
DE	Differential Evolution
GA	Genetic Algorithm
ML	Machine Learning
AI	Artificial Intelligence
DS	Data Science
LSQM	Least Square Ordinate-Directed Impact Measure
LADM	Least Absolute-Difference Ordinate-Directed Impact Measure
SLR	Systematic Literature Review
RQ	Research Question

Summary

This thesis presents a comprehensive compilation of eight published research articles, summarizing their key findings, methodologies, and contributions. Copies of these articles are included in the appendix for easy access and further exploration.

This research summary is organized as follows: Section 1 establishes the problem's relevance, motivation, and research objectives. Section 2 outlines the primary research questions examined and highlights the main contributions of this thesis. Section 3, outlines the research methodology employed. In Section 4, an overview of related work within the research context is presented. The findings are subsequently outlined in Section 5, showcasing their role in assessing the artifacts generated during the research. Section 6 presents the research context of this thesis and outlines the limitations and future directions for further improvements. Finally, the thesis concludes in Section 7 with a summary of the key findings and conclusions derived from this research.

1 Introduction

1.1 Problem Relevance

Data mining has many popular approaches for extracting useful information from large datasets. Out of these techniques, association rule mining (ARM) [5, 6] is one of the widely used data mining [4] techniques that aims to discover interesting associations and relationships among variables in datasets. It has many applications in various domains, such as market basket analysis, web mining, and bioinformatics. However, classical ARM has limitations when it comes to handling continuous and high-dimensional data. To overcome these limitations, researchers have generalized classical ARM in many different forms for handling more complex data types, including nominal, ordinal, and interval data, in addition to numerical data. In recent years, there has been a growing interest in developing unsupervised learning techniques for generalized association rule mining.

In general, real-world datasets contain numerical attributes that cannot be directly used in ARM. Therefore, to address this issue, the classical ARM was generalized to quantitative association rule mining (QARM) [100] or numerical association rule mining (NARM) [10]. These techniques extract meaningful rules from numerical datasets by discretizing the numerical attributes.

In this context, NARM is a subset of generalized ARM, which exclusively focuses on numerical data. In NARM, numeric attributes can be discretized by partitioning the range of numeric attributes into different intervals. Discretization is a critical step in NARM, which involves partitioning the range of numeric attributes into different intervals to transform continuous data into categorical variables. The resulting discrete values can then be used for mining association rules via traditional rule mining algorithms. Although several methods for discretizing numerical attributes have been proposed, determining the optimal method is still an open research question.

The motivation behind this thesis is to explore the current state-of-the-art methods for NARM, with a specific focus on the subjective and time-consuming nature of the discretization process. Furthermore, the existing methods for discretizing numerical attributes are not automated and require expert knowledge, which may not always result in an optimal partition. Therefore, there is a need to develop formal measures for finding the optimal partition of numerical attributes, which can lead to better performance and accuracy of NARM.

In the next three sections 1.1.1, 1.1.2 and 1.1.3, the thesis provides a detailed explanation of the problem's relevance of this research.

1.1.1 Generalized Association Rule Mining

In 1997, Srikant and Agrawal introduced the problem of mining generalized association rules [101]. Generalized ARM goes beyond the binary values utilized in classical ARM and works with multiple levels or types of attributes. NARM is a special case of generalized ARM, where only numerical data is considered. It is a variant of ARM that deals with numerical data where the attribute values are continuous or discrete numerical values.

The critical challenge in developing an efficient mining algorithm is devising a mechanism to utilize the initial frequent itemsets and association rules for the direct generation of novel generalized association rules without necessitating a reiteration of the database scanning process [108]. Partitioning is one of the popular techniques to address the issue of numerical attributes in ARM [100], and it is utilized in the Apriori algorithm [100]. Srikant and Agrawal [100] used an equi-depth partitioning approach for finding the intervals of numeric attributes. Initially, researchers and scientists employed various discretization approaches to incorporate numerical attributes into ARM. However, as time has progressed, numerous alternative methods have emerged within this field. Regrettably, this proliferation of alternative methods has led to a substantial knowledge gap in comprehending the diverse techniques utilized in NARM. To bridge this knowledge gap, a detailed systematic literature review (SLR) became imperative. Therefore, the author conducted the first SLR on NARM from a substantial collection of 1,140 scholarly articles spanning the period from 1996 to 2022. This extensive review followed the guidelines established by Kitchenham [58] and played a significant role in addressing RQ1.

The section 4.1 highlights the overview of the conducted literature review for NARM. The publications [I], [II], [VI] and [VII] presented the detailed study about NARM.

1.1.2 Human Perception of Partitions of Numerical Attributes

Measures driven by different discretization algorithms often lack the multifaceted depth of human perception, which is crucial in developing measures for discretizing numerical attributes. Moreover, the current state-of-the-art discretization techniques often fail to account for human perceptions and observations; therefore, by incorporating human perception, a more accurate representation of the underlying distributions and natural groupings within numerical attributes can be achieved. Human experts bring to the table their domain knowledge and expertise, which methods driven solely by algorithms can not replicate. Additionally, using human-perceived discretization as a benchmark for automation can result in iterative algorithm refinement, leading to continuous improvement.

The role of human perception encompasses a spectrum of contributions that collectively enhance the quality and relevance of discretization outcomes. Limited work has been conducted in the realm of aligning human perception with discretization techniques. The overview of pertinent literature provided in section 4.3 underscores existing research that delves into human perception studies, which indirectly intersect with discretization techniques. Nonetheless, these studies often lack direct alignment with discretization methods. In this context, Aupetit et al. [14] delve into the evaluation of clustering algorithms concerning their alignment with the human perception of clusters within 2D scatter plots. The main question was whether current clustering techniques align with human perceptions of clusters. The authors evaluated various algorithms, including Gaussian Mixture Models, CLIQUE, DBSCAN, Agglomerative Clustering, and over 1437 variations of k-means, on benchmark data.

To the best of our knowledge, this work is the first endeavor to incorporate human perception in identifying partitions for numerical attributes. The significance of human perception in partitioning numerical attributes was highlighted in the publication [IV]. Furthermore, publications [V] and [VIII] conducted a comparative analysis between human-perceived outcomes and the outcomes generated by the proposed measures.

1.1.3 Reflecting the Impact of Independent Numerical Attributes on Dependent Numerical Attributes

Understanding the relationships and dependencies between numerical attributes is essential for making informed decisions and extracting meaningful insights from data. Several discretizing techniques such as equi-depth, equi-width [19], ID3 [85], MDLP [31], Chi2 [69] and D2 [19] are available in the literature. However, existing methods often overlook the nuanced impact that independent numerical attributes can have on their dependent attributes.

This thesis addresses this gap by focusing on developing measures that capture and reflect this impact. By doing so, it aims to enhance the precision and applicability of data analysis, modeling, and decision-making processes across diverse fields such as finance, healthcare, engineering, and more. Ultimately, the findings of this research have the potential to contribute to the refinement of analytical techniques and tools, benefiting both the academic and practical aspects of data science and analysis. This thesis investigates unsupervised learning techniques for generalized association rule mining that extracts interesting relationships between continuous attributes. The proposed measures offer a straightforward approach to identifying suitable cut points for achieving optimal partitions. To determine the best cut points, a process of order-preserving partitioning is executed on an independent factor. The order of the independent variable is preserved using the value of data points. Therefore, the values of data points within one partition consistently remain lower than the values of data points in the subsequent partition. Initially, the outcomes are compared to cut points perceived by humans, followed by a subsequent comparison of these outcomes among themselves. At first glance, *LSQM* and k-means clustering might seem alike since both involve dividing data into clusters. However, they differ significantly in their underlying principles and applications. k-means algorithm relies on the Euclidean distance metric to calculate dissimilarity between data points, which involves measuring the geometric distance between vectors X and Y. In contrast, *LSQM* focuses on order-preserving partitioning for the independent variable, emphasizing preserving the variable's order. k-means clustering algorithm can be sensitive to the initial selection of cluster centroids, potentially leading to different clustering results depending on the initialization. However, *LSQM* is less dependent on the initial point chosen for starting the partitioning process, making it more robust in this regard. The output of k-means is cluster assignments for each data point and the coordinates of cluster centroids. Whereas the output of *LSQM* is a set of cut points that define the intervals for discretizing a numerical attribute. *LSQM* is tailored for discretizing numerical attributes by preserving their order, whereas k-means is a general-purpose clustering algorithm based on Euclidean distances.

The publication [III] presented an optimal way to find out partitions of a numerical attribute that reflect best the impact of one independent numerical attribute on a dependent numerical attribute and proposed two measures *Least Squared Ordinate-Directed Impact Measure* (LSQM) and *Least Absolute-Difference Ordinate-Directed Impact Measure* (LADM) for order-preserving partitioning of numerical factors. These impact-driven measures leverage human intuition and understanding to guide the partitioning process, resulting in data representations that are more interpretable and aligned with human cognitive abilities.

1.2 Objectives

The main objective of this thesis is to develop and investigate the best discretization tech-niques to generalize classical ARM to NARM. The identified gaps in the current state-of-the-art have clearly highlighted the need for efficient discretization techniques to handle complex and high-dimensional data and generate more interpretable and actionable results.

To address the identified gaps in the current state-of-the-art, this thesis aims to achieve the following objectives:

- Investigate the state-of-the-art NARM methods by conducting an SLR.
- Analyzing the role of human perception towards partitioning numerical attributes. The author has conducted experiments to understand how expert data scientists and statisticians partition numerical attributes under different types of data points, such as dense data points, outliers, and uneven random points.
- Developing measures for partitioning numerical attributes in a way that reflects their impact on a dependent target variable.
- Evaluating the proposed measures, conducting experiments, and comparing the results with the outcomes provided by humans on the same datasets.

2 Research Questions And Contributions

2.1 Research Questions

This thesis focuses on addressing the two significant research gaps in NARM: first, the lack of systematic and well-defined understanding of NARM, and second, the lack of optimal measures to discretize numerical attributes that can best reflect the impact on numerical target attributes. To address these research gaps, this thesis answers three main research questions presented in the sequel.

The primary RQs addressed in this thesis are:

- *RQ1* How can classical association rule mining (ARM) efficiently work with numerical-valued columns?
  - *RQ1.1* What are the state-of-the-art methods for mining association rules from numerical-valued columns?
  - *RQ1.2* How are existing discretization techniques used in the numerical ARM?
  - *RQ1.3* What are the limitations and the future potential of existing numerical ARM techniques?
  - *RQ1.4* Which Swarm Intelligence numerical ARM methods can be considered optimal and why?
- *RQ2* How do humans partition numerical attributes?
  - *RQ2.1* How to identify typical patterns of human perception (in partitioning numerical attributes)?
- *RQ3* How to find the partition of a numerical factor that reflects best the impact of this factor on a dependent numerical target variable?
  - *RQ3.1* How to develop formal measures and techniques for finding those partitions?
  - *RQ3.2* How do the proposed techniques/measures perform when compared with human perception?

Table 1 provides mapping among each RQ to the relevant publications for a better understanding of the research outcomes.

Table 1: Mapping among associated RQs and publications.

Research Questions	*Publications*
RQ1.1	[I]
RQ1.2	[II]
RQ1.3	[VI]
RQ1.4	[VII]
RQ2, RQ2.1	[IV]
RQ3, RQ3.1	[III]
RQ3.2	[V], [VIII]

To achieve the objectives of this thesis and answer the research questions, we utilized a comprehensive compilation of eight articles. These articles collectively contribute towards answering the core research questions outlined in this thesis.

The first publication [I], titled “On the Potential of Numerical Association Rule Mining," conducted an initial investigation into methods and algorithms associated with NARM. This article serves as a fundamental exploration, providing an introductory overview of the methods and algorithms used in NARM. The main purpose of this article was to establish a foundational understanding of the capabilities and potential applications of NARM techniques. Essentially, this publication answered RQ1.1 and laid the groundwork for further, more in-depth research and exploration of NARM, serving as a starting point for this research.

The second publication [II], titled “A Systematic Assessment of Numerical Association Rule Mining Methods", is an extended version of [I] and it is published in the journal Springer Nature Computer Science (SNCS). This publication expanded on the groundwork laid in the previous research, offering a more extensive and detailed exploration of NARM algorithms. The article comprehensively examines thirty NARM algorithms and delivers a detailed analysis and assessment of each algorithm. Additionally, the publication made a significant contribution by investigating the extent to which discretization techniques have been employed in NARM methods.

By examining both the algorithms themselves and the incorporation of discretization techniques, this publication answered RQ1.2 and provided valuable insights into the advancements and complexities of NARM methods, further enriching the field’s understanding of these techniques.

The third publication [III], “Impact-Driven Discretization of Numerical Factors: Case of Two- and Three-Partitioning", presented two formal measures *LSQM* (Least Squared Ordinate-Directed Impact Measure) and *LADM* (Least Absolute-Difference Ordinate-Directed Impact Measure) for order-preserving partitioning of numerical factor to optimally reflect the impact of a numerical factor onto another numerical target factor. The performance of these measures was evaluated for two-step staircase data sets (step functions), three-step staircase data sets, and arbitrary data sets (non-step functions). These measures are also validated against the human-perceived cut points and yield approximately similar results to each other. This publication answered RQ3.1.

The fourth publication [IV], “An Analysis of Human Perception of Partitions of Numerical Factor Domains", highlighted the necessity and significance of human perception in the development of formal measures for optimal partitioning of numerical factors. In this paper, an experiment was conducted with the graphs of nine synthetic datasets and three real-world datasets, wherein various data experts and non-experts were provided with graphs to identify the most optimal partitions. Following the collection of these responses, a comprehensive analysis was conducted to gain insights into how humans partition data. Additionally, the influence of data point characteristics such as density, outliers, linearity, and unevenness on human judgment towards identifying partitions was explored. By exploring these factors, the research aimed to understand the intricate relationship between human perception and specific data patterns, shedding light on the complexities involved in partitioning numerical factor domains effectively. This publication answered RQ2 and RQ2.1.

The fifth publication [V], “Discretizing Numerical Attributes: An Analysis of Human Perceptions", presented the results of the *LSQM* measure for different numbers of partitions (k = 1, 2, 3, 4, 5). The graphical representations in the paper are created for eight synthetic datasets, and their results are compared with the responses provided by data experts and non-experts. The essence of this research lies in the detailed analysis conducted on the comparison between the outcomes derived from human perception and the outcomes generated by the *LSQM* measure. To achieve this, the article employed a range of partitions (k = 1 to 5) and applied the *LSQM* measure, creating graphical representations of these partitions. These results were then rigorously compared with the responses obtained from both data experts and non-experts. This publication addressed RQ3.2.

The sixth publication [VI], “Numerical Association Rule Mining: A Systematic Literature Review", presented an extensive and meticulous SLR, adhering to the guidelines established by Kitchenham [58]. This work delved deeply into distinct methods, algorithms, metrics, and datasets related to NARM and addressed RQ1.3. To ensure the comprehensive coverage of the field, this research reviewed a vast collection of 1,140 scholarly articles published between the inception of NARM in 1996 and 2022.

The execution of this SLR involved a rigorous selection process, incorporating meticulous assessment of multiple inclusion and exclusion criteria. Additionally, the author conducted rigorous quality evaluations to ensure the reliability and validity of the selected articles. As a result of this meticulous process, only 68 articles were found to meet the stringent criteria and were deemed suitable for inclusion in the systematic review. This rigorous approach underscores the credibility and depth of the research findings presented in the publication, making it a valuable resource for anyone seeking a comprehensive understanding of the developments and trends in NARM up to the present year.

The seventh publication [VII], “Swarm-Intelligence Algorithms for Mining Numerical Association Rules: An Exhaustive Multi-Aspect Analysis of Performance Assessment Data", presented an exhaustive multi-aspect analysis of the four swarm intelligence-based algorithms for NARM with four real-world datasets and six metrics (performance time, the number of rules, support, confidence, comprehensibility, and interestingness). By systematically assessing these algorithms across the selected metrics, the research aimed to provide a holistic understanding of them and their performance and applicability in real-world scenarios. Furthermore, the research delved into the role of a multi-objective Swarm Intelligence-based optimization algorithm for NARM. This detailed assessment aids researchers and practitioners in selecting the most efficient algorithms for NARM.

The eighth publication [VIII], “Discretizing Numerical Attributes: An Analysis of Human Perceptions", is a technical report which is the full version of the publication [V].

In contrast to the fifth publication, this report encompasses a meticulous comparison of both proposed *LSQM* and *LADM* measures against human-perceived cut-points, aiming to evaluate their effectiveness in capturing human intuition regarding partitioning numerical attributes. To collect human responses, various graphs were utilized with diverse data points, including nine synthetic datasets and three real-world datasets. This report stands as a pivotal contribution to the field of data analysis by featuring the novel *LSQM* and *LADM* measures, which signify a substantial step in understanding human perceptions regarding numerical attribute partitions.

2.2 Contributions

This thesis primarily addresses the key issues of NARM, explicitly focusing on the challenge of dealing with numerical attributes and their discretization. The thesis proposed innovative approaches and techniques to effectively handle numerical attributes in the context of ARM. By addressing these primary issues, this research contributes to advancing the understanding and practice of NARM, providing valuable insights and practical solutions for working with numerical attributes. Table 2 illustrates the mapping between contribu-tions and corresponding evaluation methodologies.

The following are the main contributions of this work:

- *C1:* Contribution (C1) is the identification of the NARM problems and respective solutions. The increase in alternative methods has resulted in a significant knowledge gap in understanding diverse techniques employed in NARM. To address this knowl-

Table 2: *Mapping between thesis contributions, and corresponding evaluation methodologies.*

Contribution	Summary	Evaluation Methodology
C1	Presents an exhaustive study of NARM research articles	Informed Arguments, SLR
C2	Demonstrates the need for the human discretization of numeric attributes	Controlled Experiment
C3	Provides two formal measures to automate discretization of numeric attributes	Controlled Experiment
C4	Provides an analytical evaluation of the measures against human discretization.	Controlled Experiment

edge gap, this contribution undertakes a comprehensive SLR, thoroughly examining various methods, algorithms, metrics, and datasets extracted from a substantial collection of 1,140 scholarly articles spanning the period from NARM's inception in 1996 to 2022. This SLR is the first of its kind to provide an exhaustive study of the current state of the art on NARM, which is presented in the [VI]. The publications [I], [II] and [VII] contributed as the foundation to build this SLR and further research on the need for human discretization. This contribution effectively addressed research question RQ1.

- *C2:* Contribution (C2) is an explanation of the need and importance of human perception in partitioning numerical attributes. This contribution aims to investigate the impact of data points' features on human perception when partitioning numerical attributes. The publication [IV] corresponding to C2 highlights the importance of human perception in data partitioning and offers a nuanced understanding of how different data characteristics influence human judgment. This knowledge is fundamental for the development of algorithms that can mimic human-like partitioning processes, ensuring that the measures align closely with human intuition and decision-making capabilities. The datasets used in the experiment are available on the GitHub repository [47].

- *C3:* Contribution (C3) is the introduction of two novel measures designed for partitioning numerical attributes (see Definition 2 and Definition 3 on p. 92 of this thesis; resp. publication [III], p. 249). These measures are specifically developed to optimally reflect the impact of a numerical factor on another numerical target factor. Following evaluation, it was established that these newly proposed measures effectively identify optimal cut points aligned with human perception. These measures were detailed in the publication [III] and significantly contributed to addressing the research question RQ3.1. The pseudocode for finding the three partitions is also presented in this work.

- *C4:* Contribution (C4) is an analytical evaluation of the introduced measures in contrast to the outcomes of human perception. This research provides a thorough investigation into how humans perceive the partitioning of a numerical factor and further conducts a comprehensive comparison with one of the proposed measures as detailed in publication [V]. The research aimed to understand how human perception aligns with the results obtained through the *LSQM* measure.

  Moreover, publication [VIII] extends this analysis by presenting a comparison of human perception against both of the proposed measures. This comprehensive analysis not only contributed valuable insights into the effectiveness of the *LSQM* and

LADM measure but also provided a deeper understanding of the intricacies involved in discretizing numerical attributes. The eighth publication contributes by advancing both the theoretical understanding and the practical application of numerical attribute discretization, laying the groundwork for future research and innovation in the field. These publications collectively offer insights and solutions to address the research question RQ3.2.

3 Research Methodology

This thesis follows the design science research approach. Design science is a fundamental problem-solving paradigm that emphasizes the development and evaluation of innovative solutions to real-world problems by creating new artifacts, models, and systems [41, 72, 73, 83]. This approach is particularly relevant in disciplines that aim to design and build practical solutions, such as engineering, computer science, information systems, architecture, and product design.

According to [41, 72], design science is based on processes of building and evaluating artifacts, where the artifacts include constructs, models, methods, and implementations (called instantiations in [41]). Building involves constructing an artifact specifically designed to address a specific problem or challenge with a focus on creating a practical and innovative solution to meet identified needs. Evaluation is responsible for assessing the performance and value of the artifact, using rigorous evaluation methods to determine its effectiveness [41, 72].

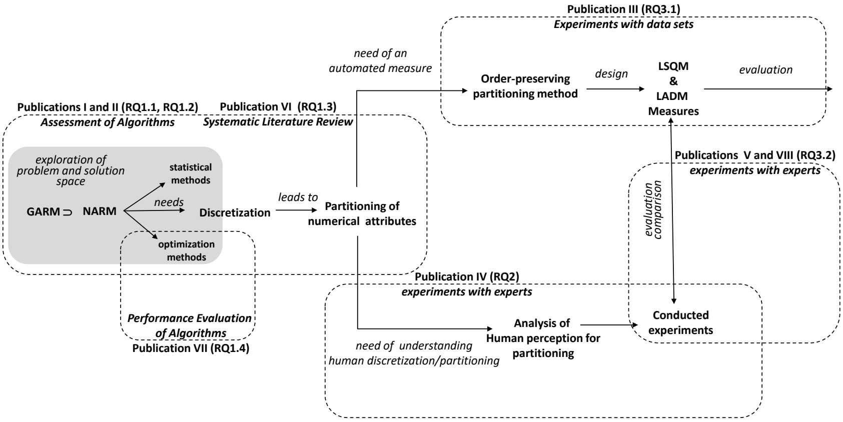

Publication III (RQ3.1)
Experiments with data sets
need of an automated measure
Order-preserving partitioning method
design
LSQM & LADM Measures
evaluation
Publications I and II (RQ1.1, RQ1.2)
Assessment of Algorithms
Publication VI (RQ1.3)
Systematic Literature Review
exploration of problem and solution space
statistical methods
GARM ⊃ NARM
needs
Discretization
leads to
Partitioning of numerical attributes
optimization methods
Publications V and VIII (RQ3.2)
experiments with experts
evaluation comparison
Publication IV (RQ2)
experiments with experts
Performance Evaluation of Algorithms
Publication VII (RQ1.4)
Analysis of Human perception for partitioning
need of understanding human discretization/partitioning
Conducted experiments

Figure 1: The design science process and corresponding publications (dashed lines).

Figure 1 illustrates the research design of the thesis, along with publications relevant to each stage and associated research questions. This thesis offers two novel artifacts in the form of two novel measures. A detailed exploration of the problem and solution space was undertaken to build the artifacts.

This thesis is based on the foundation of five peer-reviewed research articles and three technical reports. The research articles offer valuable perspectives and substantiating evidence that underpin the results and conclusions delivered in this thesis. Publication [II] is a journal article and has been published in Springer Nature Computer Science (SNCS) journal. However, four peer-reviewed articles [I], [III], [IV], [V] have been published in the proceedings of reputed conferences. An exhaustive SLR of NARM [VI] has been published in an open-access repository arXiv. Furthermore, there is an additional article [VII] available as a technical report on the renowned Social Science Research Network repository (SSRN). The publication [VIII] is an extended full version of the article [V], which is also available on arXiv.

The proposed measures were carefully developed through a rigorous examination of existing literature in the field. In order to accomplish this, a thorough examination of NARM was carried out. The publications [I], [II] and [VI] played a significant role in reaching this goal. The publication [VII] partially contributes to investigating the role of optimization algorithms for NARM.

The publication [I] delves into various methods (optimization, discretization, distribution) of NARM. The publication [II] served as an extended article of [I] and is published in the SNCS journal. Both articles [I] and [II] provide comprehensive coverage of state-of-the-art methods for NARM and emphasize the significance and necessity of the discretization method in this domain. Additionally, the publications specifically address research question RQ1.2: “In how far are existing discretization techniques used in the numerical ARM"?

The pioneering work introduced by Srikant and Agrawal [100] to discover quantitative association rules through partitioning served as a key inspiration for conducting this research on partitioning numerical attributes. The need for human perception of partitioning has simultaneously driven the research toward developing formal measures for partitioning numerical values.

The publication [IV] answered the RQ2 by conducting a study to analyze the human perception of partitioning numerical attributes. Through this research, I analyzed how humans perceive partitions in numerical data and sought to gain insights into this aspect of the partitioning process. The publication [III] presents an order-preserving method with two novel measures *LSQM* and *LADM* for partitioning numerical attributes. The evaluation of this method and measures was carried out based on extreme cases of human perception of partitions, specifically using two-step and three-step staircase scenarios. In publications [V] and [VIII], the measures were evaluated for any number of partitions with different datasets and compared the outcome of these measures with human-perceived partitions. The publication [VIII] serves as an extended version of [V], encompassing analysis of both the measures and experimentation on twelve datasets. Quantitative research aims to answer research questions and focus on measurement and testing using numerical data. This method is particularly effective for investigating relationships, patterns, and cause-and-effect relationships between variables systematically and rigorously.

The publications [I], [II], [III], [VI] and [VII] utilized the quantitative research method. On the other hand, the publications [IV], [V] and [VIII] followed quantitative research method and qualitative data collection method. The qualitative aspect of the study involves collecting and analyzing data related to human perceived responses. The study of human perception was done by conducting a survey with data science experts and non-experts. The combination of expert data scientists and non-experts as participants add depth to the study by considering diverse perspectives. The mixed-methods research approach allows the researchers to triangulate the findings from different data sources and provide a more robust analysis of the research questions.

4 Related Work

This section provides an overview of the related work that has been conducted within the scope of this thesis. Section 4.1 delves into exploring and analyzing existing methods, algorithms, metrics, datasets, and literature pertaining to NARM. Additionally, a separate section 4.2 is dedicated to the topic of discretization, which explores the different techniques used to convert numerical attributes into discrete values. It is imperative to comprehend the various discretization methods to make well-informed decisions about partitioning numerical attributes for ARM. Moreover, a comprehensive study on human perception related to the partitioning of numerical attributes is also conducted, and its overview is presented in section 4.3. It is crucial to understand how experts in the field, including data scientists and statisticians, perceive the partitioning process for developing more effective and human-intuitive approaches to NARM. By providing a thorough overview of the related work in these areas, the groundwork is established for the research and sets the stage for proposing novel solutions that address the gaps and challenges present in the current state of the field.

4.1 Numerical Association Rule Mining

This section delves into the details of various NARM techniques, examining their strengths, weaknesses, and applicability to different types of data. NARM was introduced as QARM by Srikant [100] and further evolved as NARM [10]. Srikant [100] partitioned the numeric attributes into intervals and mapped the intervals into consecutive integers and then applied the apriori [6] algorithm for finding the Boolean association rules to find quantitative association rules. Including partitioning [100, 20, 18, 34, 17, 87, 66, 30, 99], clustering [79, 65, 106, 70, 67, 38, 109, 23, 77], fuzzifying [21, 62, 42, 39, 64, 111] and hybrid [110, 80, 103, 57] approaches were also contributed to NARM. In their study, Altay et al. [10] outlined three primary approaches to addressing the NARM problem. Specifically, they discussed the use of discretization, optimization, and distribution strategies. The optimization method involves evolutionary [27], swarm intelligence based [16] and physics-based [86] algorithms. The evolution approach employs biological operators, including crossover, mutation, and selection, to mimic the evolutionary process. This ap-proach includes the genetic algorithm and differential evolution. Some prominent work in this direction to solve the NARM problem with this approach has been done by [76, 75, 74, 9, 32, 33, 11, 12]. However, the SI-based approach was first utilized for NARM by Alatas et al. in their work [7]. Further detailed work towards NARM has been conducted using different SI-based algorithms, such as [40, 63, 45, 8, 61]. Statistical concepts like mean, median, and standard deviation were utilized in some ARM studies [13, 46, 107]. Several alternative approaches have been proposed to address NARM [56, 43, 44].

During the search process, several studies [3, 2, 37] were identified that have focused on NARM approaches and their comparison. Yet, no SLR has been found and published to date. It is important to note that these existing surveys and reviews have limitations. They often lack well-defined research questions, comprehensive search strategies, and rigorous research methodologies. Indeed, the identified limitations in previous surveys and reviews illuminated a critical knowledge gap, emphasizing the need for a systematic and comprehensive examination of the existing literature. The publication [VI] addressed these limitations and fulfilled the need for a more comprehensive understanding of the field by conducting an SLR.

4.2 Discretizing Numerical Attributes

This work relies on extensively researched and highly cited literature regarding different discretization methods, including clustering and partitioning, with a particular focus on those closely aligned with the work presented in this thesis.

In state of the art, several discretization approaches such as equi-depth, equi-width [19], ID3 [85], MDLP [31], Chi2 [69], D2 [19] etc., are proposed. Surveys on discretization techniques were conducted in 2002, 2006 and 2012 by Liu [68], Kotsiantis [60] and Gracia [35]. In 1991, Catlett [19] presented the equal-width and equal-frequency discretization algorithms. The equal-width divides the range into equal-width intervals called bins based on the minimum and maximum values of the continuous attribute. However, the equal-frequency algorithm assigns an equal number of continuous values to each bin. In 1995, Dougherty et al. [24] categorized discretization methods based on global/local, supervised/unsupervised, and static/dynamic criteria. Binning, an unsupervised method, was compared to supervised methods based on entropy and purity. Global methods divide all attributes of a dataset into regions, while static methods discretize each feature separately. Dynamic methods search for inter-dependencies among features.

Most of the techniques used in this context involve dividing the continuous factor into appropriate intervals by determining the right cut points or by clustering using distance measures. However, this thesis work is related to discretization and focuses on providing partitions of one factor that best describe the impact of one factor on another. In this direction, Mehta et al. [78] proposed an unsupervised correlation-preserving discretization method based on PCA. This method effectively converts continuous attributes in multivariate data sets into discrete ones. The RUDE [71] algorithm discretizes numerical and categorical attributes with a mix of supervised and unsupervised methods. It has three steps: pre-discretizing, structure projection, and merging split points. The crucial step is structure projection, where the source attribute structure is projected onto the target attribute. Clustering is performed using the intervals, and split points are merged if the difference is within the user's specified minimum criteria.

Back in 1988, Eubank [29] and Konno et al. [59] focused on determining the most effective piecewise constant approximation of a function *f* that only has one variable. Eubank relied on the population quantile function to demonstrate how to approach the problem of finding the best piecewise constant approximation. Later on, Bergerhoff [15] suggested a method that utilized particle swarm optimization to discover the best piecewise constant approximations of one-dimensional signals. The work presented in this thesis is distinct as it does not rely on signals; instead, the primary focus is on datasets that employ multiple data points for a single value of the influencing factor. In the context of the discretization method that can focus on providing partitions of one factor that best describes the impact of one factor on another, the author of this thesis also explored other concepts related to correlation and inter-dependency among variables, discussed in statistical reasoning, e.g., Pearson correlation [82, 102], linear regression [81], ANOVA (Analysis of Variance) [36] etc. However, these tools also do not find the partition of the numerical variable that reflects best the impact on another variable.

4.3 Human Perception for Partitioning

This section provides an overview of studies related to human perception that are connected to discretization techniques.

In the current state of the art, several studies have leveraged human perception to evaluate diverse techniques. However, it is essential to note that these studies are not directly aligned with the specific context of discretization. Tatu et al. [104] explored how people perceive multidimensional data using visual quality criteria. They conducted a user study to investigate the connection between the interpretation of clusters by humans and the measurements that are automatically obtained from 2D scatter plots. This article found that contrary to intuition, separation is more crucial than density or overlap in cluster perception. While the importance of different cluster shapes remained unexplored, the results of this article indicated a need for further research to understand the impact of shapes on user perception. Additionally, this research suggested investigating a combination of measures based on both density and separation for a more comprehensive evaluation of clusters in data visualization.

The article presented by Etemadpour et al. [28], aimed to explore the significance of human perception in the analysis of projected views. The evaluation was based on human perception. Participants were asked to evaluate high-dimensional data by identifying clusters and analyzing distances both within and between clusters. The objective was to determine whether distances solely influence subjects' decisions in typical visual analysis tasks in the projected space or if other cluster properties such as cluster density, shape, size, and orientation also play a role. The article found that cluster density and shape significantly affect perception during visual inspection, leading to biased results in experiments. Cluster size and scatter plot orientation, however, did not have a significant impact. Overall, the research underscores the influence of cluster properties on the outcomes of visual analysis tasks.

In this context, another research article conducted by Demiralp et al. [22] utilized human judgments to estimate perceptual kernels for visual encoding variables, including shape, size, color, and various combinations. To facilitate the experiment, Amazon's Mechanical Turk platform was utilized, and the experiment involved twenty Turkers who collectively completed thirty MTurk jobs. The use of human perceptions in this article contributes to a deeper understanding of how these variables are perceived in visual encoding tasks.

A recent research [1] made a significant contribution by introducing a novel method for evaluating the visual quality of monochrome scatter plots. Their approach, ClustMe, utilized a data-driven visual quality measure (VQM) derived from human perceptual data. By ranking monochrome scatterplots based on cluster patterns, this method provided a valuable tool for assessing the effectiveness of visualization techniques. Despite these advancements, the question of which specific set of VQM is optimal for exploring multidimensional data remains unanswered. The article identified the need for further research in determining the most suitable VQM, indicating an open avenue for future investigations in the field of data visualization and analysis. Similarly, an article by Aupetit et al. [14] aimed to assess the effectiveness of clustering algorithms in aligning with human perception of clusters in 2D scatter plots. The investigation evaluated various algorithms, including Gaussian Mixture Models, CLIQUE, DBSCAN, Agglomerative Clustering, and over 1437 variations of k-means, on benchmark data. The primary objective was to determine how well these algorithms corresponded with human perceptions of clusters. This research also addressed the difficulties associated with expanding perception-based approaches to higher-dimensional spaces.

This work is inspired by research that relies on human judgments to evaluate different techniques. It aims to test the effectiveness of the proposed measure called *LSQM* and *LADM* in terms of how humans perceive it.

5 Evaluation

Evaluation entails the process of observing and quantifying how effectively the artifact contributes to solving a problem. This involves comparing the intended objectives of a solution with the actual observed outcomes resulting from the artifact's utilization during the demonstration [83]. Evaluating a designed IT artifact necessitates establishing relevant metrics and, potentially, collecting and analyzing pertinent data [41].

This section aims to illustrate the research's evaluation through the presentation of results that pertain to the research questions. Here, the focus lies on elucidating how these results effectively address gaps within the existing body of knowledge, advancing our understanding of the subject. Additionally, this section highlights the alignment between the outcomes of the research questions and the study's objectives, validating the research's relevance and significance.

Table 3 provides an all-encompassing overview of the outcomes derived from addressing RQ1, including the publications that address these questions. Four research articles are relevant to RQ1 and its sub-research questions. Table 3 offers a precise summary of the outcomes achieved in regard to the specific research questions, effectively highlighting the contributions put forth in each respective publication. All four publications [I], [II], [VI] and [VII] are relevant to RQ1.

- *RQ1* How can classical association rule mining (ARM) efficiently work with numerical-valued columns?
  - *RQ1.1* What are the state-of-the-art methods for mining association rules from numerical-valued columns?
  - *RQ1.2* In how far are existing discretization techniques used in the numerical ARM?
  - *RQ1.3* What are the limitations and the future potential of existing numerical ARM techniques?
  - *RQ1.4* Which Swarm Intelligence numerical ARM methods can be considered optimal and why?

Table 4, shows the results and associated publications for the research question RQ2.

- *RQ2* How do humans partition numerical attributes?
  - *RQ2.1* How to identify typical patterns of human perception (in partitioning numerical attributes)?

The results and publications related to research question RQ3 are presented in Table 5. The three publications [III], [V] and [VIII] are relevant to RQ3.

- *RQ3* How to find the partition of a numerical factor that reflects best the impact of this factor on a dependent numerical target variable?
  - *RQ3.1* How to develop formal measures and techniques for finding those partitions?
  - *RQ3.2* How do the proposed techniques/measures perform when compared with human perception?

Table 3: Summary of the RQ1 results with publication and current knowledge gaps.

Results and Associated Publications	Current Knowledge Gaps
In the publication [I], three NARM methods were presented, and 24 algorithms were reviewed in detail. This work particularly addressed the research question RQ1.1.	Section 4.1 outlined the limitations inherent in previous surveys and reviews, which have been successfully addressed and surpassed in the context of this study.
The publication [II] is an extended version of [I]. This work provides an in-depth analysis of 30 NARM algorithms and explores the extent to which discretization techniques are utilized in these methods. Importantly, this publication significantly contributed to the exploration of research question RQ1.2.	Section 4.1 also outlined the lack of studies that scrutinize the extent to which discretization techniques have been integrated into NARM methods.
The publication [VI] presents a detailed and exhaustive SLR that followed guidelines provided by Kitchenham [58]. This publication addressed RQ1.3.	The section 4.1 highlights that there is no SLR available on NARM in the existing reviews which followed any systematic research methodology.
The publication [VI] provides a thorough investigation of a wide range of methods, algorithms, metrics, and datasets sourced from 1,140 scholarly articles spanning the period from the introduction of NARM in 1996 to 2022. This SLR involved a rigorous selection process that assessed multiple inclusion and exclusion criteria, as well as quality assessment. As a result, 68 articles were chosen.	According to the investigation conducted in section 4.1, it is evident that no existing study has presented a detailed investigation about NARM and used such a rigorous selection process.
The publication [VII] conducted a comprehensive multi-aspect analysis of the Swarm Intelligence (SI)-based algorithms (along with their performance) applied in NARM. This publication played a partial role in addressing the research question RQ1.4. This work contributes by understanding the performance of the subset (SI) of the optimization method.	The overview of the studies presented in the section 4.1 also highlights that numerous research works focus on the performance analysis of NARM algorithms utilizing evolutionary algorithms. Interestingly, there seems to be a lack of performance assessment for SI-based NARM algorithms.

Table 4: Summary of the RQ2 results with publication and current knowledge gaps.

Results and Associated Publications	Current Knowledge Gaps
The publication [IV] contributed by identifying the need for human perception towards partitioning the numerical attribute. This work raised the importance of perceptual conception in developing the measure for partitioning numerical factors. This work addresses the research question RQ2. After conducting extensive experimentation with different features data points, the responses were collected and analyzed human perception for partitioning.	In the provided section 4.2, 4.3, the review of relevant literature highlights instances where human perception has been integrated into certain discretization techniques. However, it is noteworthy that there appears to be a gap in the existing body of work concerning research specifically dedicated to the utilization of human perception for partitioning numeric attributes.
This research article formulated and tested four hypotheses to investigate the impact of data point characteristics on human perception when determining optimal cut points for partitioning numerical attributes.	The studies discussed in section 4.3 provide insight into aligning human perception with clustering. However, this work only aimed to determine the cut points identified by humans.

Table 5: Summary of the RQ3 results with publication and current knowledge gaps.

Results and Associated Publications	Current Knowledge Gaps
Three publications addressed the RQ3. The publication [III] contributed to addressing the main research question RQ3 and sub-research question RQ3.1 by developing two formal measures for partitioning numerical attributes. This work introduced the order-preserving partitioning of a numerical factor that reflects best the impact of this factor on a dependent numerical target factor.	Section 4.2 specifically shows the lack of discretization technique in literature, which finds the partition of the numerical variable that reflects best the impact on another variable.
The publication [V] presented an in-depth analysis of human perceptions of partitioning a numerical factor and compared it with one of the proposed measures with eight datasets. However, publication [VIII] evaluated and analyzed both the proposed measures against human perceived responses with twelve datasets. The outcomes of both measures closely align with each other, yielding results that closely approximate human perception. Both of these publications address the research question RQ3.2.	As presented in the overview in section 4.2 and 4.3, it becomes evident that the domain lacks a formalized measure for the process of partitioning. Additionally, comparing the results of state-of-the-art measures against human responses is not a widely practiced technique in the field.

In summation, the responses to the primary research question and its accompanying sub-research questions, as furnished by this study, have significantly contributed to the fulfillment of all predefined objectives. The first objective was accomplished by conducting an SLR following the guidelines of Kitchenham [58]. Second, two IT artifacts have been developed as mathematical measures to partition numerical attributes that reflect their impact on a dependent target variable. Third, an experiment was conducted to collect human responses for partitioning numeric attributes and analyze human perception using different types of data points. Fourth, to evaluate these measures, a comparative analysis has been conducted.

6 Discussion

The thesis presents novel impact-driven measures that capitalize on the concept of human perception for partitioning numerical attributes and identifying optimal cut points. Traditional numerical discretization methods often overlook the subtleties of numerical relationships, leading to data representations that may not fully capture the underlying patterns and insights. In contrast, the impact-driven measures leverage human intuition and understanding to guide the partitioning process, resulting in data representations that are more interpretable and aligned with human cognitive abilities.

Human perception brings a qualitative dimension to the process, allowing for the incorporation of domain knowledge, contextual understanding, and cognitive reasoning. By involving human experts in the discretization process, the resulting partitions are more likely to align with how humans perceive and interpret data. This human touch enables the identification of meaningful cut points that align with how humans naturally perceive data. Moreover, the examination of existing literature through a systematic review provided a solid foundation for identifying gaps and opportunities for further exploration, leading to a deeper understanding of the landscape and methodologies of NARM.

6.1 Research Context

In the classical data analytics landscape, many data analysis practices are rooted around three popular decision support techniques (DSTs), i.e., statistical reasoning, online analytical processing (OLAP), and association rule mining (ARM). The context of this research is also rooted around the intersection of these three decision-support techniques, contributing to the evolution of the research's core idea.

The author of this thesis has particularly worked on generalizing the classical ARM techniques from binary values to numerical values. As in its original form, classical ARM faces substantial limitations, e.g., the existing ARM techniques are confined to discrete-valued columns, leaving numerical-valued columns unaddressed. This limitation becomes particularly problematic in practical situations where numerical data is predominant. Therefore, an integral facet of this research lies in exploring discretizing numerical attributes, a critical undertaking within the realm of frequent itemset mining. This process holds particular significance, especially concerning QARM [100].

Further, the inspiration for this research stems from an exploration of several research domains, including partial conditionalization [25, 26], ARM [5, 6, 98, 88], and NARM [100]. These research areas collectively provide the foundation upon which this thesis is built. In parallel, this research also draws inspiration from the development of complementary tools and frameworks. A notable mention is the introduction of the Grand report [84], and a framework [93] for unifying ARM, statistical reasoning (SR), and OLAP. The Grand report [84] reports the mean value of a chosen numeric target column concerning all possible combinations of influencing factors. Figure 2 illustrates the research context of this thesis.

The work by [93] analyzes the inconsistencies and gaps between DSTs and proposes strategies to bridge the gap among the three popular DSTs: SR, OLAP, and ARM. The research on unifying DSTs elaborated the semantic correspondences between the foundations of SR, OLAP and ARM, i.e., probability theory, relational algebra and the itemset apparatus, respectively. Rahul et al. have developed a novel framework that unifies DSTs and created a tool to validate it. This tool simplifies the unified utilization of DSTs in decision support, showing how SR, ARM, and OLAP can complement each other in improving data comprehension, visualization and decision-making processes.

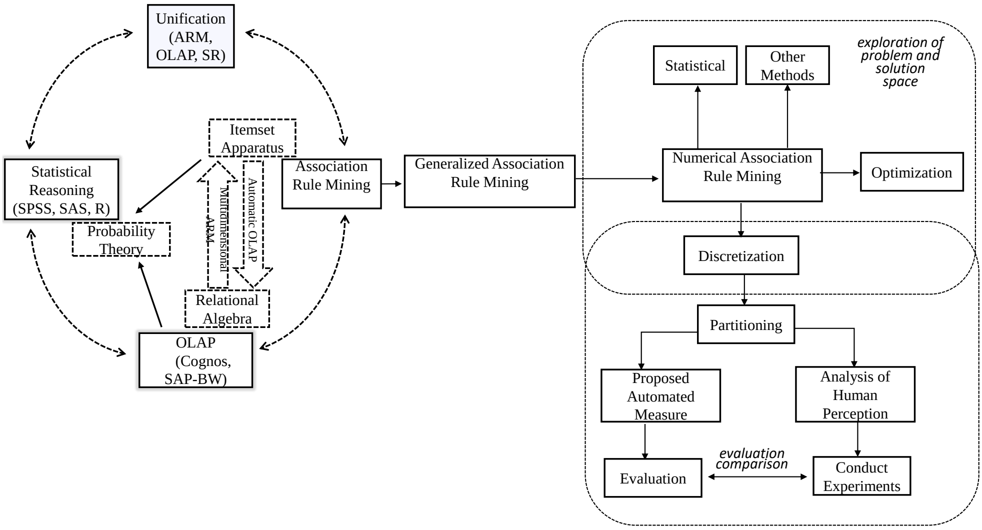


Figure 2: Research Context

This research strengthens the generalization of ARM by finding the partitions of numerical attributes that reflect best the impact of one independent numerical attribute on a dependent numerical attribute. The overarching vision of this research encompasses the development of an ecosystem that refines machine learning (ML) approaches, specifically ARM, to a new level of effectiveness, adaptability, and applicability.

6.2 Summary of Contributions

This research offers significant contributions, summarized as follows:

- This research provides an in-depth analysis of existing research articles in the field of NARM, offering a comprehensive overview of the existing literature.
- The thesis underscores the necessity of human involvement in the discretization process of numeric attributes, recognizing the significance of human perception in this domain.
- The research in this thesis introduces two formal measures, aiming to automate the discretization of numeric attributes. These measures represent innovative approaches in the field, designed to enhance the efficiency and accuracy of the discretization process.
- The thesis conducts a rigorous analytical evaluation of the proposed measures by comparing them to human discretization. This comparative analysis serves as a critical assessment, validating the effectiveness and reliability of the proposed formal measures against human perception.

6.3 Limitations

This thesis contributes valuable insights and advancements to the field of NARM; however, several limitations need to be acknowledged. The effectiveness and generalizability of the proposed measures and methodologies depend on the characteristics of the datasetsused for experimentation. Therefore, the results might not fully extend to datasets from different domains or with significantly different attributes. Further, the two proposed measures are theoretically grounded and have shown promising results within controlled experimental settings; however, their performance and robustness in complex real-world scenarios remain to be thoroughly validated. Moreover, their comparison and benchmarking against existing measures are limited. Currently, the suggested measures necessitate user input for the number of partitions (k). However, the overarching objective is to advance towards automating this process by leveraging various ML algorithms, such as the Elbow method [105], to identify the optimal value of k.

These identified limitations underscore crucial areas for improvement, offering clear guidance for future research endeavors. The primary focus lies in using ML techniques to elevate automation levels and reduce human intervention.

6.4 Future Work

This research has provided valuable insights and contributions to the fields of NARM and developed two measures for effective partitioning of numerical attributes, but still, further exploration and expansion are needed to improve the effectiveness and applicability of the proposed measures.

In the current settings in the measures, human intervention is required to set the number of partitions (k). This somehow restricts the measures to provide a fully automatic outcome. Therefore, to overcome the ratio of human involvement, future research can be focused on utilizing and developing advanced ML algorithms, e.g., Elbow method [105]. These algorithms can help to identify the number of cut points (value of k) and utilize the proposed measures to optimally decide the value of cut points. Exploring ML algorithms, heuristic methods, or evolutionary techniques tailored for this specific task can lead to a more automated and objective approach to selecting the number of cut points, aligning with the goal of complete automation. Moreover, integrating ML techniques into NARM can potentially revolutionize the NARM field to automatically detect intricate patterns and relationships in data, enhancing accuracy and reducing the effort required for pattern identification.

7 Conclusion

The core objective of this Ph.D. thesis is to address three primary and seven supplementary research questions, with the overarching goal of delivering invaluable insights and pragmatic solutions for addressing the numerical attributes in ARM. The thesis draws upon the research findings and contributions from a collection of five peer-reviewed articles and three technical reports published between 2020 and 2023.

To address the RQ1 and its associated four sub-research questions, publications [I], [II] and [VI] delve into the identification of the NARM problems and respective solutions.

The publication [VI] presents an extensive SLR that is the first of its kind to provide an exhaustive analysis of the current literature and previous surveys on NARM, comprehensively explores diverse methods, metrics, and various facets of NARM. This review draws insights from a substantial collection of 1,140 scholarly articles from 1996 to 2022.

The publication [IV] addresses RQ2 and emphasizes the importance of human perception for developing useful methods for discretizing numerical attributes. The significance of RQ3 lies in its role of pioneering the development of innovative techniques aimed at discretizing numerical attributes in a manner that closely aligns with human perception. The publications [III], [V] and [VIII] have made significant contributions in this aspect by introducing and validating two novel measures. The evaluation of these measures was conducted by gathering human perception responses, thereby enhancing the understanding of their effectiveness and practical implications. Moreover, the publication [VII] presented an exhaustive multi-aspect analysis of the four swarm intelligence-based algorithms for NARM and investigated the role of the multi-objective SI-based optimization algorithm for NARM.

In total, this research presents four substantial contributions to the domains of ARM, QARM, or NARM.

- Offering a thorough investigation of research articles focused on NARM.
- Presenting the significance and importance of human perception for discretization of numeric attributes.
- Developing two novel formal measures to automate the discretization of numeric attributes.
- Providing analytical evaluation of the measures against collected responses of human perception.

The research successfully answered the three formulated research questions.

Examining existing literature through a systematic review provided a solid foundation for identifying gaps and opportunities for further exploration, leading to a deeper understanding of the landscape and methodologies of NARM. Furthermore, analyzing human perception in data analysis underscores the significance of human perception in developing effective measures for partitioning numerical attributes.

The impact of this research resonates across decision support systems, data analytics, and the broader landscape of ML. As we move forward, the insights and solutions presented here undoubtedly shape the future of these fields, contributing to enhanced data comprehension, more accurate analyses, and informed decision-making.

List of Figures

List of Tables

References

[1] M. M. Abbas, M. Aupetit, M. Sedlmair, and H. Bensmail. Clustme: A visual quality measure for ranking monochrome scatterplots based on cluster patterns. *Computer Graphics Forum*, 38(3):225–236, 2019.

[2] D. Adhikary and S. Roy. Mining quantitative association rules in real-world databases: A review. In *2015 1st International Conference on Computing and Communication Systems (I3CS)*, volume 1, pages 87–92, India, 2015. IGI Global.

[3] D. Adhikary and S. Roy. Trends in quantitative association rule mining techniques. In *2015 IEEE 2nd International Conference on Recent Trends in Information Systems (ReTIS)*, pages 126–131, India, 2015. IEEE.

[4] R. Agrawal, T. Imielinski, and A. Swami. Database mining: a performance perspective. *IEEE Transactions on Knowledge and Data Engineering*, 5(6):914–925, 1993.

[5] R. Agrawal, T. Imieliński, and A. Swami. Mining Association Rules Between Sets of Items in Large Databases. *ACM SIGMOD Record*, 22(2):207–216, 1993.

[6] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *Proceedings of VLDB'1994 – the 20th International Conference on Very Large Data Bases*, page 487–499, Chile, 1994. Morgan Kaufmann.

[7] B. Alatas and E. Akin. Rough particle swarm optimization and its applications in data mining. *Soft Computing*, 12(12):1205–1218, 2008.

[8] B. Alatas and E. Akin. Chaotically encoded particle swarm optimization algorithm and its applications. *Chaos, Solitons & Fractals*, 41(2):939–950, 2009.

[9] B. Alatas, E. Akin, and A. Karci. Modenar: Multi-objective differential evolution algorithm for mining numeric association rules. *Applied Soft Computing*, 8(1):646–656, 2008.

[10] E. V. Altay and B. Alatas. Intelligent optimization algorithms for the problem of mining numerical association rules. *Physica A: Statistical Mechanics and its Applications*, 540:123142, 2020.

[11] E. V. Altay and B. Alatas. Differential evolution and sine cosine algorithm based novel hybrid multi-objective approaches for numerical association rule mining. *Information Sciences*, 554:198–221, 2021.

[12] E. V. Altay and B. Alatas. Chaos numbers based a new representation scheme for evolutionary computation: Applications in evolutionary association rule mining. *Concurrency and Computation: Practice and Experience*, 34(5):e6744, 2022.

[13] Y. Aumann and Y. Lindell. A statistical theory for quantitative association rules. *Journal of Intelligent Information Systems*, 20(3):255–283, 2003.

[14] M. Aupetit, M. Sedlmair, M. M. Abbas, A. Baggag, and H. Bensmail. Toward perception-based evaluation of clustering techniques for visual analytics. In *2019 IEEE Visualization Conference (VIS)*, pages 141–145, 2019.

[15] L. Bergerhoff, J. Weickert, and Y. Dar. Algorithms for piecewise constant signal approximations. In *27th European Signal Processing Conference (EUSIPCO)*, pages 1–5. IEEE, 2019.

[16] E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, 10 1999.

[17] S. Brin, R. Rastogi, and K. Shim. Mining optimized gain rules for numeric attributes. In *Proceedings of KDD'99 – the 5th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 135–144, New York, 1999. ACM.

[18] O. Büchter and R. Wirth. Discovery of association rules over ordinal data: A new and faster algorithm and its application to basket analysis. In X. Wu, R. Kotagiri, and K. B. Korb, editors, *Research and Development in Knowledge Discovery and Data Mining*, pages 36–47, Berlin, Heidelberg, 1998. Springer Berlin Heidelberg.

[19] J. Catlett. On changing continuous attributes into ordered discrete attributes. In *European working session on learning*, pages 164–178. Springer, 1991.

[20] K. C. Chan and W.-H. Au. An effective algorithm for mining interesting quantitative association rules. In *Proceedings of the 1997 ACM symposium on Applied computing*, pages 88–90, San Jose, CA, United States, 1997. ACM.

[21] K. C. Chan and W.-H. Au. Mining fuzzy association rules. In *Proceedings of the sixth international conference on information and knowledge management*, pages 209–215, Las Vegas Nevada USA, 1997. ACM.

[22] Ç. Demiralp, M. S. Bernstein, and J. Heer. Learning perceptual kernels for visualization design. *IEEE transactions on visualization and computer graphics*, 20(12):1933–1942, 2014.

[23] X. Dong and D. Pi. An effective method for mining quantitative association rules with clustering partition in satellite telemetry data. In *2014 Second International Conference on Advanced Cloud and Big Data*, pages 26–33, Huangshan, China, 2014. IEEE.

[24] J. Dougherty, R. Kohavi, and M. Sahami. Supervised and unsupervised discretization of continuous features. In *Machine learning proceedings 1995*, pages 194–202. Elsevier, 1995.

[25] D. Draheim. *Generalized Jeffrey Conditionalization: A Frequentist Semantics of Partial Conditionalization*. Springer, 2017.

[26] D. Draheim. Future perspectives of association rule mining based on partial conditionalization. In S. Hartmann, J. Küng, S. Chakravarthy, G. Anderst-Kotsis, A Min Tjoa, and I. Khalil, editors, *Proceedings of DEXA'2019 - the 30th International Conference on Database and Expert Systems Applications*, volume 11706 of *LNCS*, page xvi, Heidelberg New York Berlin, 2019. Springer.

[27] A. E. Eiben and J. E. Smith. *Introduction to evolutionary computing*. Springer, Berlin Heidelberg New York, 2015.

[28] R. Etemadpour, R. C. da Motta, J. G. de Souza Paiva, R. Minghim, M. C. F. de Oliveira, and L. Linsen. Role of human perception in cluster-based visual analysis of multidimensional data projections. In *2014 International Conference on Information Visualization Theory and Applications (IVAPP)*, pages 276–283, 2014.

[29] R. Eubank. Optimal grouping, spacing, stratification, and piecewise constant approximation. *Siam Review*, 30(3):404–420, 1988.

[30] G. Fan, W. Shi, L. Guo, J. Zeng, K. Zhang, and G. Gui. Machine learning based quantitative association rule mining method for evaluating cellular network performance. *IEEE Access*, 7:166815–166822, 2019.

[31] U. Fayyad and K. B. Irani. Multi-interval discretization of continuousvalued attributes for classification learning, 1993. In *13th Int'l Joint Conf. Artificial Intelligence (IJCAI)*, 1993.

[32] I. Fister, A. Iglesias, A. Galvez, J. Del Ser, E. Osaba, and I. Fister. Differential evolution for association rule mining using categorical and numerical attributes. In H. Yin, D. Camacho, P. Novais, and A. J. Tallón-Ballesteros, editors, *Intelligent Data Engineering and Automated Learning – IDEAL 2018*, pages 79–88, Cham, 2018. Springer International Publishing.

[33] I. Fister Jr., V. Podgorelec, and I. Fister. Improved nature-inspired algorithms for numeric association rule mining. In P. Vasant, I. Zelinka, and G.-W. Weber, editors, *Intelligent Computing and Optimization*, pages 187–195, Cham, 2021. Springer International Publishing.

[34] T. Fukuda, Y. Morimoto, S. Morishita, and T. Tokuyama. Mining optimized association rules for numeric attributes. *Journal of Computer and System Sciences*, 58(1):1–12, 1999.

[35] S. Garcia, J. Luengo, J. A. Sáez, V. Lopez, and F. Herrera. A survey of discretization techniques: Taxonomy and empirical analysis in supervised learning. *IEEE Transactions on Knowledge and Data Engineering*, 25(4):734–750, 2012.

[36] A. Gelman et al. Analysis of variance – why it is more important than ever. *The Annals of statistics*, 33(1):1–53, 2005.

[37] A. Gosain and M. Bhugra. A comprehensive survey of association rules on quantitative data in data mining. In *2013 IEEE Conference on Information & Communication Technologies*, pages 1003–1008, India, 2013. IEEE.

[38] Y. Guo, J. Yang, and Y. Huang. An effective algorithm for mining quantitative association rules based on high dimension cluster. In *2008 4th International Conference on Wireless Communications, Networking and Mobile Computing*, pages 1–4, Dalian, China, 2008. IEEE.

[39] A. Gyenesei. A fuzzy approach for mining quantitative association rules. *Acta Cybern.*, 15:305–320, 2001.

[40] K. E. Heraguemi, N. Kamel, and H. Drias. Multi-objective bat algorithm for mining numerical association rules. *International Journal of Bio-Inspired Computation*, 11(4):239–248, 2018.

[41] A. R. Hevner, S. T. March, J. Park, and S. Ram. Design science in information systems research. *MIS quarterly*, 28(1):75–105, 2004.

[42] T.-P. Hong, C.-S. Kuo, and S.-C. Chi. Mining association rules from quantitative data. *Intelligent data analysis*, 3(5):363–376, 1999.

[43] Z. Hu, M. Shao, H. Liu, and J. Mi. Cognitive computing and rule extraction in generalized one-sided formal contexts. *Cognitive Computation*, 14(6):2087–2107, 2022.

[44] I. F. Jaramillo, J. Garzás, and A. Redchuk. Numerical association rule mining from a defined schema using the vmo algorithm. *Applied Sciences*, 11(13):21, 2021.

[45] I. Kahvazadeh and M. S. Abadeh. Mocanar: a multi-objective cuckoo search algorithm for numeric association rule discovery. *Computer Science & Information Technology*, 99:113, 2015.

[46] G.-M. Kang, Y.-S. Moon, H.-Y. Choi, and J. Kim. Bipartition techniques for quantitative attributes in association rule mining. In *TENCON 2009-2009 IEEE Region 10 Conference*, pages 1–6, Singapore, 2009. IEEE.

[47] M. Kaushik. Datasets. `https://github.com/minakshikaushik/LSQM-measure.git`, 2022.

[48] M. Kaushik. Swarm-intelligence algorithms for mining numerical association rules: An exhaustive multi-aspect analysis of performance assessment data. *SSRN Electronic Journal*, 2023.

[49] M. Kaushik, R. Sharma, and D. Draheim. Discretizing numerical attributes: An analysis of human perceptions (extended version). arxiv:2311.03278, 2023.

[50] M. Kaushik, R. Sharma, I. Fister Jr.2, and D. Draheim. Numerical association rule mining: A systematic literature review, arxiv, 2307.00662, 2023.

[51] M. Kaushik, R. Sharma, S. A. Peious, and D. Draheim. Impact-driven discretization of numerical factors: Case of two- and three-partitioning. In *Proceedings of BDA: 9th International Conference on Big Data Analytics*, pages 244–260, Cham, 2021. Springer International Publishing.

[52] M. Kaushik, R. Sharma, S. A. Peious, M. Shahin, S. B. Yahia, and D. Draheim. On the potential of numerical association rule mining. In *Proceedings of FDSE: 7th International Conference on Future Data and Security Engineering*, pages 3–20, Vietnam, 2020. Springer.

[53] M. Kaushik, R. Sharma, S. A. Peious, M. Shahin, S. B. Yahia, and D. Draheim. A systematic assessment of numerical association rule mining methods. *SN Computer Science*, 2(5):1–13, 2021.

[54] M. Kaushik, R. Sharma, M. Shahin, S. A. Peious, and D. Draheim. An analysis of human perception of partitions of numerical factor domains. In *Proceedings of iiWAS 2022 – the 24th International Conference on Information Integration and Web Intelligence*, pages 137–144, Cham, 2022. Springer Nature Switzerland.

[55] M. Kaushik, R. Sharma, A. Vidyarthi, and D. Draheim. Discretizing numerical attributes: An analysis of human perceptions. In *Proceedings of ADBIS 2022 – the 26th International Conference on New Trends in Database and Information Systems*, pages 188–197, Cham, 2022. Springer International Publishing.

[56] Y. Ke, J. Cheng, and W. Ng. An information-theoretic approach to quantitative association rule mining. *Knowledge and Information Systems*, 16(2):213–244, 2008.

[57] K. Kianmehr, M. Alshalalfa, and R. Alhajj. Fuzzy clustering-based discretization for gene expression classification. *Knowledge and Information Systems*, 24(3):441–465, 2010.

[58] B. Kitchenham, O. Pearl Brereton, D. Budgen, M. Turner, J. Bailey, and S. Linkman. Systematic literature reviews in software engineering – a systematic literature review. *Information and Software Technology*, 51(1):7–15, 2009.

[59] H. Konno and T. Kuno. Best piecewise constant approximation of a function of single variable. *Operations research letters*, 7(4):205–210, 1988.

[60] S. Kotsiantis and D. Kanellopoulos. Discretization techniques: A recent survey. *GESTS International Transactions on Computer Science and Engineering*, 32(1):47–58, 2006.

[61] R.-J. Kuo, M. Gosumolo, and F. E. Zulvia. Multi-objective particle swarm optimization algorithm using adaptive archive grid for numerical association rule mining. *Neural Computing and Applications*, 31(8):3559–3572, 2019.

[62] C. M. Kuok, A. Fu, and M. H. Wong. Mining fuzzy association rules in databases. *SIGMOD Rec.*, 27(1):41–46, mar 1998.

[63] M. Ledmi, H. Moumen, A. Siam, H. Haouassi, and N. Azizi. A discrete crow search algorithm for mining quantitative association rules. *International Journal of Swarm Intelligence Research (IJSIR)*, 12(4):101–124, 2021.

[64] K.-M. Lee. Mining generalized fuzzy quantitative association rules with fuzzy generalization hierarchies. In *Proceedings Joint 9th IFSA World Congress and 20th NAFIPS International Conference (Cat. No. 01TH8569)*, volume 5, pages 2977–2982, Canada, 2001. IEEE.

[65] B. Lent, A. Swami, and J. Widom. Clustering association rules. In *Proceedings 13th International Conference on Data Engineering*, pages 220–231, Birmingham, UK, 1997. IEEE.

[66] J. Li, H. Shen, and R. Topor. An adaptive method of numerical attribute merging for quantitative association rule mining. In L. C. K. Hui and D.-L. Lee, editors, *Internet Applications*, pages 41–50, Berlin, Heidelberg, 1999. Springer Berlin Heidelberg.

[67] W. Lian, D. W. Cheung, and S. Yiu. An efficient algorithm for finding dense regions for mining quantitative association rules. *Computers & Mathematics with Applications*, 50(3-4):471–490, 2005.

[68] H. Liu, F. Hussain, C. L. Tan, and M. Dash. Discretization: An enabling technique. *Data mining and knowledge discovery*, 6(4):393–423, 2002.

[69] H. Liu and R. Setiono. Feature selection via discretization. *IEEE Transactions on Knowledge and Data Engineering*, 9(4):642–645, 1997.

[70] M.-C. Lud and G. Widmer. Relative unsupervised discretization for association rule mining. In *Principles of Data Mining and Knowledge Discovery*, pages 148–158, Berlin, Heidelberg, 2000. Springer Berlin Heidelberg.

[71] M.-C. Lud and G. Widmer. Relative unsupervised discretization for association rule mining. In *European conference on principles of data mining and knowledge discovery*, pages 148–158. Springer, 2000.

[72] S. T. March and G. F. Smith. Design and natural science research on information technology. *Decision Support Systems*, 15(4):251–266, 1995.

[73] M. L. Markus, A. Majchrzak, and L. Gasser. A design theory for systems that support emergent knowledge processes. *MIS Quarterly*, 26(3):179–212, 2002.

[74] D. Martín, J. Alcalá-Fdez, A. Rosete, and F. Herrera. Nicgar: A niching genetic algorithm to mine a diverse set of interesting quantitative association rules. *Information Sciences*, 355-356:208–228, 2016.

[75] D. Martín, A. Rosete, J. Alcalá-Fdez, and F. Herrera. Qar-cip-nsga-ii: A new multi-objective evolutionary algorithm to mine quantitative association rules. *Information Sciences*, 258:1–28, 2014.

[76] J. Mata, J. Alvarez, and J. Riquelme. Mining numeric association rules with genetic algorithms. In *Artificial neural nets and genetic algorithms*, pages 264–267. Springer, 2001.

[77] Y. Medjadba, D. Hu, W. Liu, and X. Yu. Combining graph clustering and quantitative association rules for knowledge discovery in geochemical data problem. *IEEE Access*, 8:40453–40473, 2020.

[78] S. Mehta, S. Parthasarathy, and H. Yang. Toward unsupervised correlation preserving discretization. *IEEE Transactions on Knowledge and Data Engineering*, 17(9):1174–1185, 2005.

[79] R. J. Miller and Y. Yang. Association rules over interval data. In *SIGMOD '97*, page 452–461, New York, NY, USA, 1997. Association for Computing Machinery.

[80] H. Mohamadlou, R. Ghodsi, J. Razmi, and A. Keramati. A method for mining association rules in quantitative and fuzzy data. In *2009 International Conference on Computers Industrial Engineering*, pages 453–458, France, 2009. IEEE.

[81] D. C. Montgomery, E. A. Peck, and G. G. Vining. *Introduction to linear regression analysis*. John Wiley & Sons, 2021.

[82] K. Pearson. Vii. note on regression and inheritance in the case of two parents. *proceedings of the royal society of London*, 58(347-352):240–242, 1895.

[83] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee. A design science research methodology for information systems research. *Journal of Management Information Systems*, 24(3):45–77, 2007.

[84] S. A. Peious, R. Sharma, M. Kaushik, S. A. Shah, and S. B. Yahia. Grand reports: a tool for generalizing association rule mining to numeric target values. In *International Conference on Big Data Analytics and Knowledge Discovery*, pages 28–37. Springer, 2020.

[85] J. R. Quinlan. Induction of decision trees. *Machine learning*, 1(1):81–106, 1986.

[86] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi. Gsa: a gravitational search algorithm. *Information sciences*, 179(13):2232–2248, 2009.

[87] R. Rastogi and K. Shim. Mining optimized association rules with categorical and numeric attributes. *IEEE Transactions on Knowledge and Data Engineering*, 14(1):29–50, 2002.

[88] M. Shahin, S. Arakkal Peious, R. Sharma, M. Kaushik, S. Ben Yahia, S. A. Shah, and D. Draheim. Big data analytics in association rule mining: A systematic literature review. In *International Conference on Big Data Engineering and Technology (BDET)*, page 40–49. Association for Computing Machinery, 2021.

[89] M. Shahin, M. R. Heidari Iman, M. Kaushik, R. Sharma, T. Ghasempouri, and D. Draheim. Exploring factors in a crossroad dataset using cluster-based association rule mining. *Procedia Computer Science*, 201:231–238, 2022. The 13th International Conference on Ambient Systems, Networks and Technologies (ANT) / The 5th International Conference on Emerging Data and Industry 4.0 (EDI40).

[90] M. Shahin, S. Saeidi, S. A. Shah, M. Kaushik, R. Sharma, S. A. Peious, and D. Draheim. Cluster-based association rule mining for an intersection accident dataset. In *2021 International Conference on Computing, Electronic and Electrical Engineering (ICE Cube)*, pages 1–6, 2021.

[91] R. Sharma, H. Garayev, M. Kaushik, S. A. Peious, P. Tiwari, and D. Draheim. Detecting simpson's paradox: A machine learning perspective. In C. Strauss, A. Cuzzocrea, G. Kotsis, A. M. Tjoa, and I. Khalil, editors, *Proceedings of DEXA 2022 – the 33rd International Conference on Database and Expert Systems Applications*, pages 323–335, Cham, 2022. Springer International Publishing.

[92] R. Sharma, M. Kaushik, and D. Draheim. On statistical paradoxes and overcoming the impact of bias in artificial intelligence: towards fair and trustworthy decision making. *SSRN Electronic Journal*, pages 1–107, 2022.

[93] R. Sharma, M. Kaushik, S. A. Peious, A. Bazin, S. A. Shah, I. Fister, S. B. Yahia, and D. Draheim. A novel framework for unification of association rule mining, online analytical processing and statistical reasoning. *IEEE Access*, 10:12792–12813, 2022.

[94] R. Sharma, M. Kaushik, S. A. Peious, M. Bertl, A. Vidyarthi, A. Kumar, and D. Draheim. Detecting simpson's paradox: A step towards fairness in machine learning. In S. Chiusano, T. Cerquitelli, R. Wrembel, K. Nørvåg, B. Catania, G. Vargas-Solar, and E. Zumpano, editors, *Proceedings of ADBIS 2022 – the 26th International Conference on New Trends in Database and Information Systems*, pages 67–76, Cham, 2022. Springer International Publishing.

[95] R. Sharma, M. Kaushik, S. A. Peious, M. Shahin, A. Vidyarthi, and D. Draheim. Existence of the yule-simpson effect: An experiment with continuous data. In *Proceedings of Confluence 2022 – the 12th International Conference on Cloud Computing, Data Science & Engineering*, pages 351–355, 2022.

[96] R. Sharma, M. Kaushik, S. A. Peious, M. Shahin, A. Vidyarthi, P. Tiwari, and D. Draheim. Why not to trust big data: Discussing statistical paradoxes. In U. K. Rage, V. Goyal, and P. K. Reddy, editors, *Proceedings of DASFAA 2022 International Workshops – the 27th International Conference on Database Systems for Advanced Applications*, pages 50–63, Cham, 2022. Springer International Publishing.

[97] R. Sharma, M. Kaushik, S. A. Peious, M. Shahin, A. S. Yadav, and D. Draheim. Towards unification of statistical reasoning, olap and association rule mining: Semantics and pragmatics. In A. Bhattacharya, J. Lee Mong Li, D. Agrawal, P. K. Reddy, M. Mohania, A. Mondal, V. Goyal, and R. Uday Kiran, editors, *Proceedings of DASFAA 2022 – the 27th International Conference on Database Systems for Advanced Applications*, pages 596–603, Cham, 2022. Springer International Publishing.

[98] R. Sharma, M. Kaushik, S. A. Peious, S. B. Yahia, and D. Draheim. Expected vs. unexpected: Selecting right measures of interestingness. In M. Song, I.-Y. Song, G. Kotsis, A. M. Tjoa, and I. Khalil, editors, *Proceedings of DaWaK 2020 – the 22nd International Conference on Big Data Analytics and Knowledge Discovery*, pages 38–47, Cham, 2020. Springer International Publishing.

[99] C. Song and T. Ge. Discovering and managing quantitative association rules. In *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management*, CIKM '13, page 2429–2434, New York, NY, USA, 2013. Association for Computing Machinery.

[100] R. Srikant and R. Agrawal. Mining quantitative association rules in large relational tables. In *Proceedings of the 1996 ACM SIGMOD international conference on Management of data*, pages 1–12, Canada, 1996. ACM.

[101] R. Srikant and R. Agrawal. Mining generalized association rules. *Future Generation Computer Systems*, 13(2):161–180, 1997. Data Mining.

[102] S. M. Stigler. Francis galton's account of the invention of correlation. *Statistical Science*, pages 73–79, 1989.

[103] T. Takagi and M. Sugeno. Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-15(1):116–132, 1985.

[104] A. Tatu, P. Bak, E. Bertini, D. Keim, and J. Schneidewind. Visual quality metrics and human perception: An initial study on 2d projections of large multidimensional data. In *Proceedings AVI'10*, page 49–56, New York, NY, USA, 2010. Association for Computing Machinery.

[105] R. L. Thorndike. Who belongs in the family? *Psychometrika*, 18(4):267–276, 1953.

[106] K. Wang, S. H. W. Tay, and B. Liu. Interestingness-based interval merger for numeric association rules. In *KDD*, volume 98, pages 121–128, New York, 1998. AAAI Press.

[107] G. I. Webb. Discovering associations with numeric variables. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 383–388, San Francisco California, 2001. ACM.

[108] C.-M. Wu and Y.-F. Huang. Generalized association rule mining using an efficient data structure. *Expert Systems with Applications*, 38(6):7277–7290, 2011.

[109] J. Yang and Z. Feng. An effective algorithm for mining quantitative associations based on subspace clustering. In *2010 International Conference on Networking and Digital Society*, volume 1, pages 175–178, China, 2010. IEEE.

[110] W. Zhang. Mining fuzzy quantitative association rules. In *Proceedings 11th International Conference on Tools with Artificial Intelligence*, pages 99–102, USA, 1999. IEEE.

[111] H. Zheng, J. He, G. Huang, and Y. Zhang. Optimized fuzzy association rule mining for quantitative data. In *2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 396–403, China, 2014. IEEE.

Acknowledgements

I would like to express my deepest gratitude to my supervisor, Dirk Draheim, for his exceptional guidance, unwavering support, and invaluable advice throughout my doctoral studies. His vast knowledge, extensive research experience, and genuine dedication to my academic growth have been instrumental in shaping my research journey. I am profoundly grateful for his mentorship and the opportunities he has provided me to expand my knowledge and skills in the field.

I extend my sincere appreciation to Tallinn University of Technology for providing me with the necessary resources, facilities, and opportunities to pursue my research. The institutional support I received has enabled me to carry out my studies and contribute to the field.

I want to take a moment to express my heartfelt gratitude to my beloved parents, supportive husband, and lovely daughter and mother-in-law. Their unwavering encouragement, understanding, and unwavering belief in my abilities have been the driving force behind my academic achievements. Their constant presence, love, and support have been my anchor, giving me the strength and motivation to overcome challenges and pursue my research goals. I am forever grateful for their sacrifices and unwavering commitment to my success.

Abstract
Generalized Association Rule Mining – Dimensional Unsupervised Learning

This thesis explores the synergistic use of numerical association rule mining and order-preserving partitioning methods to uncover the partitions of numerical attributes that reflect the most significant impact of an independent numerical attribute on a dependent numerical attribute. The key objective of this research work is to contribute to developing an ecosystem that elevates machine-learning approaches by refining the dimensions of the ARM. The thesis addresses three main research questions and seven sub-research questions by providing valuable insights and practical solutions for working with numerical attributes within the association rule mining domain. To fulfill these objectives, the thesis draws upon five peer-reviewed articles and three technical reports published between 2020 and 2023. This thesis provides its first contribution through an exhaustive SLR encompassing over 1,140 scholarly articles published since 1996. The review focuses on NARM, thereby identifying prevalent problems and proposing corresponding solutions within this research domain. The second contribution is demonstrating the significance and importance of human perception for discretizing numeric attributes. This research aimed to investigate the impact of data points' features on human perception when partitioning numerical attributes by experimenting with data experts and non-experts. The third significant contribution of this thesis presents two novel measures for partitioning numerical attributes. The development of these measures provides a new approach to effectively partition numerical attributes in decision-making processes. The fourth meaningful contribution entails conducting an analytical evaluation of the measures based on collected human perception responses. This thesis conducts a comprehensive investigation into human perception of numerical factor partitioning and undertakes a thorough comparison with the proposed measures. As a part of future work, It is planned to extend this research further to improve the effectiveness of the proposed measures.

Kokkuvõte
Üldistatud assotsiatsioonireeglite kaevandamine – dimensiooniline juhendamata õpe

See lõputöö uurib numbriliste assotsatsioonireeglite kaevandamise (NARM) ja järjestust säilitavate jaotusmeetodite sünergilist kasutamist, et esile tuua numbriliste atribuutide sektsioonid, mis kajastavad sõltumatu numbrilise atribuudi kõige olulisemat mõju sõltuvale numbrilisele atribuudile. Selle uurimistöö põhieesmärk on aidata kaasa sellise ökosüsteemi arendamisele, mis edenab masinõppe lähenemisviise täpsustades ARM-i ulatust. Lõputöö hõlmab kolme peamist uurimisküsimust ja kuut abistavat uurimiküsimust, pakkudes väärtuslikke teadmisi ja praktilisi lahendusi numbriliste atribuutidega töötamiseks assotsiatsioonireeglite kaevandamise valdkonnas. Nende eesmärkide täitmiseks toetub lõputöö viiele eelretsenseeritud artiklile ja kolmele tehnilisele aruandele, mis avaldati aastatel 2020–2023. See väitekiri annab oma esimese panuse ammendava süstemaatilise erialakirjanduse ülevaate kaudu, mis hõlmab üle 1140 teadusartikli, mis on avaldatud alates 1996. aastast. Ülevaade keskendub NARM-ile, tuvastades esinevad probleemid ja pakkudes vastavad lahendused selles uurimisvaldkonnas. Teine panus näitab inimtaju olulisust ja tähtsust numbriliste atribuutide diskretiseerimisel. Selle uuringu eesmärgiks oli tuvastada andmepunktide funktsioonide mõju inimese tajule numbriliste atribuutide jagamisel, katsetades andmeekspertide ja mitteekspertide rühmadega. Selle lõputöö kolmas oluline panus esitab kaks uudset meedet numbriliste atribuutide jaotamiseks. Nende meetmete väljatöötamine annab uue lähenemisviisi numbriliste atribuutide tõhusaks jaotamiseks otsustusprotsessides. Neljas oluline panus hõlmab meetmete analüütilist hindamist kogutud inimtaju vastuste põhjal. Selle uuringu käigus uuriti põhjalikult, kuidas inimene tajub arvuliste tegurite jaotust, ja võrdleb põhjalikult kavandatud meetmetega. Edasise töö osana on kavas antud uuringut veelgi laiendada, et suurendada väljapakutud meetmete tõhusust.

Appendix 1

[I]

M. Kaushik, R. Sharma, S. A. Peious, M. Shahin, S. B. Yahia, and D. Draheim. On the potential of numerical association rule mining. In *Proceedings of FDSE: 7th International Conference on Future Data and Security Engineering*, pages 3–20, Vietnam, 2020. Springer

On the Potential of Numerical Association Rule Mining

Minakshi Kaushik[1] , Rahul Sharma[1], Sijo Arakkal Peious[1], Mahtab Shahin[1], Sadok Ben Yahia[2], and Dirk Draheim[1]

[1] Information Systems Group, Tallinn University of Technology, Akadeemia tee 15a, 12618 Tallinn, Estonia
{minakshi.kaushik,rahul.sharma,sijo.arakkal,mahtab.shahin, dirk.draheim}@taltech.ee

[2] Software Science Department, Tallinn University of Technology, Akadeemia tee 15a, 12618 Tallinn, Estonia
sadok.ben@taltech.ee

Abstract. In association rule mining, both the classical algorithms and today's available tools either use binary data items or discretized data. However, in real-world scenarios, data are available in many different forms (numerical, text) and these types of data items are not supported in the classical association rule mining algorithms. There are some association rule mining algorithms that have been proposed for numerical data items but unfortunately, for working data scientists and decision makers, it is challenging to find concrete algorithms that fit their purposes best. Therefore, it is highly desired to have a study on the different existing numerical association rule mining algorithms (NARM). In this paper, we provide such a detailed study by thoroughly reviewing 24 NARM algorithms from different categories (optimization, discretization, distribution).

Keywords: Knowledge discovery in databases · Association rule mining · Numerical association rule mining

1 Introduction

Data mining is a widely used technique for extracting useful information from large repositories of data. To extract useful information from data, there are many well-known data mining techniques such as association rule mining, characterization, classification, clustering, evolution, generalization, regression, prediction, outlier detection, etc. that have been proposed in the literature. Out of all the data mining techniques, association rule mining (ARM) is one of the most established ones.

ARM was first introduced by Agrawal [2] to understand the relationship between different data items and since then has been widely used for market basket analysis, bio-informatics, medical diagnosis, etc. Agrawal [3] proposed the apriori algorithm to discover all significant association rules in large databases

T. K. Dang et al. (Eds.): FDSE 2020, CCIS 1306, pp. 3–20, 2020.
https://doi.org/10.1007/978-981-33-4370-2_1

in 1994. The main aim of ARM is not just finding frequent itemsets but also finding interesting association rules.

In classical association rule mining, most of the algorithms work in two phases. In the first phase, all frequent itemsets are found, and in the second phase, rules are drawn. Apriori and FP-growth are the two most algorithms based on binary columns and are usually perceived as the classical association rule mining algorithms. The classical association rule mining algorithms work only with binary data items and do not support numerical data items, therefore, whenever data is in numerical form (height, weight, or age) the data items need to be changed from *numerical* to *categorical* using a discretization process. This process of finding association rules in numerical data items has been referred to as numerical association rule mining (NARM).

Research in the area of association rule mining generally considers binary data items as input for the proposed algorithms but excludes numerical data sets. A tool named Grand report has been proposed that reports mean values of a chosen numeric target column concerning all possible combinations of influencing factors [45]. There are some association rule mining algorithms available for numerical data items but it is still challenging to find the best algorithms, NARM algorithms have the potential to deal with different types of attributes, therefore, it's important to have a study on different numerical association rule mining algorithms.

In this paper, we discussed different solutions and problems in the 24 NARM algorithms proposed under the optimization, discretization and distribution methods. The paper is structured as follows. In Sect. 2, we describe preliminaries. In Sect. 3, we discuss all three methods to solve numerical association rule mining problems. In Sect. 4, the optimization method is discussed with all its sub-methods. In Sect. 5, the distribution method is introduced and discussed and in Sect. 6 Discretization method is discussed. We finish the paper with a conclusion in Sect. 7.

2 Preliminaries

In ARM, association rules have been developed based on the If-then relations which consist of antecedents (If) and consequents (Then) [2]. For example, (1) shows the following association rule: "If a customer buys bread and butter then he also buys milk and sugar". Here, *bread and butter* appear as antecedent and *milk and sugar* as consequent. Generally, an association rule may be represented as a production rule in an expert system, an *if statement* in a programming language or implication in a logical calculus.

$$\{Bread, Butter\} \Rightarrow \{Milk, Sugar\} \tag{1}$$

In a database, let I be a set of m binary attributes $\{i_1, i_2, i_3, \ldots, i_m\}$ called database items. Let T be a set of n transactions $\{t_1, t_2, t_3, \ldots, t_n\}$, where each transaction t_i has a unique ID and consists of a subset of the items in I, i.e., $t_i \subseteq I$. As in (1), an association rule is an implication of the form

$$X \Rightarrow Y \tag{2}$$

where $X, Y \subseteq I$ (itemsets) and $X \cap Y = \emptyset$.

In association rule mining, frequent itemsets and association rules are discovered based on boolean data columns, therefore, it is known as boolean association rule mining. Different measures of interestingess are proposed in the literature to find out the interesting rules [53]. Boolean association rules are meaningful, but data are often available in different forms (categorical, quantitative, text) and in these cases, boolean association rule mining techniques do not fit. Thus, the term numeric association rule mining was introduced by [26] and the problem was first discussed by Srikant in 1996 [55]. A numerical association rule can easily be understood by the following example.

$$\mathit{Age} \in [40, 50] \wedge \mathit{Gender} = M \Rightarrow \mathit{NumberOfCars} = 2 \tag{3}$$

Given a set of transactions T, let *Antecedent* denote the set of transactions in T in which *Age* has a value between 40 and 50 and *Gender* equals M. Similarly, let *Consequent* denote the set of transactions in which $\mathit{NumberOfCars} = 2$. Now, the association rule (3) stands for the following fraction.

$$\frac{\text{number of transcations in } \mathit{Antecedent} \cap \mathit{Consequent}}{\text{number of transcations in } \mathit{Antecedent}} \tag{4}$$

As an early solution, the problem of association rules for numerical data was solved using a discretization process where numeric attributes are divided into different intervals and, henceforth, these attributes are treated as categorical attributes [12]. For example, an attribute *Age* with values between 20 to 80 can be divided into six different age intervals (20–30, 30–40, 40–50, 50–60, 60–70, 70–80). The data discretization process is an obvious solution, however, it reveals a loss of valuable information which might cause poor results [16]. Thus, we review solutions from three different approaches (discretization, distribution and optimization) to solve issues with numerical association rule mining in Sect. 3.

3 Methods to Solve Numerical ARM Problems

To solve the issues in numerical association rule mining, three main approaches (discretization, distribution and optimization) have been discussed in the literature. Based on these three approaches, many different NARM algorithms have been proposed. The optimization method has several sub-methods as swarm intelligence and evolution based algorithms which cover most of the area to deal with NARM. The Distribution method does not contribute much in this area, however, the discretization method is a common method that transforms continuous attributes into discrete attributes. The discretization is further subdivided into three sub-methods. Figure 1 (compare also with Fig. 1 in [9]) is showing all three approaches and different algorithms proposed under each approach.

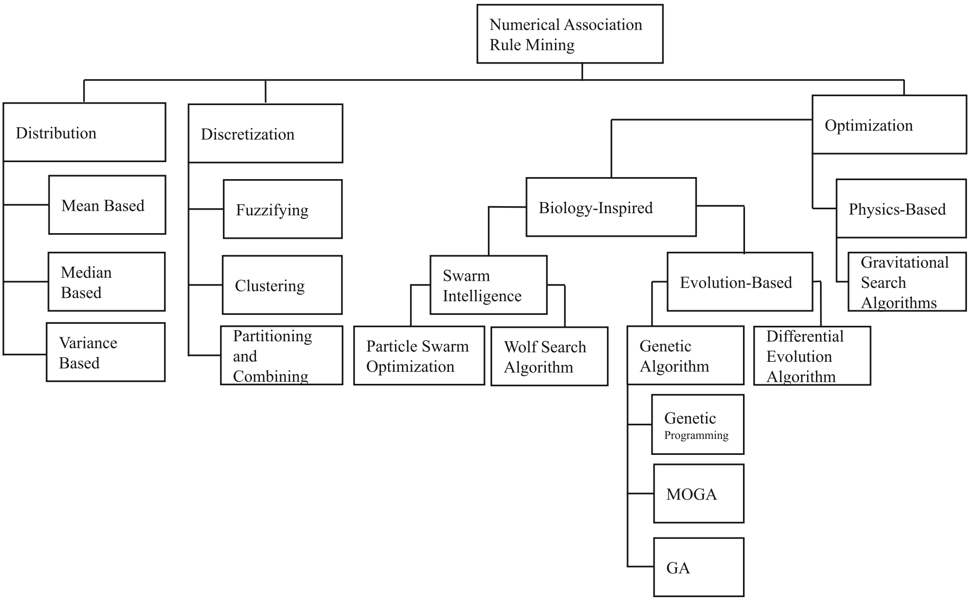


Fig. 1. Methods to solve numerical association rule mining problems.

4 The Optimization Method

To solve the numerical association rule mining problem, many researchers have moved towards optimization methods. Optimization methods provide a robust and efficient approach to explore a massive search space. In this method, researchers have invented a collection of heuristic optimization methods inspired by the movements of animals and insects. For finding association rules, optimization methods work in two phases. In the first phase, all the frequent itemsets are found and in the second phase, all relevant association rules are extracted. As shown in Fig. 1, optimization methods are divided into two parts, bio-inspired optimization methods and physics-based optimization methods. Table 1 shows an overview of all those algorithms that come under the optimization method.

4.1 The Bio-Inspired Optimization Method

Biology-based algorithms are generally divided into two parts: swarm-intelligence-based algorithms and evolution-based algorithms [14]. The origin of these algorithms is the biological behavior of natural objects [64].

Evolution-Based Algorithms. Evolution-based algorithms are inspired by Darwinian principles and were first applied in [42]. These algorithms mimic the capability of nature to develop living beings that are well-adapted to their environment [64]. Evolution-based algorithms exploit stochastic search methods that follow the idea of natural selection and genetics. The algorithms show strong

Table 1. An overview of optimization method algorithms for NARM.

Methods	Basic technique	Algorithms
GA	Genetic Algorithm	GENAR [41], GAR [42], EGAR [33], ARMGA, EARMGA [68], GAR-PLUS [10], QuantMiner [51], RelQM-J [52], RCGA [40]
MOGA	Genetic Algorithm	ARMMGA [47], QAR-CIP-NSGA-II [39], MOEA [20]
DE	Differential Evolution	MODENAR [7], ARM-DE [18]
PSO	Particle Swarm Optimization	RPSO [4], CENPSO [5], MOPAR [12], PPQAR [67], PARCD [61]
WSA	Swarm Intelligence	WSA [1]
GSA	Physics-based (Gravity)	GSA [13]

adaptability and self organization [14] and use biology-inspired operators such as crossover, mutation, and natural selection [64]. The *Genetic Algorithm* [25] and the *Differential Evolution Algorithm* [58] are two example of evolution-based algorithms. Table 2 shows an overview of the evolution-based algorithms for NARM, together with concepts.

Genetic Algorithms (GA). GA was first proposed by Holland [25] and they are one of the most popular algorithms in bio-inspired optimization methods at all. A basic genetic algorithm consists of five phases: initialization, evaluation, reproduction, crossover, and mutation. GAs for NARM can be divided into three fields, i.e., basic genetic algorithms, genetic programming and multiobjective genetic algorithms. A basic genetic algorithm has been proposed by Mata et al. [41] and together with the tool GENAR (GENetic Association Rules) to discover association rule with numeric attributes. Association rules in GENAR algorithms allow for intervals (maximum and minimum values) for each numeric attribute. Mata et al. [42] further extended the GENAR algorithms and proposed a technique named GAR (Genetic Association Rule) to discover association rules in numeric databases without discretization. In this paper, a genetic algorithm was used to find the suitable amplitude of the intervals that conform k-itemset and can have a high support value without too wide intervals. In [33], the GAR algorithm was further extended to EGAR (extended genetic association rule). This algorithm generates frequent patterns with continuous data [42].

A genetic-based strategy and two other algorithms ARMGA and EARMGA were proposed by Yan et al. [68] In this approach, an encoding method was developed with relative confidence as the fitness function. In these algorithms, there was no requirement of a minimum support threshold. The GAR-plus tool was presented by Alvarez [10]. This tool deals with categorical and numeric attributes in large databases without any need of a prior discretization of numeric attributes.

Based on the genetic algorithm, in 2013, Salleb et al. [51] proposed "QuantMiner", a quantitative association rule mining system. This tool dynamically discovers meaningful intervals in association rules by optimizing both the confidence and the support values.

Seki and Nagao [52] worked on GA-based QuantMiner for multi-relational data mining and developed RelQM-J, a tool for relational quantitative association rules5 in Java programming language. In this tool, efficient computation of the support of the rules has been realized by using a hash-based data structure.

A real-coded [30] genetic algorithm was presented in [40] in 2010. The proposed algorithm RCGA follows the CHC binary-coded evolutionary algorithm [17]. RCGA algorithm has been applied to pollutant agent time series and helps to find all existing relations between atmospheric pollution and climatological conditions.

Genetic Programming for ARM. Genetic Programming [31] is a well-known type of GA. In GA, the genome is in string structure while in GP, the genome is in the form of tree structure [24]. Genetic Network Programming (GNP) is a graph-based evolutionary algorithm and find the association rules for continuous attributes. In this method, important rules are stored in a pool and these extracted rules are measured by the chi-squared test. This pool is updated in every generation by exchanging the association rule with higher chi-squared value for the same association rule with lower chi-squared value [59].

Multi-Objective Genetic Algorithm. The multi-objective genetic algorithm was proposed by Fonseca et al. in 1993 [19]. Generally, the resource consumption of an association rule mining computation is affected by two parameters, i.e., minimum support and minimum confidence. In classical ARM algorithms, only a single measure (support or confidence) has been used as a measure to evaluate the rule interestingness, therefore, if the values of minimum support and minimum confidence are not set properly then the number of association rules may be very less or it may be very large. This problem can be solved by using more objectives or measures as referred in multi-objective ARM.

Gosh and Nath [20] used a Pareto-based genetic algorithm to solve the multi-objective rule mining problem by using three measures: interestingness, comprehensibility and predictive accuracy. The single objective algorithm, ARMGA [68] had issues that were addressed by introducing the multi-objective genetic algorithm called ARMMGA by Qodmanan et al. in [47]. The ARMGA algorithm finds high confidence and low support rules, whereas ARMMGA finds high confidence and high support rules. ARMGA has a large set of rules in comparison to ARMMGA; this problem was solved using a new fitness function in ARMMGA. To prevent invalid chromosomes in ARMGA, new crossover and mutation operators are presented in the literature.

To solve multi-objective optimization problems, Srinivasan and Deb [56] proposed a non dominated genetic sorting algorithm. In 2002, Deb et al. [15] extended NSGA to NSGA-II. In 2011, Martin et al. [39] extended NSGA-II with a trade-off between interpretability and accuracy. NSGA-II performs evolutionary learning of intervals of attributes. For each rule, condition selection is done for three objectives (interestingness, comprehensibility and performance). This method did not depend on minimum support and confidence thresholds. Martin et al. again extended their research on NSGA-II to a new approach called QAR-CIP-NSGA-II and compared the results of this algorithm with other MOEA algorithms.

Differential Evolutionary Algorithms. Differential evolutionary (DE) algorithms are evolution-based algorithms. These algorithms were proposed by Storn and Price in [57]. DE algorithms are simple and effective single-objective optimization algorithms that solve real-valued problems based on the principle of natural evolution. DE algorithms use Genetic-based operators such as crossover, mutation, and selection. Although the evolution process of DE is similar to the one of GA but it relies on a mutation operator instead of a crossover operator [65].

A pareto-based multi-objective DE algorithm for ARM was first proposed in [7] by Alatas et al. for searching accurate and comprehensible association rules. The problem of mining association rules was formulated with four objective optimization problems, i.e., support, confidence, comprehensibility and amplitude. Here, support, confidence and comprehensibility are maximization objectives and the amplitude of intervals is a minimization objective. In a single run, a pareto-based multi-objective DE algorithm search intervals of numeric attributes and association rules.

In 2018, [18] proposed a novel approach for mining association rules with numerical as well as categorical attributes based on DE. In this algorithm, a single objective optimization problem is considered in which support and confidence of association rules are combined into a fitness function. This new DE using ARM (ARM-DE) with mixed (i.e., numerical and categorical) attributes consist of three stages: 1. domain analysis, 2. representation of a solution, 3. definition of a fitness function.

Swarm Intelligence Based Algorithms are further divided into two sub-optimization methods, particle swarm optimization and the wolf search algorithm. Table 3 provides an overview of swarm intelligence algorithms for NARM.

Particle Swarm Optimization. Particle Swarm Optimization (PSO) is a population-based optimization algorithm for nonlinear function. This algorithm is oriented towards animal behavior such as birds flocking or fish schooling. It was developed in 1995 [27,46]. PSO was first used for NARM to find intervals of the numerical attributes in 2008 [4].

Rough PSOA, based on rough patterns was proposed in [4], in which rough values are defined with upper and lower intervals. This algorithm can complement the existing tools developed in rough computing. Rough values are useful in representing an interval for an attribute. In this work, each particle consists of a decision variable that has three parts. The first part of each decision variable

Table 2. An overview of evolution-based algorithms for NARM.

Algorithm	Proposer	Concept
GENAR [41]	J. Mata Vázquez et al. (2001)	Based on finding frequent itemsets in numerical databases and intervals of all attributes that conform to those frequent itemsets
GAR [42]	J. Mata Vázquez et al. (2002)	Extended version of GENAR
EGAR [33]	H. Kwaśnicka et al. (2006)	Uses medical databases where attributes are continuous and discrete; extended version of GAR
ARMGA [68]	A. Yan et al. (2009)	No requirement of minimum support threshold
EARMGA [68]	A. Yan et al. (2009)	
GARPLUS [10]	V. Álvarez et al. (2012)	Based on the finding intervals of numeric attribute
QUANTMINER [51]	A. Salleb-Aouissi et al. (2013)	Based on genetic algorithm to find good intervals by optimizing both support and confidence
RelQM-J [52]	H. Seki1 (2017)	Based on mining numeric rules from relational databases, implemented in Java
RCGA [40]	M. Martinez-Ballesterosa (2010)	Based on CHC binary-coded evolutionary algorithm
ARMMGA [47]	H. Reza Qodmanan (2011)	Based on multi-objective genetic algorithm
QAR-CIP-NSGA-II	D. Martın et al. (2011)	Based on NSGA with three measures (comprehensibility, interestingness, performance)
MODENAR [7]	B. Alatas (2008)	Based on multi-objective differential evolutionary algorithm
ARM-DE [18]	I. Fister Jr. (2018)	Single objective optimization problem where features consist of numerical as well as categorical attributes

represents the antecedent or consequent of the rule and can take values between 0 and 1. The second part represents the lower bound, the third part represents the upper bound of the item interval. The second and third parts are combined as one rough value during the implementation phase of particle representation.

Table 3. An overview of swarm-intelligence-based algorithms for NARM.

Algorithm	Proposer	Concept
RPSO [4]	B. Alatas et al. (2008)	RPSOA is based on the notion of rough patterns that use rough values defined with upper and lower intervals.
CENPSO [5]	B. Alatas et al. (2009)	CENPSO is based on chaos numbers
MOPAR [12]	V. Beiranvand et al. (2014)	MOPAR is Based on Multi objectives (confidence, comprehensibility and interestingness)
Parallel PSO [67]	A. Yan et al. (2019)	Parallel PSO is based on two methods of parallel algorithm: particle-oriented and data-oriented parallelization
PARCD [61]	I. Tahyudin et al. (2017)	Combined PSO method with cauchy distribution
WSA [1]	I.E. Agbehadji et al. (2016)	Based on wolves hunting strategy

Alatas and Akin [5] proposed a novel PSO algorithm based on chaos numbers. The CENPSOA algorithm (chaotically encoded PSO) uses chaos decision variables and chaos particles. Chaos and PSO relation were first discovered by Liu et al. [36], CENPSOA algorithm performs encoding of particles given by chaos numbers. The Chaos numbers consist of the midpoint and radius part of values [5]. Alatas and Akin [6] also proposed a multi-objective chaotic particle swarm optimization algorithm for mining accurate and comprehensible classification rules.

Yan et al. [67] proposed a parallel PSO algorithm for numerical association rule mining. This parallel algorithm was designed with two strategies called particle-oriented and data-oriented parallelization. Particle-oriented parallelization is more efficient and data-oriented parallelization is more scalable to process large datasets.

To discover association rules in a single step without prior discretization of numerical attributes, Beiranvand et al. [12] proposed a multi-objective particle swarm optimization algorithm (MOPAR). The algorithm defines multiple objectives such as confidence, comprehensibility and interestingness. In the pareto method, a candidate solution is identified better than all other candidates. And in multi-objective optimization, a set of best solutions is identified in which the members are superior among all the candidates.

Kuo et al. [32] proposed a multi-objective particle swarm optimization algorithm using an adaptive archive grid for NARM. It is also based on Pareto optimal strategy. In this algorithm, minimum support and minimum confidence are not required before mining. MOPSO algorithm includes a discretization procedure to process numerical data. This algorithm is executed in three parts: 1. initialization, 2. adaptive archive grid, and 3. particle swarm optimization searching.

PSO for numerical association rule mining with cauchy distribution (PARCD) has been evaluated by [61] and it showed that the result of PARCD is better than the method of MOPAR.

Wolf Search Algorithm. The wolf search algorithm (WSA) is a bio-inspired heuristic optimization algorithm. It was proposed by [63] and imitates the way wolves search for food and survive by avoiding their enemies. WSA is tested and compared with other heuristic algorithms and investigated with respect to its memory requirements. The group of wolves has characteristics of commuting together as a nuclear family, that is why it is different from particle swarm optimization [66].

Agbehadji and Fong [1] proposed a new meta-heuristic algorithm that used the wolf search algorithm for NARM. The wolf has three different features of preying. These are prey initiatively, prey passively and escape. The *preying initiatively* feature allows the wolf to check its visual perimeter to detect prey. If the prey is found within visual distance, the wolf moves towards the prey with the highest fitness value, else, the wolves will maintain its direction. In *prey passively* mode, the wolf only stays alert from threats and tries to improve its position. In the *escape* mode, when a threat is detected, the wolf escapes quickly by relocating itself to a new position with an escape distance that is greater than its visual range.

4.2 Physics-Based Algorithm

The physics-based meta-heuristic optimization algorithm simulates the physical behavior and properties of the matter or follows the laws of physics [14]. For NARM, the gravitational search algorithm is a physics-based meta-heuristic optimization algorithm.

Gravitational Search Algorithm. Rashedi et al. proposed a new optimization algorithm based on the law of gravity and named it gravitational search algorithm (GSA) [48]. Newtonian gravity laws state that "Every particle in the universe attracts every other particle with a force that is directly proportional to the product of their masses and inversely proportional to the square of the distance between them". In GSA, agents act as objects and their performance is evaluated by their mass. Each mass presents a solution and it is expected that masses will be attracted by the heaviest mass. GSA is like a small artificial world of masses obeying the Newtonian laws of gravitation and motion. There are four ways for representing the agents or coding the problem variables. These are continuous (real-valued), binary-valued, discrete, and mixed, which are called GSA variants [49].

Can and Alatas [13] first used GSA for NARM. GSA eliminated the task of finding the minimum values of support and confidence. Automatically mined rules have high confidence and support values. In this work, GSA has been designed to find the numerical intervals of the attributes automatically, i.e., without any *a priori* data process at the time of rule mining. The problem of interactions within attributes has been eliminated with the designed GSA by not selecting one attribute at a time and not evaluating a partially-constructed candidate rule due to its global searching with a population.

5 The Distribution Method

In [11], Aumann and Lindell have introduced a new definition for numerical association rules based on statistical inference theory. In this study, they have implemented several distribution scales including mean, median, and variance. The following example shows the kind of generalization of ARM proposed by the authors.

$$Gender{=}F \Rightarrow Wage{:}mean{=}\$8.50 \quad (overall\ mean\ wage = \$12.60) \tag{5}$$

As the above example shows, the average wage for females was $ 8.50 p/hr. The rule displays that the wage of that group was far less than the average wage; therefore, this rule can be considered useful. They also used the algorithm which identifies repeated item-sets and then calculates the desired statistics for the purpose with respect to repeated itemset. This procedure is restricted by the requirement to store every repeated item-sets in memory throughout repeated itemset generation. Where the data is not sparse, the number of frequent itemsets will be huge and repeated itemset storage and access will dominate the calculation. Moreover, they concluded that the suggested algorithm is beneficial and may find rules between two given quantitative attributes.

6 The Discretization Method

Discretization is a process of quantizing numerical attributes into groups of intervals and it is one of the most popular methods to solve the problem of numerical association rule mining. There are numerous methods of discretization in literature. Due to different needs, discretization methods have been developed in different ways such as supervised vs. unsupervised, dynamic vs. static, global vs. local, splitting (top-down) vs. merging (bottom-up) and direct vs. incremental [37]. In classical ARM algorithms, numerical columns cannot be processed directly [38], i.e., all columns need to be categorical, which is a major limitation of ARM [62].

Discretization of numerical values is used to overcome this problem [28,43,44]. When a numeric column is divided into useful target groups, it becomes easier to identify and generate association rules, i.e. discretization helps to understand the numeric columns better. The discretized groups are useful only if the variables in the same group do not have any objective difference. Discretization minimizes the impact of trivial variations between values. Discretization can be performed using fuzzifying, clustering and *partitioning and combining* [8]. In Table 4, we summarize some selected discretization algorithms used in NARM.

6.1 Fuzzifying

Fuzzy logic is a suitable way of handling numeric value columns for association rule mining systems [50]. A straightforward method is in grouping numeric values of a column by fuzzy sets [8]. Here, *fuzzifying* is the technique of illustrating numeric values as fuzzy sets [29] which can help to rectify the *sharp boundary problem* of association mining [50,60]. Sometimes, endpoint values of discretized groups have more or less influence on the result than the midpoint values: this phenomenon is known as a sharp boundary problem. Fuzzy Class Association Rule (FCAR) is a model proposed by Kianmehr et al. in [29] to get the fuzzy class association rules.

6.2 Clustering

Clustering is one of the popular methods of discretizing a numerical column in an unsupervised manner [8]. In clustering, a numerical column is segregated into different groups according to properties of each value; in this method, the probability of having values in the same group depends on the degree of similarity or dissimilarity of the values [23,54]. To obtain maximum results in clustering, the degree of similarity and dissimilarity needs to be well defined [21]: *"In other words, the intra-cluster variance is to be minimized, and the inter-cluster variance is to be maximized"* [62]. Two-step clustering [54] is the most common clustering method.

DRMiner Algorithm. Lian et al. [35] have proposed the DRMiner algorithm which exploits the notion of "density" to capture the characteristics of numeric attributes and an efficient procedure to locate the "dense regions". DRMiner scales up well with high-dimensional datasets. When mapping a database to a multidimensional space, the data points (transactions) are not distributed evenly throughout the multidimensional space. For this kind of distribution, the density measure was introduced and the problem of mining quantitative association rules transformed into the problem of finding dense regions to map them to find quantitative association rules. Weaknesses of this method were the prior requirement of many thresholds and, unsolving the dimensionality curse. It was noted that the algorithm may not perform well for data sets with uniform density between minimum density threshold and low density.

DBSMiner. DBSMiner is a density-based sub-space mining algorithm using the notion of density-connected to cluster the high-density sub-space of numeric attributes and gravitation between grid/cluster to deal with the low-density cells

[22]. DBSMiner employs an efficient high dimension clustering algorithm CBSD (Clustering Based on Sorted Dense unit) to deal with high dimensional data sets. The algorithm has a unique feature to deal with low-density sub-spaces and there is no need to scan the whole space just check the neighbor cell. It can find interesting association rules.

MQAR. MQAR (Mining Quantitative Association Rules based on a dense grid) is a novel algorithm that was proposed by Yang and Zhang [69]. The main objective of this algorithm was to mine the numeric association rules using a tree structure, DGFP-tree to cluster dense space. This algorithm is helpful to eliminate noise and redundant rules by transforming the problem into finding regions with enough density and to map them to quantitative association rules. A novel subspace clustering algorithm was also proposed which is based on searching DGFP-tree and inserts the dense cell in the database space into DGFP-tree as a path from a root node to a leaf node. MQAR has the advantage that DGFP-tree compresses the database and there is no need to scan the database several times.

ARCS. The Association Rule Clustering System [34] was presented by Lent et al. together with a new geometric-based clustering algorithm, BitOP. In this paper the problem of clustering of association rules like $(A \wedge B) => C$ where L.H.S. having quantitative attributes and R.H.S. having a categorical attribute was discussed and a two-dimensional grid is formed where each axis represents one of the L.H.S. attributes. ARCS is an automated system to compute a clustering of two-attribute spaces in large databases. In ARCS framework Binner, For a given partitioning of the input attributes, the algorithm makes only one pass through the data and allows the support or confidence thresholds to change without requiring a new pass through the data. BitOp algorithm enumerates the clusters. To locate clusters within bitmap grids the algorithm performs bit-wise operations.

6.3 Partitioning and Combining

In [55], Srikant and Agrawal discussed the problems of numeric attributes in databases. The authors addressed the problem of mining association rules from large databases containing both numerical and categorical attributes. To deal with this problem, a partitioning method was introduced but before partitioning, a measure of partial completeness was introduced which decided whether or not to partition a numeric attribute and number of partitions. The number of required partitions is computed by the following formula.

$$number\ of\ intervals = \frac{2n}{m(K-1)} \tag{6}$$

where n is number of numeric attributes, m is the minimum support and K is the partial completeness level.

Table 4. An overview of discretization-based algorithms for NARM.

Algorithm	Proposer	Concept
ARCS [34]	B. Lent et al. (1997)	Based on segmenting clusters using the geometric-based BitOp algorithm
DRMiner [35]	W. Lian (2005)	Based on finding density regions in a multidimensional space.
DBSMiner [22]	G. Yunkai et al. (2008)	Based on clustering of high density sub-spaces using a density- and grid-based cluster algorithm
MQAR [69]	Y. Junrui et al. (2010)	Based on finding dense sub-spaces using structure DGFP-tree

7 Conclusion

In this paper, a study of 24 NARM algorithms has been discussed. We briefly discussed different solutions and problems in optimization, discretization and distribution methods of solving the NARM problem. As per our findings, many algorithms have been proposed in the optimization method but there is less focused research in the area of discretization and distribution methods. NARM has huge potential to extend dimensions of classical ARM and it may be used for mining association rules in different types of data(categorical, quantitative, text, etc.).

Acknowledgments. This work has been conducted in the project “ICT programme” which was supported by the European Union through the European Social Fund.

References

1. Agbehadji, I.E., Fong, S., Millham, R.: Wolf search algorithm for numeric association rule mining. In: 2016 IEEE International Conference on Cloud Computing and Big Data Analysis (ICCCBDA), pp. 146–151. IEEE (2016)
2. Agrawal, R., Imieliński, T., Swami, A.: Mining association rules between sets of items in large databases. ACM SIGMOD Rec. **22**(2), 207–216 (1993). https://doi.org/10.1145/170036.170072
3. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules in large databases. In: Proceedings of VLDB 1994 - the 20th International Conference on Very Large Data Bases, pp. 487–499. Morgan Kaufmann (1994)
4. Alatas, B., Akin, E.: Rough particle swarm optimization and its applications in data mining. Soft Comput. **12**(12), 1205–1218 (2008)
5. Alatas, B., Akin, E.: Chaotically encoded particle swarm optimization algorithm and its applications. Chaos Solitons Fract. **41**(2), 939–950 (2009)
6. Alatas, B., Akin, E.: Multi-objective rule mining using a chaotic particle swarm optimization algorithm. Knowl. Based Syst. **22**(6), 455–460 (2009)

7. Alatas, B., Akin, E., Karci, A.: MODENAR: multi-objective differential evolution algorithm for mining numeric association rules. Appl. Soft Comput. **8**(1), 646–656 (2008)
8. Altay, E.V., Alatas, B.: Performance analysis of multi-objective artificial intelligence optimization algorithms in numerical association rule mining. J. Amb. Intel. Hum. Comp. **11**, 1–21 (2019)
9. Altay, E.V., Alatas, B.: Intelligent optimization algorithms for the problem of mining numerical association rules. Physica A Stat. Mech. Appl. **540**, 123142 (2020)
10. Álvarez, V.P., Vázquez, J.M.: An evolutionary algorithm to discover quantitative association rules from huge databases without the need for an a priori discretization. Expert Syst. Appl. **39**(1), 585–593 (2012)
11. Aumann, Y., Lindell, Y.: A statistical theory for quantitative association rules. J. Intell. Inf. Syst. **20**(3), 255–283 (2003)
12. Beiranvand, V., Mobasher-Kashani, M., Bakar, A.A.: Multi-objective PSO algorithm for mining numerical association rules without a priori discretization. Expert Syst. Appl. **41**(9), 4259–4273 (2014)
13. Can, U., Alatas, B.: Automatic mining of quantitative association rules with gravitational search algorithm. Int. J. Softw. Eng. Knowl. Eng. **27**(03), 343–372 (2017)
14. Cui, Y., Geng, Z., Zhu, Q., Han, Y.: Multi-objective optimization methods and application in energy saving. Energy **125**, 681–704 (2017)
15. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
16. Djenouri, Y., Bendjoudi, A., Djenouri, D., Comuzzi, M.: GPU-based bio-inspired model for solving association rules mining problem. In: 2017 25th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP), pp. 262–269. IEEE (2017)
17. Eshelman, L.J.: The CHC adaptive search algorithm: how to have safe search when engaging in nontraditional genetic recombination. In: Foundations of Genetic Algorithms, vol. 1, pp. 265–283. Elsevier (1991)
18. Fister, I., Iglesias, A., Galvez, A., Del Ser, J., Osaba, E., Fister, I.: Differential evolution for association rule mining using categorical and numerical attributes. In: Yin, H., Camacho, D., Novais, P., Tallón-Ballesteros, A.J. (eds.) IDEAL 2018. LNCS, vol. 11314, pp. 79–88. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-03493-1_9
19. Fonseca, C.M., Fleming, P.J., et al.: Genetic algorithms for multiobjective optimization: formulation discussion and generalization. In: ICGA, vol. 93, pp. 416–423. CiteSeer (1993)
20. Ghosh, A., Nath, B.: Multi-objective rule mining using genetic algorithms. Inf. Sci. **163**(1–3), 123–133 (2004)
21. Grabmeier, J., Rudolph, A.: Techniques of cluster algorithms in data mining. Data Mining Knowl. Disc. **6**(4), 303–360 (2002)
22. Guo, Y., Yang, J., Huang, Y.: An effective algorithm for mining quantitative association rules based on high dimension cluster. In: 2008 4th International Conference on Wireless Communications, Networking and Mobile Computing, pp. 1–4. IEEE (2008)
23. Han, J., Pei, J., Kamber, M.: Data Mining: Concepts and Techniques. Elsevier, Amsterdam (2011)
24. Hirasawa, K., Okubo, M., Katagiri, H., Hu, J., Murata, J.: Comparison between genetic network programming (GNP) and genetic programming (GP). In: Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No. 01TH8546), vol. 2, pp. 1276–1282. IEEE (2001)

25. Holland, J.H.: Adaption in Natural and Artificial Systems. An Introductory Analysis with Application to Biology, Control and Artificial Intelligence. MIT Press, Cambridge (1975)
26. Ke, Y., Cheng, J., Ng, W.: MIC framework: an information-theoretic approach to quantitative association rule mining. In: 22nd International Conference on Data Engineering (ICDE 2006), p. 112. IEEE (2006)
27. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of ICNN 1995-International Conference on Neural Networks, vol. 4, pp. 1942–1948. IEEE (1995)
28. Khade, R., Patel, N., Lin, J.: Supervised dynamic and adaptive discretization for rule mining. In: 2015 in SDM Workshop on Big Data and Stream Analytics (2015)
29. Kianmehr, K., Alshalalfa, M., Alhajj, R.: Fuzzy clustering-based discretization for gene expression classification. Knowl. Inf. Syst. **24**(3), 441–465 (2010)
30. Kim, H., Adeli, H.: Discrete cost optimization of composite floors using a floating-point genetic algorithm. Eng. Opt. **33**(4), 485–501 (2001)
31. Koza, J.R., Koza, J.R.: Genetic Programming: On the Programming of Computers by Means of Natural Selection, vol. 1. MIT press, Cambridge (1992)
32. Kuo, R., Gosumolo, M., Zulvia, F.E.: Multi-objective particle swarm optimization algorithm using adaptive archive grid for numerical association rule mining. Neural Comput. Appl. **31**(8), 3559–3572 (2019)
33. Kwaśnicka, H., Świtalski, K.: Discovery of association rules from medical data-classical and evolutionary approaches. Annales Universitatis Mariae Curie-Sklodowska, sectio AI-Informatica **4**(1), 204–217 (2006)
34. Lent, B., Swami, A., Widom, J.: Clustering association rules. In: Proceedings 13th International Conference on Data Engineering, pp. 220–231. IEEE (1997)
35. Lian, W., Cheung, D.W., Yiu, S.: An efficient algorithm for finding dense regions for mining quantitative association rules. Comput. Math. Appl. **50**(3–4), 471–490 (2005)
36. Liu, H., Abraham, A., Li, Y., Yang, X.: Role of chaos in swarm intelligence — a preliminary analysis. In: Tiwari, A., Roy, R., Knowles, J., Avineri, E., Dahal, K. (eds.) Applications of Soft Computing. AISC, vol. 36, pp. 383–392. Springer, Heidelberg (2006). https://doi.org/10.1007/978-3-540-36266-1_37
37. Liu, H., Hussain, F., Tan, C.L., Dash, M.: Discretization: an enabling technique. Data Min. Knowl. Disc. **6**(4), 393–423 (2002)
38. Lud, M.-C., Widmer, G.: Relative unsupervised discretization for association rule mining. In: Zighed, D.A., Komorowski, J., Żytkow, J. (eds.) PKDD 2000. LNCS (LNAI), vol. 1910, pp. 148–158. Springer, Heidelberg (2000). https://doi.org/10.1007/3-540-45372-5_15
39. Martín, D., Rosete, A., Alcalá-Fdez, J., Herrera, F.: A multi-objective evolutionary algorithm for mining quantitative association rules. In: 2011 11th International Conference on Intelligent Systems Design and Applications, pp. 1397–1402. IEEE (2011)
40. Martínez-Ballesteros, M., Troncoso, A., Martínez-Álvarez, F., Riquelme, J.C.: Mining quantitative association rules based on evolutionary computation and its application to atmospheric pollution. Integr. Comput. Aid. Eng. **17**(3), 227–242 (2010)
41. Mata, J., Alvarez, J., Riquelme, J.: Mining numeric association rules with genetic algorithms. In: Køurková, V., Neruda, R., Kárný, M., Steele, N.C. (eds.) Artificial Neural Nets and Genetic Algorithms, pp. 264–267. Springer, Vienna (2001). https://doi.org/10.1007/978-3-7091-6230-9_65

42. Mata, J., Alvarez, J.-L., Riquelme, J.-C.: Discovering numeric association rules via evolutionary algorithm. In: Chen, M.-S., Yu, P.S., Liu, B. (eds.) PAKDD 2002. LNCS (LNAI), vol. 2336, pp. 40–51. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-47887-6_5
43. Mlakar, U., Zorman, M., Fister Jr., I., Fister, I.: Modified binary cuckoo search for association rule mining. J. Intell. Fuzzy Syst. **32**(6), 4319–4330 (2017)
44. Moreland, K., Truemper, K.: Discretization of target attributes for subgroup discovery. In: Perner, P. (ed.) MLDM 2009. LNCS (LNAI), vol. 5632, pp. 44–52. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-03070-3_4
45. Arakkal Peious, S., Sharma, R., Kaushik, M., Shah, S.A., Yahia, S.B.: Grand reports: a tool for generalizing association rule mining to numeric target values. In: Song, M., Song, I.-Y., Kotsis, G., Tjoa, A.M., Khalil, I. (eds.) DaWaK 2020. LNCS, vol. 12393, pp. 28–37. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-59065-9_3
46. Poli, R., Kennedy, J., Blackwell, T.: Particle swarm optimization. Swarm Intell. **1**(1), 33–57 (2007)
47. Qodmanan, H.R., Nasiri, M., Minaei-Bidgoli, B.: Multi objective association rule mining with genetic algorithm without specifying minimum support and minimum confidence. Expert Syst. Appl. **38**(1), 288–298 (2011)
48. Rashedi, E., Nezamabadi-Pour, H., Saryazdi, S.: GSA: a gravitational search algorithm. Inf. Sci. **179**(13), 2232–2248 (2009)
49. Rashedi, E., Rashedi, E., Nezamabadi-pour, H.: A comprehensive survey on gravitational search algorithm. Swarm Evol. Comput. **41**, 141–158 (2018)
50. Russell, S., Norvig, P.: Prentice Hall Series in Artificial Intelligence. Prentice Hall, Englewood Cliffs (1995)
51. Salleb-Aouissi, A., Vrain, C., Nortet, C., Kong, X., Rathod, V., Cassard, D.: Quant-Miner for mining quantitative association rules. J. Mach. Learn. Res. **14**(1), 3153–3157 (2013)
52. Seki, H., Nagao, M.: An efficient java implementation of a GA-based miner for relational association rules with numerical attributes. In: 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 2028–2033. IEEE (2017)
53. Sharma, R., Kaushik, M., Peious, S.A., Yahia, S.B., Draheim, D.: Expected vs. unexpected: selecting right measures of interestingness. In: Song, M., Song, I.-Y., Kotsis, G., Tjoa, A.M., Khalil, I. (eds.) DaWaK 2020. LNCS, vol. 12393, pp. 38–47. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-59065-9_4
54. Shih, M.Y., Jheng, J.W., Lai, L.F.: A two-step method for clustering mixed categroical and numeric data. Tamkang J. Sci. Eng. **13**(1), 11–19 (2010)
55. Srikant, R., Agrawal, R.: Mining quantitative association rules in large relational tables. In: Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data, pp. 1–12 (1996)
56. Srinivas, N., Deb, K.: Muiltiobjective optimization using nondominated sorting in genetic algorithms. Evol. Comput. **2**(3), 221–248 (1994)
57. Storn, R., Price, K.: Differential evolution: a simple and efficient adaptive scheme for global optimization over continuous spaces. J. Glob. Optim. **23** (1995)
58. Storn, R., Price, K.: Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces. J. Glob. Optim. **11**(4), 341–359 (1997)
59. Taboada, K., Gonzales, E., Shimada, K., Mabu, S., Hirasawa, K., Hu, J.: Association rule mining for continuous attributes using genetic network programming. IEEJ Trans. Electr. Electron. Eng. **3**(2), 199–211 (2008)

60. Taboada, K., Mabu, S., Gonzales, E., Shimada, K., Hirasawa, K.: Genetic network programming for fuzzy association rule-based classification. In: 2009 IEEE Congress on Evolutionary Computation, pp. 2387–2394. IEEE (2009)
61. Tahyudin, I., Nambo, H.: The combination of evolutionary algorithm method for numerical association rule mining optimization. In: Xu, J., Hajiyev, A., Nickel, S., Gen, M. (eds.) Proceedings of the Tenth International Conference on Management Science and Engineering Management. AISC, vol. 502, pp. 13–23. Springer, Singapore (2017). https://doi.org/10.1007/978-981-10-1837-4_2
62. Tan, S.C.: Improving association rule mining using clustering-based discretization of numerical data. In: 2018 International Conference on Intelligent and Innovative Computing Applications (ICONIC), pp. 1–5. IEEE (2018)
63. Tang, R., Fong, S., Yang, X.S., Deb, S.: Wolf search algorithm with ephemeral memory. In: Seventh International Conference on Digital Information Management (ICDIM 2012), pp. 165–172. IEEE (2012)
64. Telikani, A., Gandomi, A.H., Shahbahrami, A.: A survey of evolutionary computation for association rule mining. Inf. Sci. **524**, 318–352 (2020)
65. Triguero, I., García, S., Herrera, F.: Differential evolution for optimizing the positioning of prototypes in nearest neighbor classification. Pattern Recognit. **44**(4), 901–916 (2011)
66. Yamany, W., Emary, E., Hassanien, A.E.: Wolf search algorithm for attribute reduction in classification. In: 2014 IEEE Symposium on Computational Intelligence and Data Mining (CIDM), pp. 351–358. IEEE (2014)
67. Yan, D., Zhao, X., Lin, R., Bai, D.: PPQAR: parallel PSO for quantitative association rule mining. Peer-to-Peer Netw. Appl. **12**(5), 1433–1444 (2019)
68. Yan, X., Zhang, C., Zhang, S.: Genetic algorithm-based strategy for identifying association rules without specifying actual minimum support. Expert Syst. Appl. **36**(2), 3066–3076 (2009)
69. Yang, J., Feng, Z.: An effective algorithm for mining quantitative associations based on subspace clustering. In: 2010 International Conference on Networking and Digital Society, vol. 1, pp. 175–178. IEEE (2010)

Appendix 2

[II]

M. Kaushik, R. Sharma, S. A. Peious, M. Shahin, S. B. Yahia, and D. Draheim. A systematic assessment of numerical association rule mining methods. *SN Computer Science*, 2(5):1–13, 2021

SN Computer Science (2021) 2:348
https://doi.org/10.1007/s42979-021-00725-2

REVIEW ARTICLE

A Systematic Assessment of Numerical Association Rule Mining Methods

Minakshi Kaushik[1] · Rahul Sharma[1] · Sijo Arakkal Peious[1] · Mahtab Shahin[1] · Sadok Ben Yahia[2] · Dirk Draheim[1]

Received: 18 March 2021 / Accepted: 22 May 2021 / Published online: 22 June 2021

Abstract

In data mining, the classical association rule mining techniques deal with binary attributes; however, real-world data have a variety of attributes (numerical, categorical, Boolean). To deal with the variety of data attributes, the classical association rule mining technique was extended to numerical association rule mining. Initially, the concept of numerical association rule mining started with the discretization method, and later, many other methods, e.g., optimization, distribution are proposed in state-of-the-art. Different authors have presented various algorithms for each numerical association rule mining method; therefore, it is hard to select a suitable algorithm for a numerical association rule mining task. In this article, we present a systematic assessment of various numerical association rule mining methods and we provide a meta-study of thirty numerical association rule mining algorithms. We investigate how far the discretization techniques have been used in the numerical association rule mining methods.

Keywords Knowledge discovery in databases · Data mining · Association rule mining · Numerical association rule mining · Quantitative association rule mining

Introduction

In today's scenario, data is growing explosively, and it is available in many various forms (numerical, text, images, etc.). To manage this humanly unmanageable large amount of data, researchers and data scientists have developed many techniques. In knowledge discovery in databases (KDD), data mining is a popular technique for extracting the required information and finding patterns between data items. Association rule mining(ARM), classification, clustering, regression, etc., are a few well-known data mining techniques. Agrawal [2] introduced ARM in 1993 for finding the relationship between different data items, and later, he proposed the Apriori [3] algorithm and its version to discover interesting rules in large databases. ARM is widely used in market basket analysis, medical diagnosis, and bioinformatics. Apriori and FP-growth [28] are also the most popular algorithms in classical association rule mining. Different authors have various opinions about the discretization process and ARM. Recently, Draheim [18] "provides a frequentist semantics for conditionalization on partially known events, which is given as a straightforward generalization of classical conditional probability via so-called probability testbeds."

This work has been partially conducted in the project "ICT programme" which was supported by the European Union through the European Social Fund.

This article is part of the topical collection "Future Data and Security Engineering 2020" guest edited by Tran Khanh Dang.

✉ Minakshi Kaushik
minakshi.kaushik@taltech.ee

Rahul Sharma
rahul.sharma@taltech.ee

Sijo Arakkal Peious
sijo.arakkal@taltech.ee

Mahtab Shahin
mahtab.shahin@taltech.ee

Sadok Ben Yahia
sadok.ben@taltech.ee

Dirk Draheim
dirk.draheim@taltech.ee

[1] Information Systems Group, Tallinn University of Technology, Akadeemia tee 15a, 12618 Tallinn, Estonia

[2] Software Science Department, Tallinn University of Technology, Akadeemia tee 15a, 12618 Tallinn, Estonia

The classical association rule mining deals only with the binary attributes, whereas real-world data have mixed attributes (numerical, categorical). Therefore, whenever data is in numerical form (height, weight, or age), the data items need to be changed from numerical to discrete using a discretization process. This process of finding association rules in numerical data items has been referred to as numerical association rule mining (NARM) or quantitative association rule mining (QARM) [60]. Initially, NARM was started with the discretization method, and later many methods (optimization, discretization, distribution) are proposed in the literature. Therefore, many other authors investigated the discretization method and proposed various alternatives to the discretization method.

In the literature, various methods with multiple algorithms are discussed; however, selecting an appropriate algorithm for a NARM task with valid reasons is not yet discussed. This article extends our previous work [32] and provide a detailed study of thirty NARM algorithms under different NARM methods. We also investigate how far the discretization techniques have been used in the numerical association rule mining methods.

We conduct an automated search process over Scopus Database and manual search on Google Scholar. We decide to have the term ("Numerical Association Rule Mining" OR "Quantitative Association Rule Mining") to search in abstract, title, and keyword. Our research is limited to the articles published between the years 1996-2020. The selected papers are again assessed on the following criteria:

- Papers introducing novel algorithm in numerical association rule mining or quantitative association rule mining.
- Papers extending the existing algorithm in numerical association rule mining or quantitative association rule mining.

Moreover, we use the following criteria to exclude the papers from the list of searched papers:

- Papers introducing the application of NARM algorithm in any field.
- Papers published in languages other than English.
- Technical reports, thesis and other documents had no peer-review process.

The paper is structured as follows. In section "Preliminaries," we describe preliminaries. In section "Methods to Solve Numerical ARM Problems," we discuss all three methods to solve numerical association rule mining problems. In section "The Optimization Method," the optimization method is discussed with all its sub-methods. In section "The Distribution Method," the distribution method is introduced and discussed, and in section "The Discretization Method," the discretization method is discussed. A discussion on various methods and algorithms is given in section "Discussion." The conclusion is given in section "Conclusion."

Preliminaries

In this section, we provide basic introductions about ARM and NARM.

Association rule mining

In ARM, association rules are based on the If-then relations, which consist of antecedents (If) and consequents (Then) [2]. For example, (1) shows the following association rule: "If a customer buys bread, then he also buys milk." Here, Bread appears as antecedent and Milk as consequent. Generally, an association rule may be represented as a production rule in an expert system, an *if statement* in a programming language, or an implication in a logical calculus.

$$\{\text{Bread}\} \Rightarrow \{\text{Milk}\} \tag{1}$$

In a database, let I be a set of m binary attributes $\{i_1, i_2, i_3, \dots, i_m\}$ called database items. Let T be a set of n transactions $\{t_1, t_2, t_3, \dots, t_n\}$, where each transaction t_i has a unique ID and consists of a subset of the items in I, i.e., $t_i \subseteq I$. As in (1), an association rule is an implication of the form

$$X \Rightarrow Y \tag{2}$$

where $X, Y \subseteq I$ (itemsets) and $X \cap Y = \emptyset$. An association rule can be extracted on the basis of two important measures: support and confidence. Support of an association rule can be defined as the percentage of transactions of the total records containing both sets of items X and Y that are $(X \cup Y)$. Confidence of an association rule can be described as the percentage of transactions that contain X also contain Y.

$$\text{Support}(X \Rightarrow Y) = \text{Supp}(X \cup Y) \tag{3}$$

$$\text{Confidence}(X \Rightarrow Y) = \frac{\text{Supp}(X \cup Y)}{\text{Supp}(X)} \tag{4}$$

For instance, with the reference of Table 1, we can understand the concept of support and confidence. The support of the association rule (Bread ⇒ Milk) is 2/6= 0.33. Since both items are bought together two times out of six transactions, so support is called 20%. However, both items are bought together two times out of four transactions that contain Bread. This indicates the confidence 2/4= 0.5 is 50%.

In ARM, to find out the interesting rules, various interestingness measures are proposed in the literature [58]. In

Table 1 Market basket analysis in association rule mining

TID	Items
T1	Milk, butter
T2	Butter, bread
T3	Sugar, milk
T4	Milk, bread
T5	Sugar, bread
T6	Milk, sugar, bread

classical ARM, frequent itemsets and association rules are discovered from a Boolean dataset; therefore, it is also known as binary or Boolean ARM. Table 2 shows a Boolean dataset for classical ARM. This table contains attributes corresponding to each item and a row corresponding to each transaction. Each attribute has a value “1” if the item is available in the transaction else “0”.

Numerical Association Rule Mining

To extract association rules from numerical data, the problem of the quantitative or categorical attribute was first discussed by Srikant in 1996 [60]. In NARM, whenever data is in numerical form (height, weight, or age), the data items need to be changed from numerical to discrete using a discretization process. This process of finding association rules in numerical data items has been referred to as numerical association rule mining (NARM) [60]. NARM can easily be understood by the following example.

$$\text{Age }[25,40] \wedge \text{Gender : [Female]} \Rightarrow \text{Salary }[1300,2000]$$
$$(\text{Supp} = 30\%, \text{Confidence} = 60\%)$$

Given a set of transactions T, let Antecedent denote the set of transactions in T in which Age has a value between 25 and 40 and Gender is Female. Similarly, let Consequent denote the set of transactions in which Salary has a value between $1300 and $2000. For instance, with reference to Table 3, here $\text{Supp} = 30\%$ denotes that 30% of the employees are females and between the ages 25 and 40, earning a salary of between $1300 and $2000. $Conf = 60\%$ denotes that 60% of the female employees between age 25 and 40 are earning a salary of between $1300 and $2000. Here Age and Salary are numerical attributes and Gender is a categorical attribute.

Table 2 Example of Boolean dataset

TID	Milk	Butter	Bread	Sugar
T1	1	1	0	0
T2	0	1	1	0
T3	1	0	0	1
T4	1	0	1	0
T5	0	0	1	1
T6	1	0	1	1

Table 3 Example of numerical values dataset

Age	Gender	Salary
25	Male	1200
26	Female	1250
28	Female	1250
30	Female	1350
35	Female	1600
38	Female	1700
40	Male	1900
42	Male	1950
48	Female	2500
50	Male	3000

As an early solution, the problem of association rules for numerical data was solved using a discretization process where numeric attributes are divided into different intervals and, henceforth, these attributes are treated as categorical attributes [12]. For example, an attribute Age with values between 20 and 80 can be divided into six different age intervals $(20-30, 30-40, 40-50, 50-60, 60-70, 70-80)$. The data discretization process is an obvious solution; however, it reveals a loss of valuable information, which might cause poor results [17]. Thus, we review solutions from three different approaches (discretization, distribution and optimization) to solve issues with numerical association rule mining in section “Methods to Solve Numerical ARM Problems.”

Methods to Solve Numerical ARM Problems

To solve the issues in NARM, three main approaches (discretization, distribution and optimization) have been discussed in the literature. Based on these three approaches, many different NARM algorithms are proposed. The optimization method has several sub-methods as swarm intelligence and evolution-based algorithms, covering most of the area to deal with NARM. The distribution method does not contribute much in this area; however, the discretization method is a common method that transforms continuous attributes into discrete attributes and it is further subdivided into three sub-methods. Figure 1 (also compared with Fig. 1 in [9]) shows all three approaches and different algorithms proposed under each approach.

The Optimization Method

To solve the NARM problems, many researchers have moved towards optimization methods. Optimization methods provide a robust and efficient approach to explore a massive

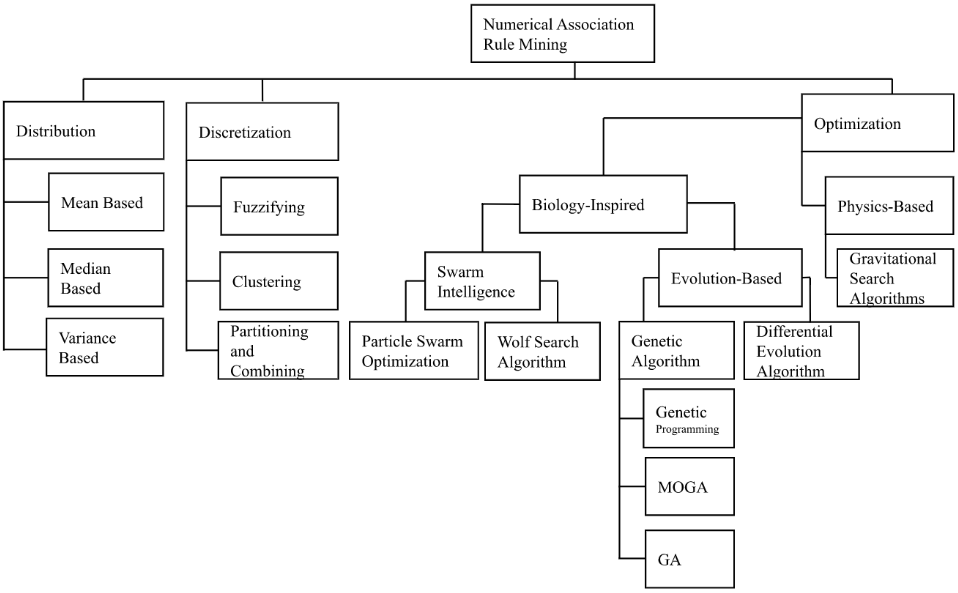


Fig. 1 Different methods and algorithms to solve numerical association rule mining problems

search space. In this method, researchers have invented a collection of heuristic optimization methods inspired by the movements of animals and insects. For finding association rules, optimization methods work in two phases. In the first phase, all the frequent itemsets are found and in the second phase, all relevant association rules are extracted. As shown in Fig. 1, optimization methods are divided into bio-inspired optimization and physics-based optimization methods. Table 4 shows an overview of all those algorithms that come under the optimization method.

The Bio-inspired Optimization Method

Biology-based algorithms are generally divided into two parts: swarm-intelligence-based algorithms and evolution-based algorithms [15]. The main origin of these algorithms is the biological behavior of natural objects [68].

Evolution-Based Algorithms

Evolution-based algorithms are inspired by Darwinian principles and were first applied in [48]. These algorithms mimic the capability of nature to develop living beings that are well-adapted to their environment [68]. Evolution-based algorithms exploit stochastic search methods that follow the idea of natural selection and genetics. The algorithms show strong adaptability and self-organization [15] and use biology-inspired operators such as crossover, mutation, and natural selection [68]. The *Genetic Algorithm* [30] and the *Differential Evolution Algorithm* [63]

Table 4 An overview of optimization method algorithms for NARM

Methods	Basic technique	Algorithms
GA	Genetic Algorithm	GENAR [47], GAR [48], EGAR [39], ARMGA [74], EARMGA [74], GAR-PLUS [10], QuantMiner [56], RelQM-J [57], RCGA [46]
MOGA	Genetic Algorithm	ARMMGA [53], QAR-CIP-NSGA-II [45]
DE	Differential Evolution	MODENAR [7], ARM-DE [20]
PSO	Particle Swarm Optimization	RPSO [4], CENPSO [5], MOPAR [12], PPQAR [73], PARCD [65]
WSA	Swarm Intelligence	WSA [1]
GSA	Physics-based (Gravity)	GSA [13]

are two examples of evolution-based algorithms. Table 5 shows an overview of the evolution-based algorithms for NARM, together with concepts.

Genetic Algorithms (GA) GA was first proposed by Holland [30] and they are one of the most popular algorithms in bio-inspired optimization methods. A basic genetic algorithm consists of five phases: initialization, evaluation, reproduction, crossover, and mutation. GAs for NARM can be divided into three fields, i.e., basic genetic algorithms, genetic programming and multiobjective genetic algorithms. A basic genetic algorithm has been proposed by Mata et al. [47] and together with the tool GENAR (GENetic Association Rules) to discover association rules with numeric attributes. With this tool, an undetermined amount of numeric attributes in antecedent and unique numeric attribute in consequent can be obtained. Association rules in GENAR algorithms allow for intervals (maximum and minimum values) for each numeric attribute. Mata et al. [48] further extended the GENAR algorithms and proposed a technique named GAR (Genetic Association Rule) to discover association rules in numeric databases without discretization. Authors present a technique to find frequent itemsets in numeric databases without needing to discretize numeric attributes. This algorithm was useful only for finding the frequent itemsets, not for association rules. In this paper, a genetic algorithm was used to find the suitable amplitude of the intervals that conform *k*-itemset and can have a high support value without too wide intervals. In [39], the GAR algorithm was further extended to EGAR (extended genetic association rule). This algorithm generates frequent patterns with continuous data [48].

A genetic-based strategy and two other algorithms ARMGA and EARMGA, were proposed by Yan et al. [74]. In this approach, an encoding method was developed with relative confidence as the fitness function. ARMGA was proposed for Boolean ARM and EARMGA for quantitative attributes or generalized association rules. In these algorithms, there was no requirement of a minimum support threshold. The GAR-plus tool was presented by Alvarez [10]. This tool deals with categorical and numeric attributes in large databases without any need for a prior discretization of numeric attributes.

In 2013, Salleb et al. [56] proposed “Qu antMiner, a quantitative association rule mining system based on the genetic algorithm. This tool dynamically discovers meaningful intervals in association rules by optimizing both the confidence and the support values.

Seki and Nagao [57] worked on GA-based QuantMiner for multi-relational data mining and developed RelQM-J, a tool for relational quantitative association rules in Java programming language. In this tool, efficient computation of the support of the rules has been realized by using a hash-based data structure.

A real-coded [36] genetic algorithm was presented in [46] in 2010. The proposed algorithm RCGA follows the CHC binary-coded evolutionary algorithm [19]. RCGA algorithm has been applied to pollutant agent time series and helps to find all existing relations between atmospheric pollution and climatological conditions.

Table 5 An overview of evolution based algorithms for NARM

Algorithm	Proposer	Concept
GENAR [47]	J. Mata Vázquez et al. (2001)	Based on finding frequent itemsets in numerical databases and intervals of all attributes that conform to those frequent itemsets
GAR [48]	J. Mata Vázquez et al. (2002)	Extended version of GENAR
EGAR [39]	H. KwaŚnicka et al. (2006)	Uses medical databases where attributes are continuous and discrete; extended version of GAR.
EARMGA [74]	A. Yan et al. (2009)	No requirement of minimum support threshold.
GARPLUS [10]	V. Álvarez et al. (2012)	Based on the finding intervals of numeric attribute.
QUANTMINER [56]	A. Salleb Aouissi et al. (2013)	Based on genetic algorithm to find good intervals by optimizing both support and confidence.
RelQM-J [57]	H. Seki1(2017)	Based on mining numeric rules from relational databases, implemented in Java
RCGA [46]	M. Martinez Ballesterosa (2010)	Based on CHC binary-coded evolutionary algorithm.
ARMMGA [53]	H. Reza Qodmanan (2011)	Based on multi-objective genetic algorithm.
QAR-CIP-NSGA-II [45]	D. Martın et al. (2011)	Based on NSGA with three measures (comprehensibility, interestingness, performance).
MODENAR [7]	B. Alatas (2008)	Based on multi-objective differential evolutionary algorithm.
ARM-DE [20]	I. Fister Jr. (2018)	Single objective optimization problem where features consist of numerical as well as categorical attributes.

Genetic Programming for ARM Genetic Programming [37] is a well-known type of GA. In GA, the genome is in string structure, while in GP, the genome is in the form of tree structure [29]. Genetic Network Programming (GNP) is a graph-based evolutionary algorithm and finds the association rules for continuous attributes. In this method, important rules are stored in a pool and these extracted rules are measured by the chi-squared test. This pool is updated in every generation by exchanging the association rule with a higher chi-squared value for the same association rule with a lower chi-squared value [64].

Multi-Objective Genetic Algorithm The multi-objective genetic algorithm was proposed by Fonseca et al. [21] in 1993. Generally, the resource consumption of an association rule mining computation is affected by two parameters, i.e., minimum support and minimum confidence. In classical ARM algorithms, only a single measure (support or confidence) has been used as a measure to evaluate the rule interestingness, therefore, if the values of minimum support and minimum confidence are not appropriately set, then the number of association rules may be significantly less, or it may be very large. This problem can be solved by using more objectives or measures as referred to in multi-objective ARM.

Gosh and Nath [23] used a Pareto-based genetic algorithm to solve the multi-objective rule mining problem using three measures: interestingness, comprehensibility and predictive accuracy. The single-objective algorithm, ARMGA [74], had issues that were addressed by introducing the multi-objective genetic algorithm called ARMMGA by Qodmanan et al. in [53]. The ARMGA algorithm finds high confidence and low support rules, whereas ARMMGA finds high confidence and high support rules. ARMGA has a large set of rules compared to ARMMGA; this problem was solved using a new fitness function in ARMMGA. To prevent invalid chromosomes in ARMGA, new crossover and mutation operators are presented in the literature.

Srinivasan and Deb [61] proposed a non-dominated genetic sorting algorithm to solve multi-objective optimization problems. In 2002, Deb et al. [16] extended NSGA to NSGA-II. In 2011, Martin et al. [45] extended NSGA-II with a trade-off between interpretability and accuracy. NSGA-II performs evolutionary learning of intervals of attributes. For each rule, condition selection is made for three objectives (interestingness, comprehensibility and performance). This method did not depend on minimum support and confidence thresholds. Martin et al. again extended their research on NSGA-II to a new approach called QAR-CIP-NSGA-II and compared the results of this algorithm with other MOEA(Multi-objective evolutionary algorithm) algorithms.

Differential Evolutionary Algorithms Differential evolutionary (DE) algorithms are evolution-based algorithms. These algorithms were proposed by Storn and Price in [62]. DE algorithms are simple and effective single-objective optimization algorithms that solve real-valued problems based on the principle of natural evolution. DE algorithms use Genetic-based operators such as crossover, mutation, and selection. Although the evolution process of DE is similar to the one of GA, it relies on a mutation operator instead of a crossover operator [69].

A Pareto-based multi-objective DE algorithm for ARM was first proposed in [7] by Alatas et al. for searching accurate and comprehensible association rules. The problem of mining association rules was formulated with four objective optimization problems, i.e., support, confidence, comprehensibility and amplitude. Here, support, confidence and comprehensibility are maximization objectives and the amplitude of intervals is a minimization objective. In a single run, a Pareto-based multi-objective DE algorithm search intervals of numeric attributes and association rules.

In 2018, [20] proposed a novel approach for mining association rules with numerical and categorical attributes based on DE. In this algorithm, a single objective optimization problem is considered in which support and confidence of association rules are combined into a fitness function. This new DE using ARM (ARM-DE) with mixed (i.e., numerical and categorical) attributes consists of three stages: (1) domain analysis, (2) representation of a solution, (3) definition of a fitness function.

Swarm Intelligence Based Algorithms

Swarm intelligence-based algorithms are further divided into two sub-optimization methods, particle swarm optimization and the wolf search algorithm. Table 6 provides an overview of swarm intelligence algorithms for NARM.

Particle Swarm Optimization Particle Swarm Optimization (PSO) is a population-based optimization algorithm for nonlinear functions. This algorithm is oriented towards animal behavior, such as bird flocking or fish schooling. It was developed in 1995 [33, 52]. PSO was first used for NARM to find intervals of the numerical attributes in 2008 [4].

Rough PSOA, based on rough patterns, was proposed in [4], in which rough values are defined with upper and lower intervals. This algorithm can complement the existing tools developed in rough computing. Rough values are helpful in representing an interval for an attribute. In this work, each particle consists of a decision variable that has three parts. The first part of each decision variable represents the antecedent or consequent of the rule and can take values between 0 and 1. The second part describes the lower bound; the third part represents the upper bound of the item interval.

Table 6 An overview of Swarm intelligence based algorithms for NARM.

Algorithm	Proposer	Concept
RPSO [4]	B. Alatas et al. (2008)	RPSOA is based on the notion of rough patterns that use rough values defined with upper and lower intervals.
CENPSO [5]	B. Alatas et al. (2009)	CENPSO is based on chaos numbers.
MOPAR [12]	V. Beiranvand et al. (2014)	MOPAR is Based on Multi objectives (confidence, comprehensibility and interestingness).
MOPSO [38]	Kuo et al. (2019)	Based on pareto optimal strategy using adaptive archive grid for multi-objective PSO.
Parallel PSO [73]	A. Yan et al. (2019)	Parallel PSO is based on two methods of parallel algorithm: particle-oriented and data-oriented parallelization.
PARCD [65]	I. Tahyudin et al. (2017)	Combined PSO method with cauchy distribution.
WSA [1]	I.E. Agbehadji et al. (2016)	Based on wolves hunting strategy.

The second and third parts are combined as one rough value during the implementation phase of particle representation.

Alatas and Akin [5] proposed a novel PSO algorithm based on chaos numbers. The CENPSOA algorithm (chaotically encoded PSO) uses chaos decision variables and chaos particles. Chaos and PSO relation were first discovered by Liu et al. [42]; the CENPSOA algorithm performs encoding of particles given by chaos numbers. The Chaos numbers consist of the midpoint and radius part of values [5]. Alatas and Akin [6] also proposed a multi-objective chaotic particle swarm optimization algorithm for mining accurate and comprehensible classification rules.

Yan et al. [73] proposed a parallel PSO algorithm for NARM. This parallel algorithm was designed with two strategies called particle-oriented and data-oriented parallelization. Particle-oriented parallelization is more efficient and data-oriented parallelization is more scalable to process large datasets.

To discover association rules in a single step without prior discretization of numerical attributes, Beiranvand et al. [12] proposed a multi-objective particle swarm optimization algorithm (MOPAR). The algorithm defines multiple objectives such as confidence, comprehensibility and interestingness. In the Pareto method, a candidate solution is identified better than all other candidates. In multi-objective optimization, a set of best solutions is identified in which the members are superior among all the candidates.

Kuo et al. [38] proposed a multi-objective particle swarm optimization algorithm using an adaptive archive grid for NARM. It is also based on Pareto's optimal strategy. In this algorithm, minimum support and minimum confidence are not required before mining. MOPSO algorithm is executed in three parts: (1) initialization, (2) adaptive archive grid, and (3) particle swarm optimization searching.

PSO for NARM with Cauchy distribution (PARCD) has been evaluated by [65] and it showed that the result of PARCD is better than the method of MOPAR.

Wolf Search Algorithm The wolf search algorithm (WSA) is a bio-inspired heuristic optimization algorithm. It was proposed by [67] and imitated the way wolves search for food and survive by avoiding their enemies. WSA is tested and compared with other heuristic algorithms and investigated with respect to its memory requirements. The group of wolves has characteristics of commuting together as a nuclear family; that is why it is different from particle swarm optimization [72].

Agbehadji and Fong [1] proposed a new meta-heuristic algorithm that used the wolf search algorithm for NARM. The wolf has three different features of preying. These are prey initiatively, prey passively and escape. The preying initiatively feature allows the wolf to check its visual perimeter to detect prey. If the prey is found within visual distance, the wolf moves towards the prey with the highest fitness value; else, the wolves will maintain their direction. In prey passively mode, the wolf only stays alert from threats and tries to improve its position. In the escape mode, when a threat is detected, the wolf escapes quickly by relocating itself to a new position with an escape distance greater than its visual range.

Physics-Based Algorithm

The physics-based meta-heuristic optimization algorithm simulates the physical behavior and properties of the matter or follows the laws of physics [15]. For NARM, the gravitational search algorithm is a physics-based meta-heuristic optimization algorithm.

Gravitational Search Algorithm

Rashedi et al. proposed a new optimization algorithm based on the law of gravity and named it gravitational search algorithm (GSA) [54]. Newtonian gravity laws state that "Every particle in the universe attracts every other particle with a force that is directly proportional to the product of their masses and inversely proportional to the square of the distance between them." In GSA, agents act as objects and their performance is evaluated by their mass. Each mass presents a solution and it is expected that masses will be attracted by the heaviest mass. GSA is like a small artificial world of masses obeying the Newtonian laws of gravitation and motion. There are four ways of representing the agents or coding the problem variables. These are continuous (real-valued), binary-valued, discrete, and mixed, which are called GSA variants [55].

Can and Alatas [13] first used GSA for NARM. GSA eliminated the task of finding the minimum values of support and confidence. Automatically mined rules have high confidence and support values. In this work, GSA has been designed to automatically find the numerical intervals of the attributes, i.e., without any *a priori* data process at the time of rule mining. The problem of interactions within attributes has been eliminated with the designed GSA by not selecting one attribute at a time and not evaluating a partially-constructed candidate rule due to its global searching with a population.

The Distribution Method

In [11], Aumann and Lindell have introduced a new definition for numerical association rules based on statistical inference theory. In this study, they have implemented several distribution scales, including mean, median, and variance. The following example shows the kind of generalization of ARM proposed by the authors.

$$\text{Gender} = F \Rightarrow \text{Wage} : \text{mean} = \$8.50 \quad (\text{overall mean wage} = \$12.60) \tag{5}$$

As the above example shows, the average wage for females was $ 8.50 p/hr. The rule displays that the wage of that group was far less than the average wage; therefore, this rule can be considered useful. They also used the algorithm to identify repeated itemsets and then calculate the desired statistics for the purpose with respect to repeated itemsets. This procedure is restricted by the requirement to store every repeated itemsets in memory throughout repeated itemset generation. Where the data is not sparse, the number of frequent itemsets will be huge and repeated itemset storage and access will dominate the calculation. Moreover, they concluded that the suggested algorithm is beneficial and may find rules between two given quantitative attributes. Webb [71] extended the work proposed by Aumann and Lindell in [11] with name impact rules using the OPUS search algorithm [70]. In this paper, the author evaluated the impact of conditions on a numeric variable that association rules with discretization can not emulate. The author compared the frequent itemset approach with the OPUS_IR approach. The author found OPUS_IR avoids large memory requirements with a frequent itemset approach by avoiding the need to store all frequent itemsets.

The Discretization Method

Discretization is a process of quantizing numerical attributes into groups of intervals and it is one of the most popular methods to solve the problem of numerical association rule mining. There are numerous methods of discretization in literature. Due to different needs, discretization methods have been developed in different ways, such as supervised vs. unsupervised, dynamic vs. static, global vs. local, splitting (top-down) vs. merging (bottom-up) and direct vs. incremental [43]. In classical ARM algorithms, numerical columns cannot be processed directly [44], i.e., all columns need to be categorical, which is a major limitation of ARM [66].

Discretization of numerical values is used to overcome this problem [34, 49, 50]. When a numeric column is divided into useful target groups, it becomes easier to identify and generate association rules, i.e., discretization helps to understand the numeric columns better. The discretized groups are useful only if the variables in the same group do not have any objective difference. Discretization minimizes the impact of trivial variations between values. Discretization can be performed using fuzzifying, clustering and partitioning and combining [8]. In Table 7, we summarize some selected discretization algorithms used in NARM.

Fuzzifying

Fuzzifying is the technique of illustrating numeric values as fuzzy sets [35], which can help to rectify the *sharp boundary problem* of ARM. Sometimes, endpoint values of discretized groups have more or less influence on the result than the midpoint values: this phenomenon is known as a sharp boundary problem. Fuzzy Class Association Rule Support Vector Machine (FCARSVM) is a model proposed by Kianmehr et al. [35] to get the fuzzy class association rules. In the first phase of the model, Fuzzy class association rules (FCAR) are extracted using fuzzy c-means clustering algorithm for quantitative datasets and in the second phase, extracted FCARs are weighted based on scoring metric strategy.

For mining fuzzy quantitative association rules, those have crisp values, fuzzy terms and intervals in both antecedent and consequent, Zhang [76] presented an algorithm EDPFT(equal-depth partition with the fuzzy term). The author used an equal-depth partition algorithm for finding the intervals of numeric values and map crisp values and fuzzy terms of each categorical attribute into consecutive integers and generate frequent itemsets using

Table 7 An overview of discretization-based algorithms for NARM

Algorithm	Proposer	Concept
ARCS [40]	B. Lent et al. (1997)	Based on segmenting clusters using the geometric-based BitOp algorithm.
DRMiner [41]	W .Lian (2005)	Based on finding density regions in a multidimensional space.
DBSMiner [25]	G. Yunkai et al. (2008)	Based on clustering of high density sub-spaces using a density- and grid-based cluster algorithm.
MQAR [75]	Y. Junrui et al. (2010)	Based on finding dense sub-spaces using structure DGFP-tree.
Srikant's Partitioning and combining algorithm [60]	R. Srikant(1996)	Partitioning the numeric attribute into interval using equi-depth method
APACS2 [14]	K.C.C. Chan et al. (1997)	Based on partitioning approach
EDPFT [76]	W. Zhang (1999)	Based on fuzzifying approach with equal-depth partition method
FTDA [31]	Hong et al. (1999)	Based on fuzzifying approach with apriori algorithm
OFARM [77]	H. Zheng et al. (2014)	Based on fuzzifying approach with multi-objective functions

the extended apriori algorithm. In 1999 Hong et al. [31] also proposed an algorithm FTDA (fuzzy transaction data-mining algorithm), which integrates the fuzzy-set concepts with an apriori algorithm. This method encounters the problem of requiring the fuzzy-sets and their corresponding membership functions in advance. Choosing the best fuzzy-sets for mining the association rule is difficult, as anomalies may occur if fuzzy-sets are not well chosen. To tackle this problem, [26] introduced an additional fuzzy normalization process and proposed an algorithm for fuzzy quantitative association rules. [26] also compared with normalization and without normalization methods for mining fuzzy quantitative rules and show with normalization method gives a high number of interesting rules compare to with normalization method. The authors used three interest measures: fuzzy support, fuzzy confidence, and fuzzy correlation. In 2014, [77] proposed a novel algorithm OFARM (optimized fuzzy association rule mining) to optimize the partition points of fuzzy sets with multiple objective functions. A two-level iteration process is used to generate the frequent itemsets and employ certainty factor with confidence to evaluate fuzzy association rules.

Clustering

Clustering is one of the popular methods of discretizing a numerical column in an unsupervised manner [8]. In clustering, a numerical column is segregated into different groups according to the properties of each value; in this method, the probability of having values in the same group depends on the degree of similarity or dissimilarity of the values [27, 59]. To obtain maximum results in clustering, the degree of similarity and dissimilarity needs to be well defined [24]: "In other words, the intra-cluster variance is to be minimized, and the inter-cluster variance is to be maximized" [66]. Two-step clustering [59] is the most common clustering method.

DRMiner Algorithm

Lian et al. [41] have proposed the DRMiner algorithm, which exploits the notion of "density" to capture the characteristics of numeric attributes and an efficient procedure to locate the "dense regions." DRMiner scales up well with high-dimensional datasets. When mapping a database to a multi-dimensional space, the data points (transactions) are not distributed evenly throughout the multi-dimensional space. For this kind of distribution, the density measure was introduced and the problem of mining quantitative association rules transformed into the problem of finding dense regions to map them to find quantitative association rules. Weaknesses of this method were the prior requirement of many thresholds and unsolving the dimensionality curse. It was noted that the algorithm might not perform well for datasets with uniform density between minimum density threshold and low density.

DBSMiner

DBSMiner is a density-based sub-space mining algorithm using the notion of density-connected to cluster the high-density sub-space of numeric attributes and gravitation between grid/cluster to deal with the low-density cells [25]. DBSMiner employs an efficient high dimension clustering algorithm CBSD (Clustering Based on Sorted Dense unit) to deal with high dimensional data sets. The algorithm has a unique feature to deal with low-density sub-spaces and there is no need to scan the whole space; check the neighbor cell. It can find interesting association rules.

MQAR

MQAR (Mining Quantitative Association Rules based on a dense grid) is a novel algorithm that was proposed by Yang and Zhang [75]. The main objective of this algorithm was to mine the numeric association rules using a tree structure, DGFP-tree, to cluster dense space. This algorithm is helpful to eliminate noise and redundant rules by transforming the problem into finding regions with enough density and to map them to quantitative association rules. A novel subspace clustering algorithm was also proposed based on searching DGFP-tree and inserting the dense cell in the database space into DGFP-tree as a path from a root node to a leaf node. MQAR has the advantage that DGFP-tree compresses the database and there is no need to scan the database several times.

ARCS

The Association Rule Clustering System [40] was presented by Lent et al. together with a new geometric-based clustering algorithm, BitOP. In this paper, the problem of clustering of association rules like $(A \wedge B) => C$ where L.H.S. is having quantitative attributes and R.H.S. having a categorical attribute was discussed and a two-dimensional grid is formed where each axis represents one of the L.H.S. attributes. ARCS is an automated system to compute a clustering of two-attribute spaces in large databases. In ARCS framework Binner, For a given partitioning of the input attributes, the algorithm makes only one pass through the data. and allows the support or confidence thresholds to change without requiring a new pass through the data. BitOp algorithm enumerates the clusters. To locate clusters within bitmap grids, the algorithm performs bit-wise operations.

Partitioning and Combining

In [60], Srikant and Agrawal discussed the problems of numeric attributes in databases. The authors addressed the issue of mining association rules from large databases containing both numerical and categorical attributes. A partitioning method was introduced to deal with this problem, which partitions quantitative attributes into intervals and map pairs (attribute, interval) to Boolean attributes. Before partitioning, a measure of partial completeness was introduced to quantify information lost due to partitioning and to decide the number of partitions and whether or not to partition a quantitative attribute. The following formula computes the number of required partitions.

$$\text{Number of intervals} = \frac{2n}{m(K-1)} \tag{6}$$

where n is the number of numeric attributes, m is the minimum support and K is the partial completeness level. To identify interesting rules and to prevent the generation of similar rules, the authors used the “greater-then-expected-value” interest measure.

In [14], a novel algorithm, APACS2 was proposed, which implemented *adjusted difference analysis* to find the interesting associations among attributes. This algorithm has the advantage of discovering both positive and negative associations and it avoids user-specified threshold, which is hard to

Table 8 Summary of different numerical association rule mining methods

Methods	Algorithms	Advantages	Disadvantage	Discretization
Optimization method	Evolution based algorithms [7, 10, 20, 39, 45–48, 53, 56, 57, 74]	Strong adaptability and self organization, No need of discretization,	Comparatively higher computational cost	No
	Swarm based algorithms [1, 4, 5, 12, 38, 65, 73]	Efficient, fast convergence rate	Low local search ability	No
	Physics based algorithms [13]	Automatically find the interval of numeric attributes, easy execution	Not very efficient in searching	No
Discretization Method	Clustering [25, 41]	Ability to scale up for high-dimensional cases	Require many user specified threshold, dimension curse	Yes
	Partitioning [14, 22, 60]	Use basic equi-depth and equi-width techniques to the partition which are easy to implement	Challenge to find the best interval, loss of information	Yes
	Fuzzifying [31, 76, 77]	Non-sharp boundaries for interval	Need to choose perfect fuzzy sets	Yes
Distribution Method	Distribution Algorithm [11]	Generate sub rules	No solution for Multiple comparison procedure	Yes

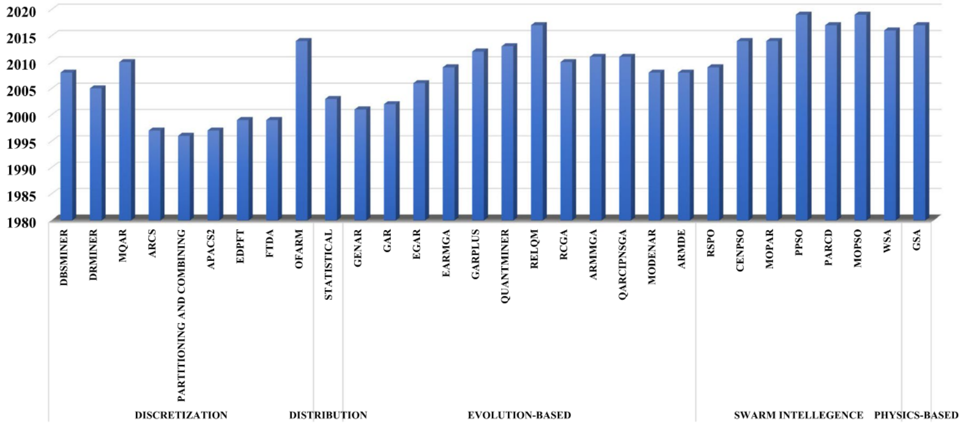


Fig. 2 Year-wise contribution of existing algorithms of NARM

determine. Fukuda et al. [22] presented a novel algorithm to generate optimized intervals in linear time for sorted data. They used *randomized bucketing* as a prepossessing method because it was expensive to sort the quantitative attribute for large databases.

Discussion

In Table 8, we discuss the advantages and disadvantages of the optimization method, discretization, and distribution method. We assessed that every mining method for numerical association rules has some pros and cons. However, being fundamentally different, these approaches have standard support and confidence and mostly have a user-specified threshold. We have investigated that which methods use the discretization technique as a pre-processing step for partitioning or finding the interval of numeric attributes. We observed that all the sub-methods of optimization methods do not use the discretization technique but used in the distribution method. Figure 2 is depicting the year-wise contribution of each method in NARM. It is clear that most of the algorithms of the discretization method were proposed in the 20th century, and few of them were proposed in the 21st century. OFARM is the most recent algorithm that was proposed in 2014. In the swarm intelligence method, parallel PSO and MOPSO are the most recent algorithm among algorithms under all other methods. Algorithms from evolution-based methods came into the scene after 2000. The distribution method was proposed in 2003 and it does not contribute much to NARM. Recently, a Grand report tool has also been proposed that reports mean values of a chosen numeric target column concerning all possible combinations of influencing factors [51].

Conclusion

Real-world databases contain a high volume of quantitative/numerical and categorical data. Therefore, it is essential to use NARM methods for discovering knowledge from these data sets. In this article, we conducted a detailed study on three NARM methods and their supporting algorithms. We have investigated the use of the discretization technique for partitioning the numerical attributes in various NARM methods. We find that the optimization methods (evolution-based algorithms, swarm-intelligence-based algorithms and physics-based algorithms) do not use discretization techniques; however, they have higher computational costs. The distribution method has not been discussed much in the literature and it does not support the multiple comparison procedure. In the discretization method, dimensionality curse and requirement of many user-specified thresholds is also a disadvantage. Finding the best partition is still very challenging and it has a vast scope in NARM. This article highlighted open research challenges, pros and cons of popular NARM methods and algorithms. We concluded that no single NARM method seems to be perfect for discovering patterns from real-world datasets.

Declarations

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

1. Agbehadji IE, Fong S, Millham R. Wolf Search Algorithm for numeric association rule mining. In: IEEE International Conference on Cloud Computing and Big Data Analysis (ICCCBDA), IEEE; 2016. pp. 146–51.
2. Agrawal R, Imieliński T, Swami A. Mining association rules between sets of items in large databases. ACM SIGMOD Record. 1993;22(2):207–16. https://doi.org/10.1145/170036.170072.
3. Agrawal R, Srikant R. Fast Algorithms for mining association rules in large databases. In: Proceedings of 20th International Conference on Very Large Data Bases, Morgan Kaufmann;1994 p. 487–99.
4. Alatas B, Akin E. Rough particle swarm optimization and its applications in data mining. Soft Comput. 2008;12(12):1205–18.
5. Alatas B, Akin E. Chaotically encoded particle swarm optimization algorithm and its applications. Chaos Solit Fract. 2009;41(2):939–50.
6. Alatas B, Akin E. Multi-objective rule mining using a chaotic particle swarm optimization algorithm. Knowl Based Syst. 2009;22(6):455–60.
7. Alatas B, Akin E, Karci A. Modenar: multi-objective differential evolution algorithm for mining numeric association rules. Appl Soft Comput. 2008;8(1):646–56.
8. Altay EV, Alatas B. Performance analysis of multi-objective artificial intelligence optimization algorithms in numerical association rule mining. J Ambient Intell Human Comput. 2019;2019:1–21.
9. Altay EV, Alatas B. Intelligent optimization algorithms for the problem of mining numerical association rules. Phys A. 2020;540:123142.
10. Álvarez VP, Vázquez JM. An evolutionary algorithm to discover quantitative association rules from huge databases without the need for an a priori discretization. Expert Syst Appl. 2012;39(1):585–93.
11. Aumann Y, Lindell Y. A statistical theory for quantitative association rules. J Intell Inf Syst. 2003;20(3):255–83.
12. Beiranvand V, Mobasher-Kashani M, Bakar AA. Multi-objective pso algorithm for mining numerical association rules without a priori discretization. Expert Syst Appl. 2014;41(9):4259–73.
13. Can U, Alatas B. Automatic mining of quantitative association rules with gravitational search algorithm. Int J Softw Eng Knowl Eng. 2017;27(03):343–72.
14. Chan KC, Au WH. An effective algorithm for mining interesting quantitative association rules. In: Proceedings of the 1997 ACM symposium on Applied computing; 1997. pp. 88–90.
15. Cui Y, Geng Z, Zhu Q, Han Y. Multi-objective optimization methods and application in energy saving. Energy. 2017;125:681–704.
16. Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multiobjective genetic algorithm: Nsga-ii. IEEE Trans Evol Comput. 2002;6(2):182–97.
17. Djenouri Y, Bendjoudi A, Djenouri D, Comuzzi, M. Gpu-based bio-inspired model for solving association rules mining problem. In: 2017 25th euromicro international conference on parallel, distributed and network-based processing (PDP), IEEE; 2017. pp. 262–9.
18. Draheim D. Generalized Jeffrey conditionalization: a frequentist semantics of partial conditionalization. Berlin: Springer; 2017.
19. Eshelman LJ. The chc adaptive search algorithm: How to have safe search when engaging in nontraditional genetic recombination. In: Foundations of genetic algorithms, Elsevier; 1991. vol. 1. pp. 265–83.
20. Fister I, Iglesias A, Galvez A, Del Ser J, Osaba E. Differential evolution for association rule mining using categorical and numerical attributes. In: International conference on intelligent data engineering and automated learning, Springer; 2018. pp. 79–88.
21. Fonseca CM, Fleming PJ et al. Genetic algorithms for multiobjective optimization: Formulation discussion and generalization. In: Icga, Citeseer; 1993. vol. 93, pp. 416–23.
22. Fukuda T, Morimoto Y, Morishita S, Tokuyama T. Mining optimized association rules for numeric attributes. J Comput Syst Sci. 1999;58(1):1–12.
23. Ghosh A, Nath B. Multi-objective rule mining using genetic algorithms. Inf Sci. 2004;163(1–3):123–33.
24. Grabmeier J, Rudolph A. Techniques of cluster algorithms in data mining. Data Min Knowl Disc. 2002;6(4):303–60.
25. Guo Y, Yang J, Huang Y. An effective algorithm for mining quantitative association rules based on high dimension cluster. In: 2008 4th international conference on wireless communications, networking and mobile computing, IEEE; 2008. pp. 1–4.
26. Gyenesei A. A fuzzy approach for mining quantitative association rules. Acta Cybern. 2001;15(2):305–20.
27. Han J, Pei J, Kamber M. Data mining: concepts and techniques. Hoboken: Elsevier; 2011.
28. Han J, Pei J, Yin Y, Mao R. Mining frequent patterns without candidate generation: a frequent-pattern tree approach. Data Min Knowl Disc. 2004;8(1):53–87.
29. Hirasawa K, Okubo M, Katagiri H, Hu J, Murata J. Comparison between genetic network programming (gnp) and genetic programming (gp). In: Proceedings of the 2001 congress on evolutionary computation (IEEE Cat. No. 01TH8546), IEEE; 2001. vol. 2, pp. 1276–82.
30. Holland JH. Adaption in natural and artificial systems. In: An introductory analysis with application to biology, control and artificial intelligence; 1975.
31. Hong TP, Kuo CS, Chi SC. Mining association rules from quantitative data. Intell Data Anal. 1999;3(5):363–76.
32. Kaushik M, Sharma R, Peious SA, Shahin M, Yahia SB, Draheim D. On the potential of numerical association rule mining. In: International conference on future data and security engineering, Springer; 2020. pp. 3–20.
33. Kennedy J, Eberhart R. Particle swarm optimization. In: Proceedings of ICNN'95-international conference on neural networks, IEEE; 1995. vol. 4, pp. 1942–48.
34. Khade R, Patel N, Lin J. Supervised dynamic and adaptive discretization for rule mining. In: 2015 In SDM Workshop on Big Data and Stream Analytics; 2015.
35. Kianmehr K, Alshalalfa M, Alhajj R. Fuzzy clustering-based discretization for gene expression classification. Knowl Inf Syst. 2010;24(3):441–65.
36. Kim H, Adeli H. Discrete cost optimization of composite floors using a floating-point genetic algorithm. Eng Optim. 2001;33(4):485–501.
37. Koza JR, Koza JR. Genetic programming: on the programming of computers by means of natural selection, vol. 1. Berlin: MIT press; 1992.
38. Kuo R, Gosumolo M, Zulvia FE. Multi-objective particle swarm optimization algorithm using adaptive archive grid for numerical association rule mining. Neural Comput Appl. 2019;31(8):3559–72.
39. Kwaśnicka H, Świtalski K. Discovery of association rules from medical data-classical and evolutionary approaches. Ann Univ Mariae Curie-Sklodowska Sect AI-Inf. 2006;4(1):204–17.

40. Lent B, Swami A, Widom J. Clustering association rules. In: Proceedings 13th international conference on data engineering, IEEE; 1997. pp. 220–31.
41. Lian W, Cheung DW, Yiu S. An efficient algorithm for finding dense regions for mining quantitative association rules. Comput Math Appl. 2005;50(3–4):471–90.
42. Liu H, Abraham A, Li Y, Yang X. Role of chaos in swarm intelligence a preliminary analysis. In: Applications of soft computing, Springer; 2006. pp. 383–92.
43. Liu H, Hussain F, Tan CL, Dash M. Discretization: an enabling technique. Data Min Knowl Disc. 2002;6(4):393–423.
44. Lud MC, Widmer G. Relative unsupervised discretization for association rule mining. In: Zighed DA, Komorowski J, Żytkow J, editors. Principles of data mining and knowledge discovery. Berlin, Heidelberg: Springer; 2000. p. 148–58.
45. Martín D, Rosete A, Alcalá-Fdez J, Herrera F. A multi-objective evolutionary algorithm for mining quantitative association rules. In: 2011 11th international conference on intelligent systems design and applications, IEEE; 2011. pp. 1397–402.
46. Martínez-Ballesteros M, Troncoso A, Martínez-Álvarez F, Riquelme JC. Mining quantitative association rules based on evolutionary computation and its application to atmospheric pollution. Integr Comput-Aided Eng. 2010;17(3):227–42.
47. Mata J, Alvarez J, Riquelme J. Mining numeric association rules with genetic algorithms. In: Artificial neural nets and genetic algorithms, Springer; 2001. pp. 264–7.
48. Mata J, Alvarez JL, Riquelme JC. Discovering numeric association rules via evolutionary algorithm. In: Pacific-Asia conference on knowledge discovery and data mining, Springer; 2002. pp. 40–51.
49. Mlakar U, Zorman M, Fister I Jr, Fister I. Modified binary cuckoo search for association rule mining. J Intell Fuzzy Syst. 2017;32(6):4319–30.
50. Moreland K, Truemper K. Discretization of target attributes for subgroup discovery. In: International workshop on machine learning and data mining in pattern recognition, Springer; 2009. pp. 44–52.
51. Peious SA, Sharma R, Kaushik M, Shah SA, Yahia SB. Grand reports: a tool for generalizing association rule mining to numeric target values. In: International conference on big data analytics and knowledge discovery, Springer; 2020. pp. 28–37.
52. Poli R, Kennedy J, Blackwell T. Particle swarm optimization. Swarm Intell. 2007;1(1):33–57.
53. Qodmanan HR, Nasiri M, Minaei-Bidgoli B. Multi objective association rule mining with genetic algorithm without specifying minimum support and minimum confidence. Expert Syst Appl. 2011;38(1):288–98.
54. Rashedi E, Nezamabadi-Pour H, Saryazdi S. Gsa: a gravitational search algorithm. Inf Sci. 2009;179(13):2232–48.
55. Rashedi E, Rashedi E, Nezamabadi-pour H. A comprehensive survey on gravitational search algorithm. Swarm Evol Comput. 2018;41:141–58.
56. Salleb-Aouissi A, Vrain C, Nortet C, Kong X, Rathod V, Cassard D. Quantminer for mining quantitative association rules. J Mach Learn Res. 2013;14(1):3153–7.
57. Seki H, Nagao M. An efficient java implementation of a ga-based miner for relational association rules with numerical attributes. In: 2017 ieee international conference on systems, man, and cybernetics (SMC), IEEE; 2017. pp. 2028–33.
58. Sharma R, Kaushik M, Peious SA, Yahia SB, Draheim D. Expected vs. unexpected: Selecting right measures of interestingness. In: International conference on big data analytics and knowledge discovery, Springer; 2020. pp. 38–47.
59. Shih MY, Jheng JW, Lai LF. A two-step method for clustering mixed categroical and numeric data. Tamkang J Sci Eng. 2010;13(1):11–9.
60. Srikant R, Agrawal R. Mining quantitative association rules in large relational tables. In: Proceedings of the 1996 ACM SIGMOD international conference on Management of data; 1996. pp. 1–12.
61. Srinivas N, Deb K. Muiltiobjective optimization using nondominated sorting in genetic algorithms. Evol Comput. 1994;2(3):221–48.
62. Storn R, Price K. Differential evolution: a simple and efficient adaptive scheme for global optimization over continuous spaces. J Glob Optim. 1995;1995:23.
63. Storn R, Price K. Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces. J Glob Optim. 1997;11(4):341–59.
64. Taboada K, Gonzales E, Shimada K, Mabu S, Hirasawa K, Hu J. Association rule mining for continuous attributes using genetic network programming. IEEE J Trans Electr Electron Eng. 2008;3(2):199–211.
65. Tahyudin I, Nambo H. The combination of evolutionary algorithm method for numerical association rule mining optimization. In: Proceedings of the tenth international conference on management science and engineering management, Springer; 2017. pp. 13–23.
66. Tan SC. Improving association rule mining using clustering-based discretization of numerical data. In: 2018 international conference on intelligent and innovative computing applications (ICONIC), IEEE; 2018. pp. 1–5.
67. Tang R, Fong S, Yang XS, Deb S. Wolf search algorithm with ephemeral memory. In: Seventh international conference on digital information management (ICDIM 2012), IEEE; 2012. pp. 165–72.
68. Telikani A, Gandomi AH, Shahbahrami A. A survey of evolutionary computation for association rule mining. Inf Sci. 2020;2020:5.
69. Triguero I, García S, Herrera F. Differential evolution for optimizing the positioning of prototypes in nearest neighbor classification. Pattern Recogn. 2011;44(4):901–16.
70. Webb GI. OPUS: An efficient admissible algorithm for unordered search. J Artif Intell Res. 1995;3:431–65. https://doi.org/10.1613/jair.227
71. Webb GI. Discovering associations with numeric variables. In: Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2001; pp. 383–8.
72. Yamany W, Emary E, Hassanien AE. Wolf search algorithm for attribute reduction in classification. In: IEEE Symposium on Computational Intelligence and Data Mining (CIDM), IEEE. 2014; pp. 351–8. https://doi.org/10.1109/CIDM.2014.7008689
73. Yan D, Zhao X, Lin R, Bai D. Ppqar Parallel pso for quantitative association rule mining. Peer-to-Peer Netw Appl. 2019;12(5):1433–44.
74. Yan X, Zhang C, Zhang S. Genetic algorithm-based strategy for identifying association rules without specifying actual minimum support. Expert Syst Appl. 2009;36(2):3066–76.
75. Yang, J., Feng, Z. An effective algorithm for mining quantitative associations based on subspace clustering. In: International Conference on Networking and Digital Society IEEE; 2010;1:175–8.
76. Zhang W. Mining fuzzy quantitative association rules. In: Proceedings of 11th International Conference on Tools with Artificial Intelligence, IEEE;1999. pp. 99–102.
77. H. Zheng, J. He, G. Huang and Y. Zhang. Optimized fuzzy association rule mining for quantitative data. In: IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), IEEE; 2014. pp. 396-403. https://doi.org/10.1109/FUZZ-IEEE.2014.6891735

Appendix 3

[III]

M. Kaushik, R. Sharma, S. A. Peious, and D. Draheim. Impact-driven discretization of numerical factors: Case of two- and three-partitioning. In *Proceedings of BDA: 9th International Conference on Big Data Analytics*, pages 244–260, Cham, 2021. Springer International Publishing

Impact-Driven Discretization of Numerical Factors: Case of Two- and Three-Partitioning

Minakshi Kaushik( ), Rahul Sharma, Sijo Arakkal Peious, and Dirk Draheim

Information Systems Group, Tallinn University of Technology,
Akadeemia tee 15a, 12618 Tallinn, Estonia
{minakshi.kaushik,rahul.sharma,sijo.arakkal,dirk.draheim}@taltech.ee

Abstract. Many real-world data sets contain a mix of various types of data, i.e., binary, numerical, and categorical; however, many data mining and machine learning (ML) algorithms work merely with discrete values, e.g., association rule mining. Therefore, the discretization process plays an essential role in data mining and ML. In state-of-the-art data mining and ML, different discretization techniques are used to convert numerical attributes into discrete attributes. However, existing discretization techniques do not reflect best the impact of the independent numerical factor onto the dependent numerical target factor. This paper proposes and compares two novel measures for order-preserving partitioning of numerical factors that we call *Least Squared Ordinate-Directed Impact Measure* and *Least Absolute-Difference Ordinate-Directed Impact Measure*. The main aim of these measures is to optimally reflect the impact of a numerical factor onto another numerical target factor. We implement the proposed measures for two-partitions and three-partitions. We evaluate the performance of the proposed measures by comparison with human-perceived cut-points. We use twelve synthetic data sets and one real-world data set for the evaluation, i.e., school teacher salaries from New Jersey (NJ). As a result, we find that the proposed measures are useful in finding the best cut-points perceived by humans.

Keywords: Discretization · Partitioning · Numerical attributes · Data mining · Machine learning · Association rule mining

1 Introduction

In data mining and machine learning, discretization is an essential data preprocessing step to achieve discretized values from numeric columns. Numeric attributes can be discretized by partitioning the range of numeric attributes into different intervals. In-state of the art, several discretization approaches such as equi-depth, equi-width [3], ID3 [19], etc., are proposed. However, the existing

S. N. Srirama et al. (Eds.): BDA 2021, LNCS 13147, pp. 244–260, 2021.
https://doi.org/10.1007/978-3-030-93620-4_18

discretization methods do not provide optimal results for the discretization process, and they also have some drawbacks like information loss, etc., for mining algorithms.

In this paper, we propose an optimal way to find out intervals or partitions of a numerical attribute that reflect best the impact of one independent numerical attribute on a dependent numerical attribute. We provide a *Least Squared Ordinate-Directed Impact Measure (LSQM)* and *Least Absolute-Difference Ordinate-Directed Impact Measure (LADM)* for order-preserving partitioning of numerical factors. The measures provide a simple way to search the appropriate cut-points for finding the optimal partitions. For best cut-points, order-preserving partitioning on an independent factor is performed and implemented for two-partitions and three-partitions. The order of the independent variable is preserved using the value of data points. Therefore, the value of data points of one partition will always be less than the value of data points of the next partition. The measures' performance is assessed using one real-world data set and twelve synthetic data sets (including two-step and three-step functions). The outcomes are first compared to human perceived cut-points, and then their outcomes are also compared to one another. The following are the key contributions of this article:

1. We develop two measures to find out the partitions which best reflect the impact of one numerical factor on another numerical factor.
2. We evaluate the proposed measures on one real-world data set and twelve synthetic data sets, including two-step functions, three-step functions, compare it with human-perceived cut-points.
3. We provide the comparison of the results of both measures.

In Sect. 2, related work is discussed. In Sect. 3, we discuss the motivation of the proposed measures with an example. We provide methods in Sect. 4. In Sect. 5, we evaluate the proposed measures with a variety of data sets, including one real-world data set, two-step, and three-step data points. We finish with the paper with a conclusion in Sect. 6.

2 Related Work

In the literature, many concepts related to correlation and inter-dependency among variables are discussed in statistical reasoning, e.g. Pearson correlation [17,23], linear regression [15], ANOVA (Analysis of Variance) [8] etc. However, these tools do not find the partition of the numerical variable that reflects best the impact on another variable.

The idea of this research emerged from the research on partial conditionalization [5], association rule mining [20,21] and numerical association rule mining [9,22]. In these papers, the discretization process is discussed as an essential step for numerical association rule mining. We have also presented a tool named Grand report [18] which reports the mean value of a chosen numeric target column concerning all possible combinations of influencing factors. The measures proposed in this paper are important for discretization, which is an essential step in frequent itemset mining, especially for quantitative association rule mining [22] or numerical association rule mining.

There are various discretization processes available in the literature. Researchers and data scientists proposed different algorithms using different methods such as clustering, partitioning. However, these methods mainly focus on discretizing the continuous factor by finding the appropriate cut-points to make suitable intervals, or some of them use distance measures to create clusters. In this paper, our work is related to discretization and provides the partitions of one factor that best describes the impact of one factor on another.

Mehta et al. [14] worked in this direction and proposed a PCA-based unsupervised correlation preserving discretization method, which discretizes continuous attributes in multivariate data sets. The work ensures the use of all attributes simultaneously to decide the cut-points in place of one attribute at a time.

Dougherty et al. [4] reviewed and classified discretization methods along three separate axes; global versus local, supervised versus unsupervised, and static versus dynamic. Dougherty et al. [4] compared binning, unsupervised discretization method to entropy-based and purity-based supervised methods. Global methods, such as binning, partition all the data set attributes into regions, and each attribute is independent of other attributes. The static methods discretize each feature separately, whereas dynamic methods obtain inter-dependencies among features via conducting the search through space.

Liu et al. [12] performed a systematic study of existing discretization methods and proposed a hierarchical framework for discretization methods from the perspective of splitting and merging. The unsupervised static discretization methods such as equal-width and equal-frequency are simple and relevant to our work. The Equal-width discretization algorithm uses the minimum and maximum values of the continuous attribute and then divides the range into equal-width intervals called bins. The equal-frequency algorithm determines an equal number of continuous values and places them in each bin.

Ludl and Widmer [13] present RUDE (Relative Unsupervised Discretization) algorithm for discretizing numerical and categorical attributes. The algorithm combines the aspect of both supervised and unsupervised discretization. The algorithm is implemented in three steps: pre-discretizing, structure projection, and merging split points. The primary step is structure projection which projects the structure of each source attribute onto the target attribute. Then clustering is performed using projected intervals and merges split points if the difference is less than or equal to the user-specified minimum difference.

Recently, H. M. Abachi et al. [1] worked on statistical unsupervised method SUFDA (Statistical Unsupervised Feature Discretization Algorithm). The SUFDA tries to provide discrete intervals with low temporal complexity and good accuracy by decreasing the differential entropy of the normal distribution. Multi-scale and information entropy-based discretization method is also proposed in [24]. In 1988, Eubank [6] and Konno et al. [10] worked on the best piecewise constant approximation of a function f of single variable. Eubank used the population quantile function as a tool to show the best piecewise constant approximation problem. Later Bergerhoff [2] proposed an approach using particle swarm optimization for finding optimal piecewise constant approximations of one-dimensional signals. Our work is different because we are not using signals, and our main focus is on data sets that use several data points for one value of influencing factor.

3 Motivation

A number of discretization methods have been proposed in the state of the art [7,11]; however, they have not considered the type of target attribute, such as binary, categorical, or numerical. We use numerical attributes as both influencing and response factors in the proposed impact-driven discretization method. In general, when one variable influences another, the human brain is trained to notice changes and can easily discern compartments or partitions. However, in a real-world data set, it is difficult for a human to determine the most suitable compartments; for example, both the *Experience* and *Salary* attributes are numerical in the graph shown in Fig. 1. A human cannot easily find the appropriate compartment using this graph. As a result, the proposed measures partition the numerical attribute and determine its impact on a target attribute. This section presents a motivating example explaining why a specific measure is required to locate the suitable compartments.

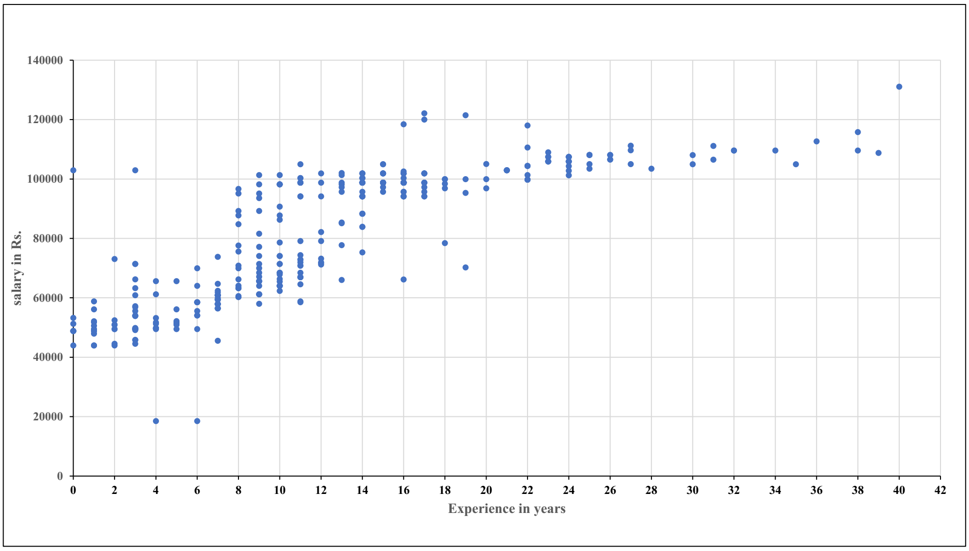


Fig. 1. An example for motivation of real-world data set.

4 Our Approach

The basic idea of our approach is to take one numerical independent variable and one target variable from a data set and discretize the independent variable in such a way as to find the appropriate cut-points, which are not observed easily by humans.

4.1 Key Intuition

We claim to discretize the independent numerical attribute by using order-preserving partitioning and see the impact on the numerical target attribute. In Fig. 2, we provide the graph for two-step data points where *X factor* is the numerical independent attribute, and *Y factor* is the numerical target attribute. It is the extreme case where data points are distributed as a step-function. In this case, humans can easily find out the cut-points without any difficulty. We evaluate the same data set with the proposed measures and compare the results with human perceived partitions.

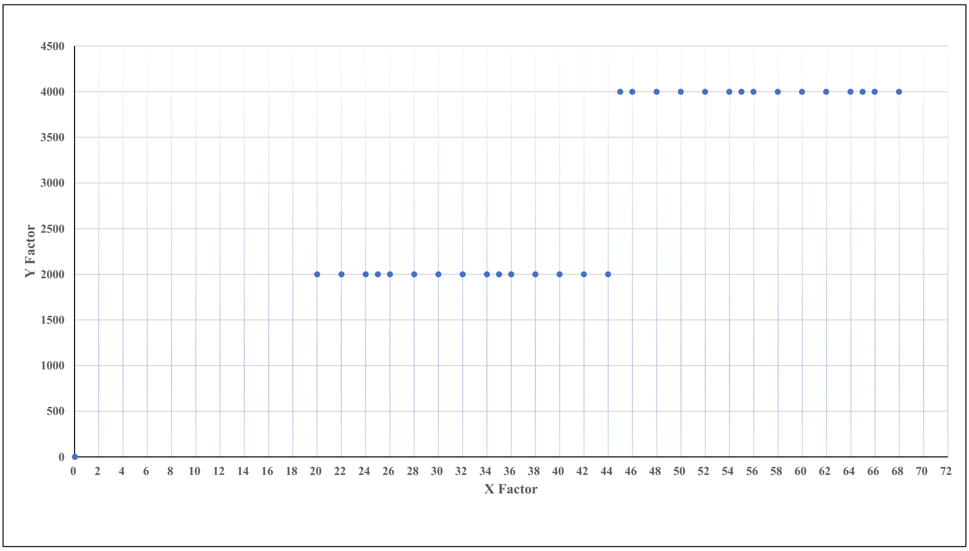


Fig. 2. Graph for two-step data-points (DS5 data set).

4.2 Step Function

In a *Step function f*, the domain is partitioned into several intervals. *f(x)* is constant for each interval, but the constant can be different for each interval. The different constant values for each interval create the jumps between horizontal line segments and develop a staircase which is also known as a step function.

Definition 1. *A Step function f on interval [a,b] is a piece wise constant function which contains many finite pieces. There exist a partition $P = \{a = x_0, x_1, \ldots, x_n = b\} \in p[a,b]$ such that $f(x)$ for all $x \in (x_{r-1}, x_r)$ for each $r \in \{1, 2, 3, \ldots, n\}$. The jump of x_r for $r \in \{0, 1, 2, \ldots, n\}$ is defined to be $f(x_r^+) - f(x_r^-)$.*

4.3 Definitions

To compute appropriate cut-points of independent numerical variables, we introduce the following impact measures as per the below definition.

Definition 2 (Least Squared Ordinate-Directed Impact Measure). *Given $n \geq 2$ real-valued data points $(< x_i, y_i >)_{1 \leq i \leq n}$, we define the* least squared ordinate-directed impact measure *for k-partitions (with $k-1$* cut-points*) as follows:*

$$\min_{i_0=0<i'_1<\ldots<i'_{k-1}<i'_k=n} \sum_{j\,=\,1}^{k} \sum_{i'_{j-1}<i"\leq i'_j} (y_{i"} - \mu_{i'_{j-1}<\phi\leq i'_j})^2 \tag{1}$$

where the average of data values in a partition $\mu_{a<\phi\leq b}$ *between indexes a and b ($a < b \leq n$) is defined as*

$$\mu_{a<\phi\leq b} = \frac{\sum\limits_{a<\phi\leq b} y_\phi}{b-a} \tag{2}$$

In (1), we have that i'_j is the highest element in the *j-th* partition, where *highest element* means the data point with the highest index.

Definition 3 (Least Absolute-Difference Ordinate-Directed Impact Measure). *Given $n \geq 2$ real-valued data points $(<x_i, y_i>)_{1 \leq i \leq n}$, we define the* least absolute-difference ordinate-directed impact measure *for k-partitions (with $k-1$* cut-points*) as follows:*

$$\min_{i_0=0<i'_1<\ldots<i'_{k-1}<i'_k=n} \sum_{j\,=\,1}^{k} \sum_{i'_{j-1}<i"\leq i'_j} |y_{i"} - \mu_{i'_{j-1}<\phi\leq i'_j}| \tag{3}$$

where the average of data values in a partition $\mu_{a<\phi\leq b}$ *between indexes a and b ($a < b \leq n$) is defined as*

$$\mu_{a<\phi\leq b} = \frac{\sum\limits_{a<\phi\leq b} y_\phi}{b-a} \tag{4}$$

Listing 1. Pseudo-code for finding the three-partitions that reflect best the impact of a numerical variable on another numerical variable.

```
FUNCTION finding the first partition(array_d)
  FOR a=0 to (array_d.length-2)
    MEAN_POINT1: mean(array_d[0 to a+1])
    FOR b= 0 to a+1
      ABS_DIFF1: absolute.difference(array_d[b]-MEAN_POINT1)
      C1_SUM: C1_SUM + SQUARE(ABS.DIFF1)
    ENDFOR
    FOR j =(a+1) to (array_d.length-1)
        MEAN_POINT2: mean(array_d[a+1 to j+1])
        FOR i = a+1 to (j+1)
           ABS.DIFF2: absolute.difference(array_d[i]-MEAN_POINT2)
           C2_SUM: C2_SUM + SQUARE(ABS.DIFF2)
        ENDFOR
        MEAN_POINT3: mean(array_d[j+1 to (array_d.length)]
        FOR k = j+1 to (array_d.length)
           ABS_DIFF3: absolute.difference(array_d[k]-MEAN_POINT3)
           C3_SUM = C3_SUM + SQUARE(ABS_DIFF3)
        ENDFOR
        TOTAL_SUM = C1_SUM1 + C2_SUM2 + C3_SUM3
        IF a ==0 AND j == 1 THEN
           CUT1 = 0
           CUT2 = 1
           temporary_var= TOTAL_SUM
        ENDIF
        IF TOTAL_SUM < temporary_var
           CUT1 = a
           CUT2 = j
           temporary_var = TOTAL_SUM
        ENDIF

     ENDFOR
  ENDFOR
  PRINT(CUT1)
  PRINT(array_d.[CUT1])
  PRINT(CUT2)
  PRINT(array_d.[CUT2])
ENDFUNCTION
```

4.4 Method

Let D be a collection of n data points $D = (\langle x_i, y_i \rangle)_{1 \leq i \leq n}$, where $\langle x_i, y_i \rangle$ are data points of real values. As per (1), the proposed measure *LSQM* computes appropriate cut-points. The number of cut-points is $k - 1$, where k is the number of partitions suggested by the user. The measure first calculates the squared difference of the y-value of each data point of the current partition. In (1), the condition

($i_{j-1} < r \leq i_j$) requires that the index of data points of the current partition should be greater than the highest index of the previous partition and less than or equal to the highest index of the current partition. After summing up the squared differences of the several partitions, it selects the minimum values, which correspond to the appropriate cut-points. For the second measure *LADM*, we just take the sum of the absolute differences of the several partitions. Next, we first provide the pseudo-code for the case of three partitions ($k = 3$); and then, we compute the cut-points for two partitions and three partitions. In Listing 1, we provide pseudo-code for the three-partitioning approach according to Definition 2.

5 Evaluation

In this section, we experimentally validate the proposed measure in terms of the quality of the resulting discretization and its ability to find the independent variable's impact on the target variable. In Fig. 2, a two-step function graph is shown. In the graph, manual selection of cut-points can be performed for two-partitions and three-partitions easily. However, we demonstrate the cut-points after implementing the proposed measure on the same step-data sample and then compare its cut-points with the human perceived manual methods.

5.1 Data Sets

We have conducted the experiment using twelve synthetic data sets and one real-world data set. The real-world data set, New Jersey (NJ) school teacher salaries (2016) [16] is sourced from the (NJ) Department of Education. It contains 138715 records and 15 attributes. However, we have reduced the number of rows from the data set to analyze the cut-points visually. We have taken only initial 350 rows from the data set. The Data set NJ Teacher Salaries (2016) consists of salary, job and experience data for the teachers and employees in New Jersey schools. We are interested in the column {experience_total} and {salary}. The column {experience_total} is a numeric and independent attribute, whereas {salary} is a numeric target attribute. The twelve synthetic data sets are DS1 to DS12[1]. These twelve synthetic data sets have only two attributes named *Age* and *Salary* which are numeric. These data sets have a different number of rows and different values of attributes. In Table 1, we describe the data sets. As the limit of pages, all the graphs for all data sets are not included in this article. Repository of data sets has been given on the GitHub (See footnote 1).

5.2 Results and Discussion

Two-Partitioning. For two-partitioning, $k = 2$, we need one cut-point. We provide a graph for two-step data points in Fig. 2. The data set DS5 is from the

[1] https://github.com/minakshikaushik/Least-square-measure.git.

Table 1. Data sets used in evaluation.

Dataset	Number of records	Number of attributes
NJ Teacher Salaries(2016)	347	15
DS1	31	2
DS2	31	2
DS3	35	2
DS4	24	2
DS5	30	2
DS6	100	2
DS7	40	2
DS8	31	2
DS9	30	2
DS10	30	2
DS11	45	2
DS12	30	2

list of synthetic data sets. The data set DS5 is a sample of two-step function data points. We use this data set for the manual selection method and later implement the *LSQM* and *LADM* on the same data set. In the given data set, the human would identify 44 as the natural cut-point of the two partitions 0–44 and 45–72, see Fig. 3. Next, we implement the proposed measures on the same data set and see the cut-points.

Table 2. Comparison of the two-step and the three-step function using manual selection of cut-points and using proposed measures.

Dataset	DS5 (Two-step function)	DS12 (Three-step function)
Two-partitioning		
Manual cut point	44	35
LSQM cut point	44	35
LADM cut point	44	35
Three-partitioning		
Manual cut point	(20,44)	(35,52)
LSQM cut point	(20,44)	(35,52)
LADM cut point	(20,44)	(35,52)

Three-Partitioning. For three-partitioning, $k = 3$, we need two cut-points. We provide a graph for three-step data points in Fig. 4. We use data set DS12 as an extreme case of a three-step function. Earlier described in two-partitioning, we

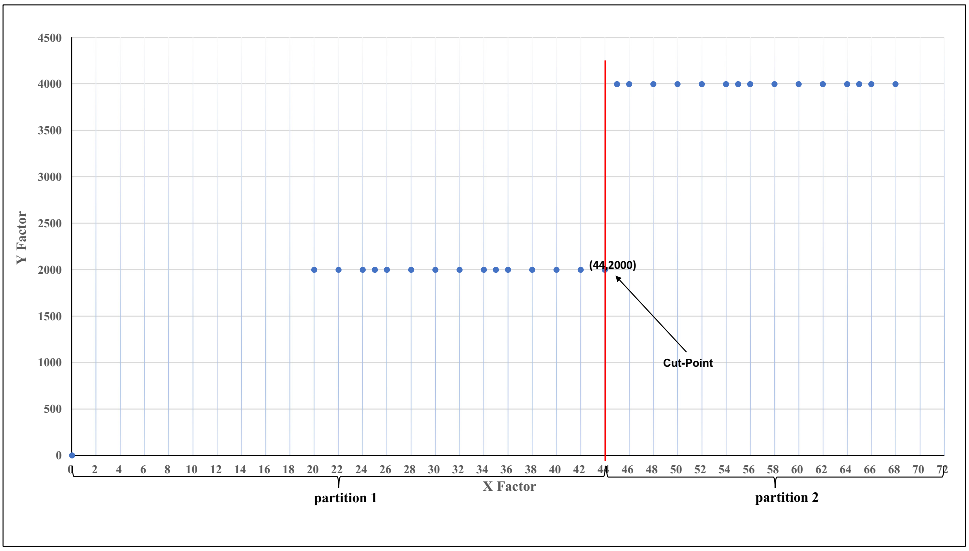


Fig. 3. Graph for showing the cut-point and two-partitions using manual method.

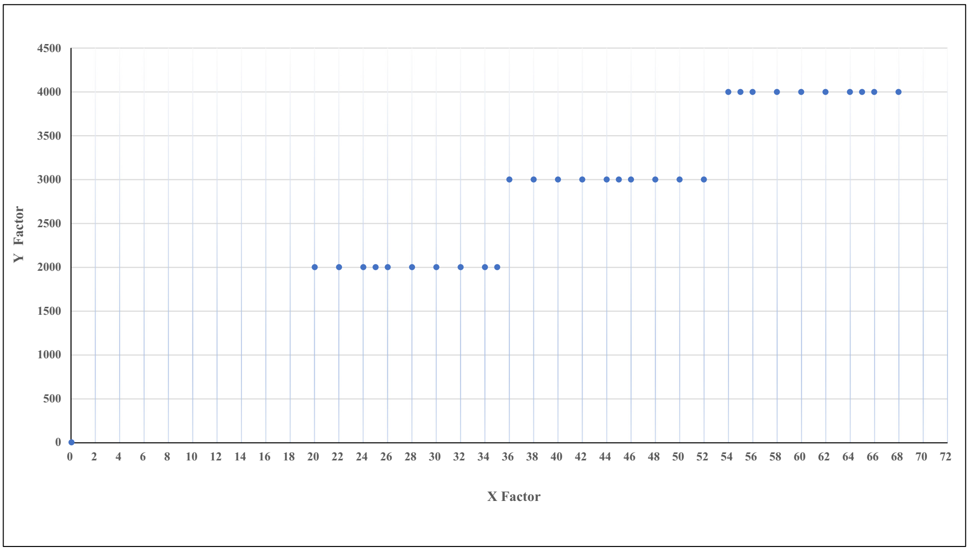


Fig. 4. Graph for three-step data-points (DS12 data set).

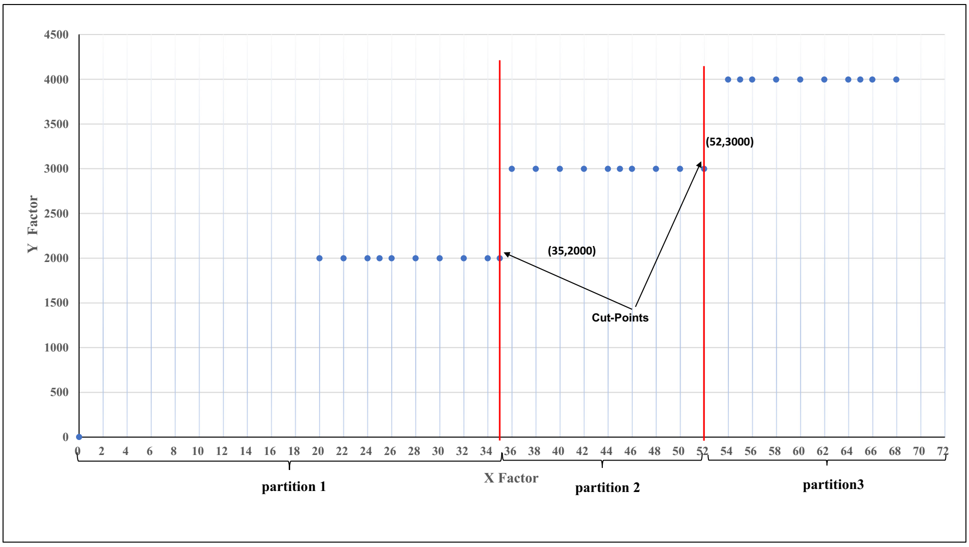


Fig. 5. Graph for showing the cut-points and three-partitions using manual method.

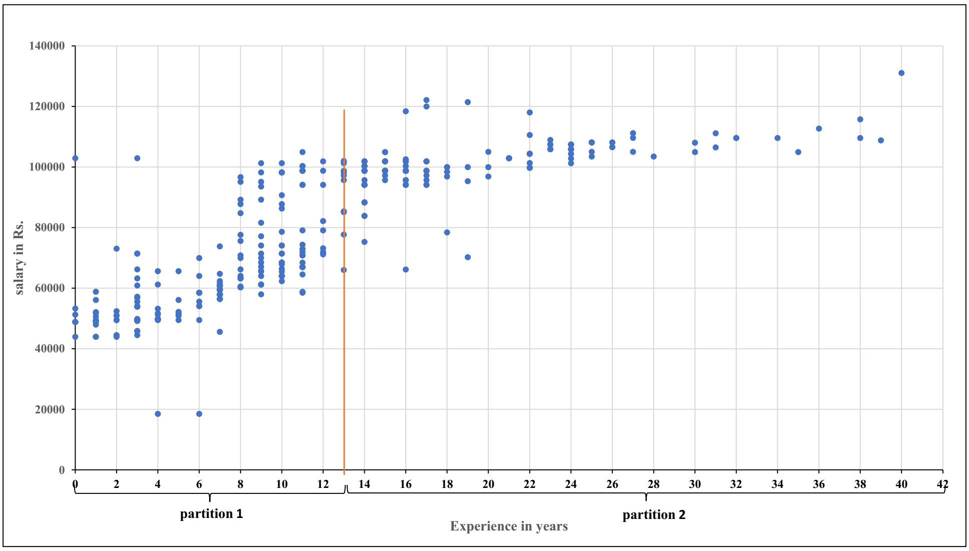


Fig. 6. Graph for showing the one cut-point and two-partitions using *Least Squared Ordinate-Directed Impact Measure* on real-world data set.

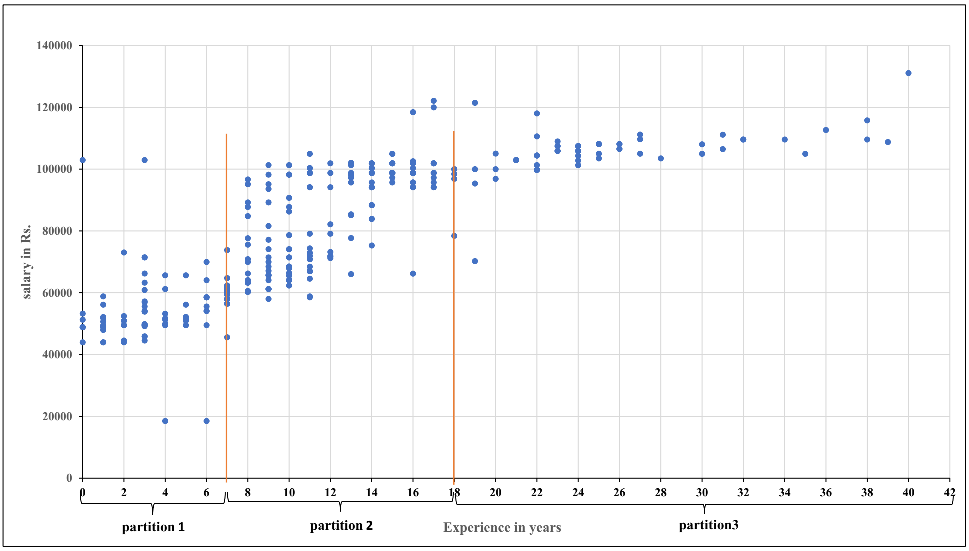


Fig. 7. Graph for showing the two cut-points and three-partitions using *Least Squared Ordinate-Directed Impact Measure* on real-world data set.

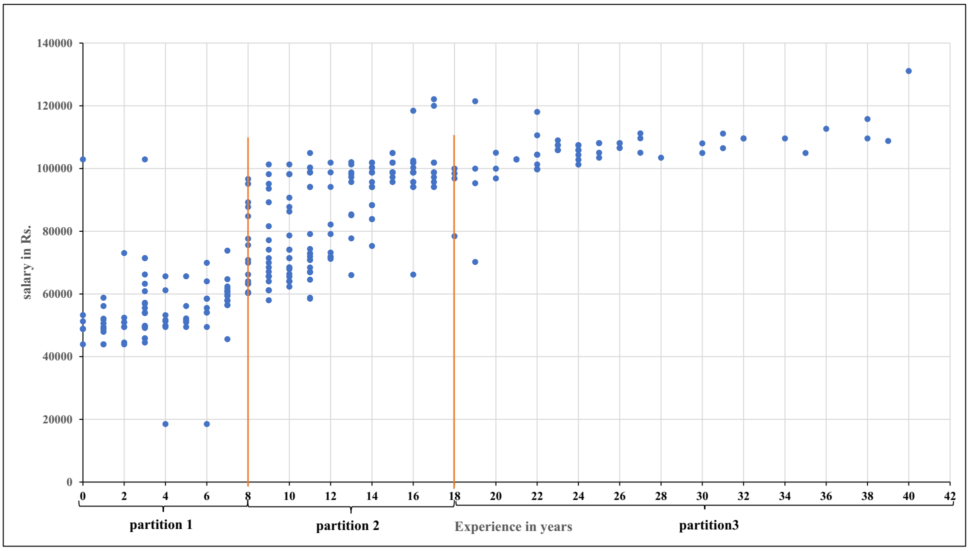


Fig. 8. Graph for showing the two cut-points and three-partitions using *Least Absolute-Difference Ordinate-Directed Impact Measure* on real-world data set.

Table 3. Result of *Least Squared Ordinate-Directed Impact Measure* using two-partitions and three-partitions approach on different data samples.

Dataset	Two-partitions	Three-partitions	
	Cut-point	Cut-point1	Cut-point2
NJ Teacher Salaries(2016)	13	18	7
DS5 (Two-step data-points)	44	20	44
DS12 (Three-step data-points)	35	35	52
DS1	52	52	54
DS2	52	32	52
DS3	25	25	56
DS4	40	29	40
DS6	20	12	24
DS7	19	14	27
DS8	32	32	52
DS9	52	35	52
DS10	35	35	52
DS11	42	32	42

Table 4. Result of *Least Absolute Ordinate-Directed Impact Measure* using two-partitions and three-partitions approach on different data samples.

Dataset	Two-partitions	Three-partitions	
	Cut-point	Cut-point1	Cut-point2
NJ Teacher Salaries(2016)	13	18	8
DS5 (Two-step data-points)	44	20	44
DS12 (Three-step data-points)	35	35	52
DS1	52	52	54
DS2	52	32	52
DS3	25	25	56
DS4	40	29	40
DS6	20	12	25
DS7	19	15	27
DS8	32	32	52
DS9	52	35	52
DS10	35	35	52
DS11	42	32	42

use this data set for the manual selection method and implement the proposed measures. By using the manual method human would identify 35 and 52 as two cut-points of the three-partitions 0–35, 36–52 and 53–72 in Fig. 5. After implementing the proposed measures on the same data set, we can verify the cut-points.

We implement both measures on data sets DS5 and DS12. We compare manually selected cut-points with cut-points provided by *LSQM* and *LADM* for two-partitioning and three-partitioning. As shown in Table 2, the cut-points for data set DS5 are the same for the manual selection method and *LSQM* and *LADM* measures. In the same way, cut-points for data set DS12 are also the same for manual method and proposed measures.

Next, we implement both measures on real-world data set NJ Teacher Salaries (2016) and the rest ten synthetic data sets (DS1, DS2, DS3, DS4, DS6, DS7, DS8, DS9, DS10, DS11). Figure 6 shows one cut-point for two-partitioning using *LSQM* measure on data set NJ Teacher Salaries(2016). Figure 7 shows two cut-points for three-partitioning using *LSQM* measure on data set NJ Teacher Salaries(2016). We received one cut-point 13 using *LSQM* for two-partitioning and received two cut-points 18 and 7 for three-partitioning for the same data set. Figure 8 is showing the two cut-points 18 and 8 after applying *LADM* measure. Both the measures *LSQM* and *LADM* provide the same cut-point for two-partitioning as given in Fig. 6. However, their cut-point for three partitioning is different and it is given in Figs. 7 and 8.

Table 5. Comparison of cut-points provided for measures *LSQM* and *LADM* for two-partitioning and three-partitioning.

Dataset	k=2		k=3				Deviation
	LSQM	LADM	LSQM		LADM		
	I	I	I	II	I	II	
NJ Teacher Salaries(2016)	13	13	18	7	18	8	Yes (In cut point2)
DS5 (Two-step data-points)	44	44	20	44	20	44	No
DS12 (Three-step data-points)	35	35	35	52	35	52	No
DS1	52	52	52	54	52	54	No
DS2	52	52	32	52	32	52	No
DS3	25	25	25	56	25	56	No
DS4	40	40	29	40	29	40	No
DS6	20	20	12	24	12	25	Yes (In cut point2)
DS7	19	19	14	27	15	27	Yes(In cut point1)
DS8	32	32	32	52	32	52	No
DS9	52	52	35	52	35	52	No
DS10	35	35	35	52	35	52	No
DS11	42	42	32	42	32	42	No

The results of *LSQM* and *LADM* measures for two-partitioning and three-partitioning on all the data sets are given in Tables 3 and 4, respectively. In Table 5, We have compared the results of both measures for $k = 2$ and $k = 3$. As we can see in the Table 5, NJ Teacher Salaries(2016) data set has one point deviation in cut-point2 for $k = 3$. The measure *LSQM* cut-point2 has a value of 7, whereas *LADM* cut-point2 has a value of 8. The data sets DS6 also have only one point difference in cut-point2 that is 24 and 25 whereas DS7 has one point difference in cut-point1 14 and 15. We observed a deviation in the result when $k = 3$. Except for these data sets, all the data sets have the same cut-points for both measures. After analyzing and comparing the results of both measures, we find out that the outcomes of both proposed measures are approximately similar. We analyzed that the proposed measures provide the cut-points which reflect best the impact of one independent numerical factor on a dependent numerical target factor.

6 Conclusion

This paper aimed to find the partitions that best reflect the impact of a numerical independent variable on a dependent numerical target variable. We proposed two *Least Squared Ordinate-Directed Impact Measure* and *Least Absolute-Difference Ordinate-Directed Impact Measure*. In the case of step functions, there is an immediate, intuitive understanding of best cut-points regarding human judgment. Therefore, we evaluated the performance of these measures for two-step staircase data sets (step functions), three-step staircase data set and arbitrary data set (non-step function). We examined that the proposed measures provide the same human perceived cut-points for two-step staircase and three-step staircase data sets. Furthermore, the results of both proposed measures on twelve synthetic data sets and one real-world data set are approximately similar. As future work, we plan to evaluate the proposed measures in long series of data repositories against respective human judgments (by data experts and domain experts). We also plan to implement the measure for arbitrary numbers of k-partitions beyond two- and three-partitions. A particular challenge will be to come up with *inter*-measures for comparing partitions of different numbers of k-partitions.

Acknowledgements. This work has been conducted in the project "ICT programme" which was supported by the European Union through the European Social Fund.

References

1. Abachi, H.M., Hosseini, S., Maskouni, M.A., Kangavari, M., Cheung, N.M.: Statistical discretization of continuous attributes using Kolmogorov-Smirnov test. In: Wang, J., Cong, G., Chen, J., Qi, J. (eds.) Databases Theory and Applications, pp. 309–315. Springer International Publishing, Cham (2018)

2. Bergerhoff, L., Weickert, J., Dar, Y.: Algorithms for piecewise constant signal approximations. In: 27th European Signal Processing Conference (EUSIPCO), pp. 1–5. IEEE (2019)
3. Catlett, J.: On changing continuous attributes into ordered discrete attributes. In: Kodratoff, Y. (ed.) EWSL 1991. LNCS, vol. 482, pp. 164–178. Springer, Heidelberg (1991). https://doi.org/10.1007/BFb0017012
4. Dougherty, J., Kohavi, R., Sahami, M.: Supervised and unsupervised discretization of continuous features. In: Machine learning proceedings 1995, pp. 194–202. Elsevier (1995)
5. Draheim, D.: Generalized Jeffrey Conditionalization: A Frequentist Semantics of Partial Conditionalization. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-69868-7
6. Eubank, R.: Optimal grouping, spacing, stratification, and piecewise constant approximation. Siam Rev. **30**(3), 404–420 (1988)
7. Garcia, S., Luengo, J., Sáez, J.A., Lopez, V., Herrera, F.: A survey of discretization techniques: taxonomy and empirical analysis in supervised learning. IEEE Trans. Knowl. Data Eng. **25**(4), 734–750 (2012)
8. Gelman, A., et al.: Analysis of variance - why it is more important than ever. Ann. Stat. **33**(1), 1–53 (2005)
9. Kaushik, M., Sharma, R., Peious, S.A., Shahin, M., Ben Yahia, S., Draheim, D.: On the potential of numerical association rule mining. In: Dang, T.K., Küng, J., Takizawa, M., Chung, T.M. (eds.) FDSE 2020. CCIS, vol. 1306, pp. 3–20. Springer, Singapore (2020). https://doi.org/10.1007/978-981-33-4370-2_1
10. Konno, H., Kuno, T.: Best piecewise constant approximation of a function of single variable. Oper. Res. Lett. **7**(4), 205–210 (1988)
11. Kotsiantis, S., Kanellopoulos, D.: Discretization techniques: a recent survey. GESTS Int. Trans. Comput. Sci. Eng. **32**(1), 47–58 (2006)
12. Liu, H., Hussain, F., Tan, C.L., Dash, M.: Discretization: an enabling technique. Data Min. Knowl. Discov. **6**(4), 393–423 (2002)
13. Lud, M.C., Widmer, G.: Relative unsupervised discretization for association rule mining. In: Zighed, D.A., Komorowski, J., Żytkow, J. (eds.) PKDD 2000. LNCS (LNAI), vol. 1910, pp. 148–158. Springer, Heidelberg (2000). https://doi.org/10.1007/3-540-45372-5_15
14. Mehta, S., Parthasarathy, S., Yang, H.: Toward unsupervised correlation preserving discretization. IEEE Trans. Knowl. Data Eng. **17**(9), 1174–1185 (2005)
15. Montgomery, D.C., Peck, E.A., Vining, G.G.: Introduction to Linear Regression Analysis. John Wiley & Sons, Hoboken (2021)
16. Naik, S.: Nj teacher salaries (2016). https://data.world/sheilnaik/nj-teacher-salaries-2016
17. Pearson, K.: VII. Note on regression and inheritance in the case of two parents. Proc. Roy. Soc. London **58**(347–352), 240–242 (1895)
18. Arakkal Peious, S., Sharma, R., Kaushik, M., Shah, S.A., Yahia, S.B.: Grand reports: a tool for generalizing association rule mining to numeric target values. In: Song, M., Song, lY., Kotsis, G., Tjoa, A.M., Khalil, I. (eds.) DaWaK 2020. LNCS, vol. 12393, pp. 28–37. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-59065-9_3
19. Quinlan, J.R.: Induction of decision trees. Mach. Learn. **1**(1), 81–106 (1986)
20. Shahin, M., et al.: Big data analytics in association rule mining: a systematic literature review. In: International Conference on Big Data Engineering and Technology (BDET), pp. 40–49. Association for Computing Machinery (2021)

21. Sharma, R., Kaushik, M., Peious, S.A., Yahia, S.B., Draheim, D.: Expected vs. unexpected: selecting right measures of interestingness. In: Song, M., Song, I.Y., Kotsis, G., Tjoa, A.M., Khalil, I. (eds.) DaWaK 2020. LNCS, vol. 12393, pp. 38–47. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-59065-9_4
22. Srikant, R., Agrawal, R.: Mining quantitative association rules in large relational tables. In: Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data, pp. 1–12 (1996)
23. Stigler, S.M.: Francis galton's account of the invention of correlation. Stat. Sci. **4**, 73–79 (1989)
24. Xun, Y., Yin, Q., Zhang, J., Yang, H., Cui, X.: A novel discretization algorithm based on multi-scale and information entropy. Appl. Intell. **51**(2), 991–1009 (2021)

Appendix 4

[IV]

M. Kaushik, R. Sharma, M. Shahin, S. A. Peious, and D. Draheim. An analysis of human perception of partitions of numerical factor domains. In *Proceedings of iiWAS 2022 – the 24th International Conference on Information Integration and Web Intelligence*, pages 137–144, Cham, 2022. Springer Nature Switzerland

An Analysis of Human Perception of Partitions of Numerical Factor Domains

Minakshi Kaushik( ), Rahul Sharma, Mahtab Shahin, Sijo Arakkal Peious, and Dirk Draheim

Information Systems Group, Tallinn University of Technology, Akadeemia tee 15a, 12618 Tallinn, Estonia
{minakshi.kaushik,rahul.sharma,mahtab.shahin, sijo.arakkal,dirk.draheim}@taltech.ee

Abstract. In Machine learning (ML), several discretization techniques and mathematical approaches are used to partition numerical data attributes. However, cut-points retrieved by discretizing techniques often do not match with human perceived cut-points. Therefore, understanding the human perception for discretizing the numerical attribute is important for developing an effective discretizing technique. In this paper, we conduct a study of human perception of partitions in numerical data that reflects best the impact of one independent numerical attribute on another dependent numerical attribute. We aim to understand how expert data scientists and statisticians partition numerical attributes under different types of data points, such as dense data points, outliers, and uneven random points. The findings lead to an interesting discussion about the importance of human perception under distinct kinds of data points for finding partitions of numerical attributes.

Keywords: Discretization · Partitioning · Numerical attributes · Data mining · Machine learning · Human perception

1 Introduction

Discrete values are significantly used in statistics, machine learning, and data mining. Moreover, to find the intervals of numeric attributes, several discretization techniques are presented in the literature [6,12,13]. However, these techniques are unable to find the ideal intervals with appropriate ranges and it is still difficult to get an ideal discretizer.

Humans can easily visualize the ideal partitions and even the number of compartments in extreme situations (like step-functions). However, in some other unusual cases, e.g., mixed data point, uneven random data points, the partition ranges completely depend on data experts' perceptions and opinions. Perceptual conception is an important factor in developing an automated measure for

This work has been partially conducted in the project "ICT programme" which was supported by the European Union through the European Social Fund.

E. Pardede et al. (Eds.): iiWAS 2022, LNCS 13635, pp. 137–144, 2022.
https://doi.org/10.1007/978-3-031-21047-1_13

discretizing numerical attributes. However, in the state of the art discretization techniques, human perceptions and observations are overlooked.

We conduct this study to identify the typical patterns of human perception in partitioning numerical attributes. We have also presented an order-preserving partitioning method to find the partitions of numerical attributes that reflect best the impact of one independent numerical attribute on a dependent numerical attribute [10]. We aim to investigate the impact of data points' features on human perception when partitioning numerical attributes. We mainly focus on data point density, the effect of outliers, uneven random distribution, and linear function while performing perceptual analysis. We set four hypotheses related to the data points of partitions that can influence human interpretation. The human responses are collected through several experiments with data scientists and machine learning experts. We used nine synthetic and three real-world datasets to create a series of graphs for the experiment. This study's concept is inspired by previous studies on partial conditionalization [3,4], association rule mining [16,19], and numerical association rule mining [11,20]. These articles cover the discretization process as an important stage in numerical association rule mining. Earlier, we have also presented a tool named Grand report [15] and a framework [17,18] for the unification of ARM, statistical reasoning, and online analytical processing.

The paper is organized as follows. In Sect. 2, we discuss related work. We formulate hypotheses in Sect. 3. Then we describe the design of the experiment in Sect. 4. We perform analysis and present the results in Sect. 5. We finish the paper with a conclusion in Sect. 6.

2 Related Work

Many studies have used human perception to evaluate various techniques. These studies primarily focused on visual perceptual analysis. However, they are not completely related to discretization. For example, Etemadpour et al. [5] conducted a perception-based evaluation of high-dimensional data where humans were asked to identify clusters and analyze distances inside and across clusters. Demiralp et al. [2] used human judgments to estimate perceptual kernels for visual encoding variables such as shape, size, color, and combinations. The experiment used Amazon's Mechanical Turk platform, with twenty Turkers completing thirty MTurk jobs. In [1] authors also evaluated bench-marking clustering algorithms based on human perception of clusters in 2D scatter plots. The authors' main concern was how well existing clustering algorithms corresponded to human perceptions of clusters. Our work is also related to considering human perceptions when discretizing numerical attributes.

3 Hypotheses

In this study, we want to see if the distances between the data points matter or if other characteristics influence human perception when finding the cut-points to partition a numerical attribute. We make the following hypotheses, which investigate how different aspects affect humans' partitioning process.

- H1: We expect that the density of data points influences the response.
- H2: We expect that outliers influence human responses.
- H3: Linear data functions will be partitioned using the mean of the function.
- H4: Random distribution of data points influences the responses.

4 Design of Experiment

We provide a set of graphs and discussed them with our team to create a diverse collection of graphs with different data points. Finally, twelve graphs were selected to be shared with humans to partition the data, as given in Fig. 1. These graphs are obtained from nine synthetic datasets (D1 to D9) and three real-world datasets (D10 to D12). The synthetic datasets (D1 to D9) consist only of two numerical attributes. The graph D10 is drawn from a real-world dataset DC public government employees [8]. It contains 33,424 records of DC public government employees and their salaries in 2011. This dataset is sourced from the washington times via freedom of information act (FOIA) requests. The dataset D11 is the Heart Disease dataset [7] and is sourced from the UCI machine learning repository. This dataset has 13 attributes and 303 records. We used attribute {Age} and {Cholesterol} for drawing the graph. The graph D12 is drawn from New Jersey (NJ) school teacher salaries (2016) [14] sourced from the New Jersey (NJ) Department of Education. It contains 138715 records and 15 attributes. We have only taken the initial 23000 rows from the dataset. We are interested in the column {experience_total} and {salary}. A copy of all these datasets is available in the GitHub repository [9]. We designed a Google form with a number of graphs (Fig. 1) and questions to get responses from individuals and their perceptions on discretization. The Google form was distributed to fifty DS/ML experts and non-experts to estimate the number of partitions and the ranges of these partitions to determine the cut-points. Respondent identity (name), email addresses, domain expertise (DS/ML expert or non-expert), the number of partitions observed, and the ranges of each partition were collected together and compiled after the experiments.

5 Analysis and Result

Two of the fifty responses submitted via the Google form were incomplete, therefore, they are not included in the analysis. We classified expert and non-expert responses from the remaining forty-eight responses into two categories. Table 1 illustrates the comparison of human perception to identify the number of partitions between the DS/ML experts' responses and non-expert people. We received 60% responses from DS/ML experts and 40% of answers from non-expert people.

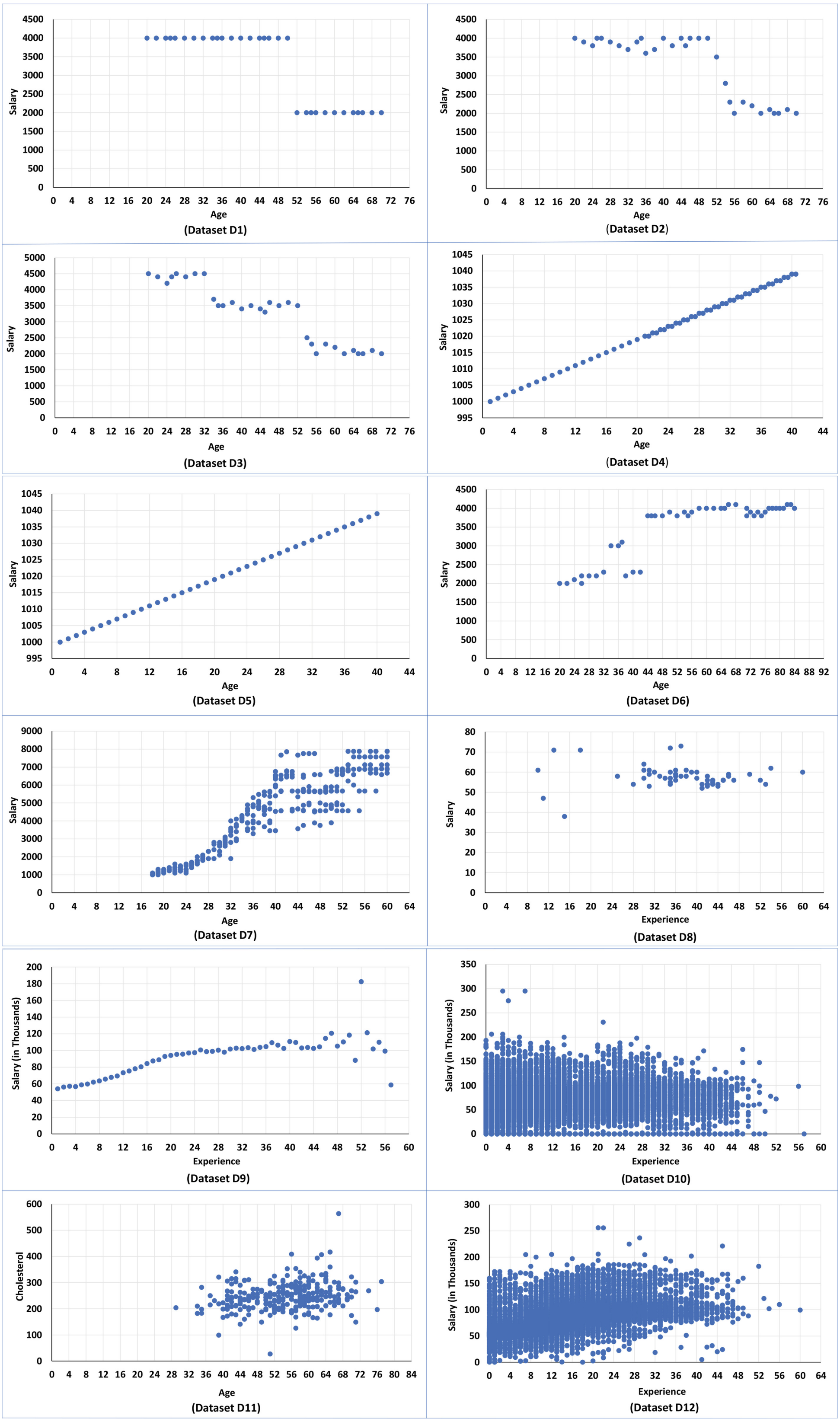


Fig. 1. Graphs for datasets D1 to D12.

5.1 Step-Function

In a *Step function f*, the domain is partitioned into several intervals. *f(x)* is constant for each interval, but the constant can be different for each interval. The different constant values for each interval create the jumps between horizontal line segments and develop a staircase, which is also known as a step function. The datasets D1, D2 and D3, are examples of step functions. We can also include D6 for the example of the step function. For datasets, D1 and D2, most responses from both categories (experts 93.3%, 73.3% and non-experts 90%, 60%) were for two partitions, and for dataset D3, three partitions were identified by contributors (expert 93.3% and non-experts 100%). The dataset D6 received maximum responses for four and five partitions, which indicates D6 as a step function. Humans identified partitions based on dense regions of data points. As for datasets D1 and D2, two dense regions were identified. However, for D3 and D6, three and four dense groups were identified, respectively. Hypothesis H1 confirms for datasets D1, D2, D3 and D6.

5.2 Linear Function

A linear function is a straight line between one independent and one dependent variable. The datasets D4 and D5 are examples of linear functions. The dataset D4 has more dense data points on another side of the slope and received a total of 60% of responses (see Fig. 2) for two partitions, which means contributors split the data points based on the mean of the function and identified two partitions. However, in contrast, dataset D5 did not receive any responses for two partitions and got a total of 92% responses for no partition. We argue that there is insufficient ground for selecting cut-points when a continuous variable is distributed uniformly in the environment, so splitting would be pretty random. Hence, contributors did not identify any partition for dataset D5. The hypothesis H3 confirms for dataset D4 but contradicts for D5. Here we can notice that H1 is also true for D4 and partially true for D5.

Table 1. The comparison of human perception to identify number of partitions based on their profile.

Resp.	P	Datasets											
		D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12
DS/ML Experts (60%)	No				33.3%	93.3%					40%	53.3%	40%
	2	93.3%	73.3%	0%	53%	0%	13.3%	60%	33.3%	73.3%	40%	26.6%	26.6%
	3	6.67%	26.6%	93.3%	13.3%	6.6%	26.6%	20%	66.6%	26.6%	6.66%	20%	6.66%
	4			6.66%			26.6%	20%	0%	0%	6.66%	0%	20%
	5						33.3%				6.66%		6.66%
Non-experts (40%)	No				20%	90%					40%	60%	20%
	2	90%	60%	0%	70%	0%	30%	40%	60%	60%	30%	30%	30%
	3	10%	40%	100%	10%	10%	0%	40%	30%	30%	20%	0%	30%
	4			0%			40%	20%	10%	10%	0%	10%	10%
	5						30%				10%		10%

Resp: Responders; P: number of partitions

5.3 Uneven Random Function

The datasets D7 to D12 fall under the uneven, randomly scattered plot category. We found that responses from both categories were opposite for graph D8. Out of the total responses for D8, 33.3% responses of DS/ML experts marked two partitions and 66.6% responses of experts marked three partitions; however, 60% of non-experts marked two partitions, and only 30% marked three partitions. Overall, 52% responses favor three partitions, and 44% of responses identify two partitions. In this case, experts include the scattered data points and consider them as one partition, and the remaining dense data points are identified as two more partitions. However, non-experts observed two partitions, one with a dense and the other with a scattered group of data points. The same situation occurs with dataset D9; here, a total of 68% of responses identified two partitions, one with the dense data points and the second with the scattered data points. Hypothesis H1 is also confirmed by the datasets D8 and D9. Datasets D10, D11, and D12 received high responses for no partition compared to other partitions. For the dataset D10, 40% contributors responded with no partition, and for the rest of the contributors, some random cut-points were marked for two, three, four, and five partitions. Similarly, for the dataset D11, 56% responses are favored for no partition, and the rest of the responses are answered for two, three, and four partitions. The dataset D12 encountered the same situation where 32% contributors responded with no partition, and 68% contributors marked some random cut-points for two, three, four, and five partitions. For these cases, hypothesis H4 confirms, but H2 contradicts, as outliers do not influence human response (case of datasets D8, D9, and D11). Hence, it proves that humans have no clear perception of these types of datasets and they are unable to identify cut-points.

The percentage of responses of each partition for each dataset is demonstrated in Fig. 2. Datasets D5, D10, D11 and D12 received high responses for

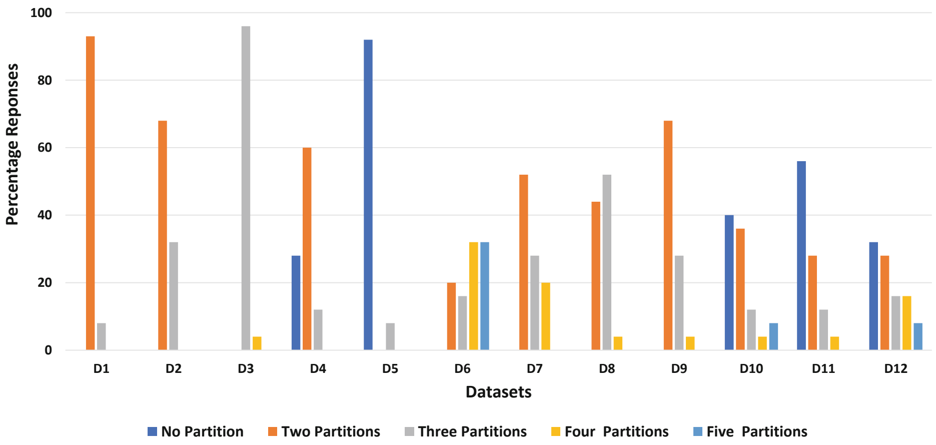


Fig. 2. Percentage responses of partitions for each dataset.

no partition compared to other partitions. Hence, it proves that humans have no clear perception of these types of datasets, and they are unable to identify cut-points. It is important to note that datasets D5 and D6 have similar appearances, but both datasets received different cut-points responses because of the distribution of their data points. D6 has not received any responses with no partition, and D5 has not gotten any responses with two partitions. For the datasets D1, D2, D4, D7 and D9, mainly two partitions were suggested by contributors. We find that the density of partitions has a substantial impact on perception during a visual interpretation. The random distribution of data points and linear function also influence human perception. However, outliers do not affect human judgement. After analyzing Table 1, we also reach the conclusion that the opinions of experts and non-expert responders do not make a huge difference, except in some situations.

6 Conclusion

The main objective of this research is to analyze the human perception of partitioning the numerical attribute. In this paper, we analyzed the perception of DS/ML experts and non-experts by providing them a series of graphs with numerical data. The analysis gives us insights that the perceptions of experts and non-experts while partitioning the numerical attribute are not much different. However, the data points' features influence most of the outcomes. Therefore, human judgment plays a vital role in developing an automated approach for partitioning numerical attributes with the best cut points. In future work, we plan to assess the accuracy of our proposed measures by comparing the outcomes of human perceptions.

References

1. Aupetit, M., Sedlmair, M., Abbas, M.M., Baggag, A., Bensmail, H.: Toward perception-based evaluation of clustering techniques for visual analytics. In: Proceedings of VIS2019 - IEEE Visualization Conference, pp. 141–145 (2019)
2. Demiralp, Ç., Bernstein, M.S., Heer, J.: Learning perceptual kernels for visualization design. IEEE Trans. Visual Comput. Graph. **20**(12), 1933–1942 (2014)
3. Draheim, D.: Generalized Jeffrey conditionalization: a frequentist semantics of partial conditionalization. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-69868-7
4. Draheim, D.: Future perspectives of association rule mining based on partial conditionalization. In: Proceedings of DEXA'2019 - the 30th International Conference on Database and Expert Systems Applications, LNCS, vol. 11706, p. xvi. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-27615-7
5. Etemadpour, R., da Motta, R.C., de Souza Paiva, J.G., Minghim, R., de Oliveira, M.C.F., Linsen, L.: Role of human perception in cluster-based visual analysis of multidimensional data projections. In: Proceedings of IVAPP -International Conference on Information Visualization Theory and Applications, pp. 276–283 (2014)

6. Garcia, S., Luengo, J., Sáez, J.A., Lopez, V., Herrera, F.: A survey of discretization techniques: taxonomy and empirical analysis in supervised learning. IEEE Trans. Knowl. Data Eng. **25**(4), 734–750 (2012)
7. Janosi, A., Steinbrunn, W., Pfisterer, M., Detrano, R.: Heart Disease. UCI machine learning repository (1988)
8. Kalish, M.: DC public employee salaries (2011). https://data.world/codefordc/dc-public-employee-salaries-2011
9. Kaushik, M.: Datasets (2022). https://github.com/minakshikaushik/LSQM-measure.git
10. Kaushik, M., Sharma, R., Peious, S.A., Draheim, D.: Impact-Driven Discretization of Numerical Factors: Case of Two- and Three-Partitioning. In: Srirama, S.N., Lin, J.C.-W., Bhatnagar, R., Agarwal, S., Reddy, P.K. (eds.) BDA 2021. LNCS, vol. 13147, pp. 244–260. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-93620-4_18
11. Kaushik, M., et al.: A systematic assessment of numerical association rule mining methods. SN Comput. Sci. **2**(5), 1–13 (2021)
12. Kotsiantis, S., Kanellopoulos, D.: Discretization techniques: a recent survey. GESTS Int. Trans. Comput. Sci. Eng. **32**(1), 47–58 (2006)
13. Liu, H., Hussain, F., Tan, C.L., Dash, M.: Discretization: An enabling technique. Data Min. Knowl. Disc. **6**(4), 393–423 (2002)
14. Naik, S.: NJ teacher salaries. (2016). https://data.world/sheilnaik/nj-teacher-salaries-2016
15. Arakkal Peious, S., Sharma, R., Kaushik, M., Shah, S.A., Yahia, S.B.: Grand reports: a tool for generalizing association rule mining to numeric target values. In: Song, M., Song, I.-Y., Kotsis, G., Tjoa, A.M., Khalil, I. (eds.) DaWaK 2020. LNCS, vol. 12393, pp. 28–37. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-59065-9_3
16. Shahin, M., et al.: Big data analytics in association rule mining: A systematic literature review. In: Proceedings of BDET 2021- International Conference on Big Data Engineering and Technology, pp. 40–49. ACM (2021)
17. Sharma, R., et al.: A novel framework for unification of association rule mining, online analytical processing and statistical reasoning. IEEE Access **10**, 12792–12813 (2022). https://doi.org/10.1109/ACCESS.2022.3142537
18. Sharma, R., Kaushik, M., Peious, S.A., Shahin, M., Yadav, A.S., Draheim, D.: Towards unification of statistical reasoning, OLAP and association rule mining: semantics and pragmatics. In: Database Systems for Advanced Applications. DASFAA 2022, LNCS, vol. 13245. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-00123-9_48
19. Sharma, R., Kaushik, M., Peious, S.A., Yahia, S.B., Draheim, D.: Expected vs. unexpected: selecting right measures of interestingness. In: Song, M., Song, I.-Y., Kotsis, G., Tjoa, A.M., Khalil, I. (eds.) DaWaK 2020. LNCS, vol. 12393, pp. 38–47. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-59065-9_4
20. Srikant, R., Agrawal, R.: Mining quantitative association rules in large relational tables. In: Proceedings of ACM SIGMOD 1996 - International Conference on Management of Data, pp. 1–12 (1996)

Appendix 5

[V]

M. Kaushik, R. Sharma, A. Vidyarthi, and D. Draheim. Discretizing numerical attributes: An analysis of human perceptions. In *Proceedings of ADBIS 2022 – the 26th International Conference on New Trends in Database and Information Systems*, pages 188–197, Cham, 2022. Springer International Publishing

Discretizing Numerical Attributes: An Analysis of Human Perceptions

Minakshi Kaushik[1]( ), Rahul Sharma[1], Ankit Vidyarthi[2], and Dirk Draheim[1]

[1] Information Systems Group, Tallinn University of Technology, Akadeemia tee 15a, 12618 Tallinn, Estonia
{minakshi.kaushik,rahul.sharma,dirk.draheim}@taltech.ee
[2] Jaypee Institute of Information Technology, Noida, India

Abstract. To partition numerical attributes, machine learning (ML) has used a variety of discretization approaches that partition the numerical attribute into intervals. However, an effective method for discretization is still missing in various ML approaches, e.g., association rule mining. Moreover, the existing discretization techniques do not reflect best the impact of the independent numerical factor on the dependent numerical target factor. The main objective of this research is to develop a benchmark approach for partitioning numerical factors. We present an in-depth analysis of human perceptions of partitioning a numerical factor and compare it with one of our proposed measures. We also examine the perceptions of various experts in data science, statistics and engineering disciplines by using a series of graphs with numerical data. The analysis of the collected responses indicates that 68.7% of the human responses were approximately close to the values obtained by the proposed method. Based on this analysis, the proposed method may be used as one of the methods for discretizing the numerical attributes.

Keywords: Machine learning · Data mining · Discretization · Numerical attributes · Partitioning

1 Introduction

Various types of variables are available in real-world data. However, discrete values have explicit roles in statistics, machine learning, and data mining. Presently, there is no benchmark approach to find the optimum partitions for discretizing complex real-world datasets. Generally, if a factor impacts another factor, in that case, humans can easily perceive the compartments or partitions because the human brain can easily perceive the differences between the factors and detect the partitions. However, it is not easy for a human or even an expert to find the appropriate compartments in complex real-world datasets.

This work has been partially conducted in the project "ICT programme" which was supported by the European Union through the European Social Fund.

S. Chiusano et al. (Eds.): ADBIS 2022, CCIS 1652, pp. 188–197, 2022.
https://doi.org/10.1007/978-3-031-15743-1_18

Existing discretization techniques do not reflect best the impact of the independent numerical factor on the dependent numerical target factor. Moreover, no discretization approach uses numerical attributes as influencing and response factors. To find the cut-points for the cases of two-partitioning and three-partitioning, we have proposed two measures *Least Squared Ordinate-Directed Impact Measure* (LSQM) and *Least Absolute-Difference Ordinate-Directed Impact Measure* (LADM) [10]. These measures provide a simple way to find partitions of numerical attributes that reflect best the impact of one independent numerical attribute on a dependent numerical attribute.

In this paper, the outcome of *LSQM* measure is compared with the human perceived cut-points to assess the accuracy of the measure. We use numerical attributes as influencing and response factors to distinguish them from the existing approaches. A series of graphs with different data points are used to collect the human responses. Here, data scientists, machine learning experts and other non-expert persons are referred to as humans.

The idea of this research emerged from the research on partial conditionalization [5,6], association rule mining (ARM) [17,19] and numerical association rule mining (NARM) [11,12,20]. These papers discuss the discretization process as an essential step for NARM. Moreover, research on discretizing the numerical attributes is an essential step in frequent itemset mining, especially for quantitative association rule mining [20].

In the same sequence, we have also presented a tool named Grand report [16] and a framework [18] for unifying ARM, statistical reasoning, and online analytical processing. These paper strengthens the generalization of ARM by finding the partitions of numerical attributes that reflect best the impact of one independent numerical attribute on a dependent numerical attribute. Our vision is to develop an ecosystem to generalize the machine learning approaches by significantly improving the ARM from different dimensions.

The paper is organized as follows. In Sect. 2, we discuss related work. In Sect. 3, we explain the motivation for conducting this study. Section 4 describes the *LSQM* method. Then we discuss the design of the experiment in Sect. 5. In Sect. 6, analysis and results are given. The conclusion and future work are given in Sect. 7.

2 Related Work

Based on human perception evaluation and different discretization techniques, we discuss the related work in the direction of discretization, clustering techniques and human perception.

A variety of discretization methods are available in the literature [9,13,14]. Dougherty et al. [4] compared and analyzed discretization strategies along three dimensions: global versus local, supervised versus unsupervised, and static versus dynamic. Liu et al. [14] performed a systematic study of existing discretization methods and proposed a hierarchical framework for discretization methods from the perspective of splitting and merging. The unsupervised static discretization method, such as equal-width, uses the minimum and maximum values of the continuous attribute and then divides the range into equal-width intervals called bins. In contrast, the equal-frequency algorithm determines an equal number of continuous values and places them in each bin [2].

In state of the art, many studies have used human perception to evaluate the various techniques. However, they are not completely related to discretization. Etemadpour et al. [7] conducted a perception-based evaluation of high-dimensional data where humans were asked to identify clusters and analyze distances inside and across clusters. Demiralp et al. [3] used human judgments to estimate perceptual kernels for visual encoding variables such as shape, size, colour, and combinations. The experiment used Amazon's Mechanical Turk platform, with twenty Turkers completing thirty MTurk jobs. In [1] authors evaluated benchmarking clustering algorithms based on human perception of clusters in 2D scatter plots. The authors' main concern was how well existing clustering algorithms corresponded to human perceptions of clusters. Our work is also related to considering human perceptions for evaluating our proposed *LSQM* measure for discretizing numerical attributes.

3 Motivation

For years, obtaining discrete values from numerical values has been a complex and ongoing task. The main issue with the discretization process is obtaining the perfect intervals with specific ranges and numbers of intervals. In the state of the art, several discretization approaches such as equi-depth, equi-width [2], MDLP [8], Chi2 [15], D2 [2], etc. have been proposed. However, determining the most effective discretizer for each situation is still a challenging problem.

In [10], we presented an order-preserving partitioning method to find the partitions of numerical attributes that reflect best the impact of one independent numerical attribute on a dependent numerical attribute. In extreme cases (such as step-functions), humans can easily visualize the perfect partitions and even the number of compartments. However, in distinct cases, the ideal partition range depends on the perception of data experts. In state of the art, no investigation is available to understand the human perception of partitioning. Moreover, the current literature provides a comparison of discretization methods and compares their results. In this paper, we take a different approach to compare the human perception of discretization with the outcome of the proposed discretization method. We aim to visualize the differences between the outcomes of the proposed methods and the human perception of discretization.

4 The LSQM Method

In the *LSQM* method [10], we discretize the independent numerical attribute on the basis of order-preserving partitioning to learn the impact on the numerical target attribute. The number of cut-points is $k - 1$, where k is the number of partitions suggested by the user. The measure first calculates the squared difference between the y-value of each data point and the average of y-values of the current partition. The order of the independent variable is preserved using the value of data points. Therefore, the value of data points of one partition will always be less than the value of data points of the next partition. After summing up the squared differences of the several partitions, *LSQM* retrieves the minimum values as cut-points.

Definition 1 (Least Squared Ordinate-Directed Impact Measure). *Given real-valued data points* $(<x_i, y_i>)_{2\leq i\leq n}$*, we define the* least squared ordinate-directed impact measure *for* k*-partitions as follows:*

$$\underset{i_0=0<i'_1<\ldots<i'_{k-1}<i'_k=n}{min} \sum_{j\,=\,1}^{k} \sum_{i'_{j-1}<i"\leq i'_j} (y_{i"} - \mu_{i'_{j-1}<\phi\leq i'_j})^2 \tag{1}$$

where the average of data values in a partition $\mu_{a<\phi\leq b}$ *between indexes* a *and* b *(*$a < b \leq n$*) is defined as*

$$\mu_{a<\phi\leq b} = \frac{\sum\limits_{a<\phi\leq b} y_\phi}{b - a} \tag{2}$$

In (1), we have that i'_j is the highest element in the j-*th* partition, where *highest element* means the data point with the highest index.

The definition of the *LSQM* measure seems similar to the k-means clustering algorithm. The k-means clustering algorithm is a partitioning clustering algorithm to classify objects into k different clusters. The *LSQM* measure is different from the k-means algorithm as k-means is based on the Euclidean distance metric between two vectors, X and Y. It also has the severe drawback that its efficiency is highly dependent on the initial random selection of cluster centres. However, the *LSQM* measure is based on order-preserving partitioning for the independent variable. This measure also does not depend on the initial point chosen for starting.

5 Experimental Design

To understand how humans partition numerical factors, we designed a series of graphs and asked several experts to partition the data points given in the graphs. Initially, to produce a diverse collection of graphs with different data points, a set of graphs was shared and discussed with our own research team. These graphs include step functions, linear functions, and mixed data graphs. Finally, eight graphs were selected to be shared with humans (see Fig. 1). These graphs are obtained from eight synthetic datasets (D1 to D8). These synthetic datasets (D1 to D8) consist only two numerical attributes. A copy of all these datasets is available in the GitHub repository[1].

[1] https://github.com/minakshikaushik/LSQM-measure.git.

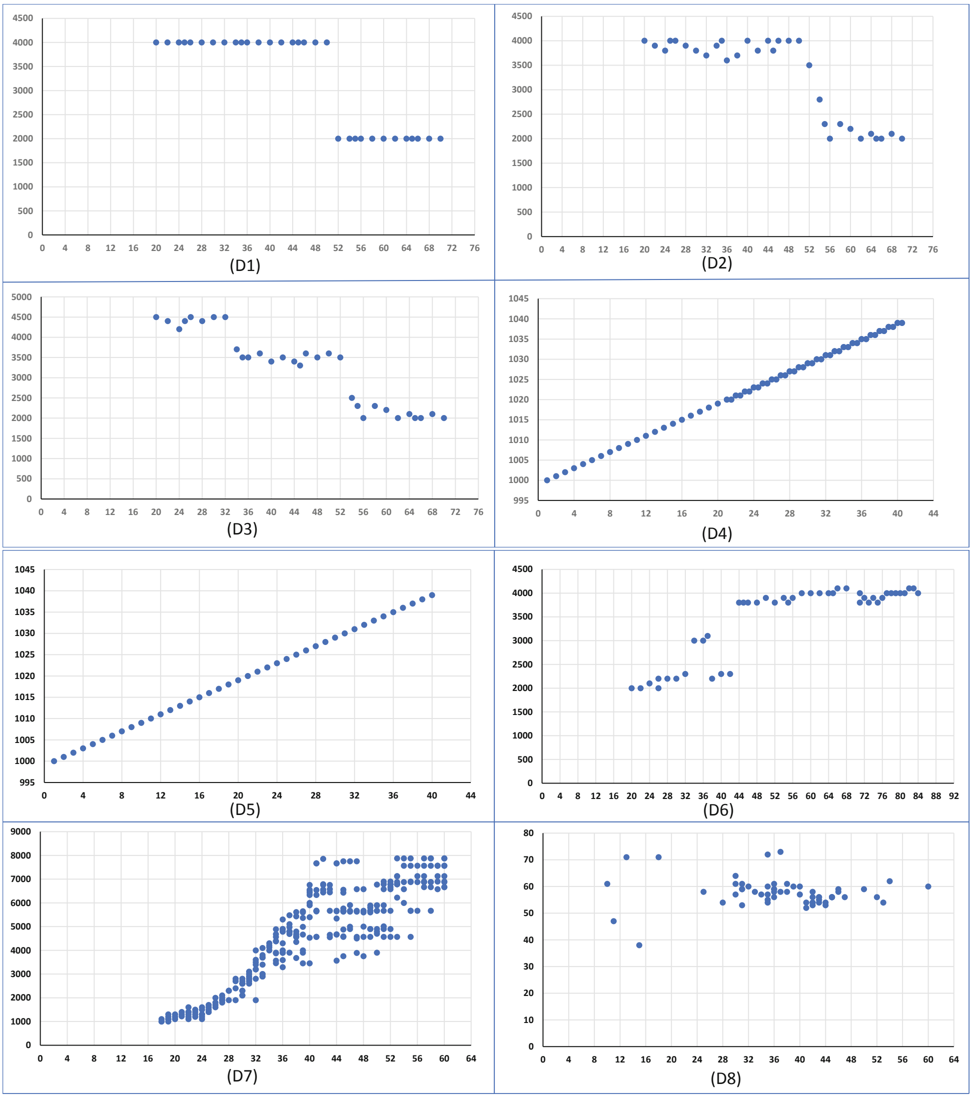


Fig. 1. Graphs for datasets D1 to D8.

We designed a Google form by providing a series of graphs containing different types of numerical data points and relevant questions to collect human responses and their perceptions about discretization. The google form was sent to fifty DS/ML experts and non-experts to estimate the number of partitions and the ranges of these partitions to obtain the cut-points.

Table 1. The comparison of human perception to identify the number of partitions based on their profile.

Responders	Partitions	Datasets							
		D1	D2	D3	D4	D5	D6	D7	D8
DS/ML-Experts (60%)	No				33.3%	93.3%			
	2	93.3%	73.3%	0%	53%	0%	13.3%	60%	33.3%
	3	6.67%	26.6%	93.3%	13.3%	6.6%	26.6%	20%	66.6%
	4			6.66%			26.6%	20%	0%
	5						33.3%		
Non-experts (40%)	No				20%	90%			
	2	90%	60%	0%	70%	0%	30%	40%	60%
	3	10%	40%	100%	10%	10%	0%	40%	30%
	4			0%			40%	20%	10%
	5						30%		

Table 2. The comparison of human perceived cut-points with the LSQM measure.

D	P	Human perception		LSQM
		R	Approx. near cut-points	Cut-points
D1	2	92%	50(91.3%), 48(8.6%)	50
	3	8%	(48,60)(50%), (20,50)(50%)	(20, 50)
D2	2	68%	50(88.2%), 52(11.7%)	52
	3	32%	(50,54)(37.5%), (20,53)(25%)	(52, 54)
D3	3	96%	(32,52)(62%), (30,52)(16.6%)	32,52
	4	4%	(20,32,52)(100%)	(32,52,55)
D4	0	28%	NA	NA
	2	60%	20(86.6%), 25(13.3%)	20
	3	12%	(20,45)(66.6%), (20,30)(33.3%)	(12, 24)
D5	0	92%	NA	NA
	2	0%	NA	20
	3	8%	(14,28)(100%)	(13, 26)
D6	2	20%	32(40%), 42(40%) 50(20%)	42
	3	16%	(42,68)(50%), (32,42)(25%)	(32, 42)
	4	32%	(32,37,42)(87.5%), (33,37,43)(12.5%)	(32, 37, 42)
	5	32%	(32,42,37,68)(87.5%), (17,32,38,42)(12.5%)	(32, 37, 42, 56)
D7	2	52%	40(84.6%), 50(7.6%), 36(7.6%)	35
	3	28%	(32,39)(57.1%)	(32, 39)
	4	20%	(32,39,50)(60%), (41,47,53)(40%)	(32,39,52)
D8	2	44%	18(36%), 30(27%)	40
	3	52%	(28,47)(53.8%), (18,47)(23%)	(13, 15)
	4	4%	(18,47,54)(100%)	(11, 13, 15)

D: Datasets; P: number of partitions; R: percentage of responses

The following data was gathered and compiled from the experiments: respondent identification (name), their email addresses, domain expertise (DS/ML expert or non-expert), number of partitions identified, and ranges of each partition.

6 Analysis and Result

Out of the fifty responses received via the Google form, two were incomplete; therefore, we did not consider them for the analysis. From the rest of the forty-eight responses, we divided the responses into two categories, expert responses and non-expert responses.

Table 3. Similarity between human perceived cut-points and LSQM cut-points.

	P	Datasets							
		D1	D2	D3	D4	D5	D6	D7	D8
LSQM	2	50	52		20		42	35	40
	3	(20,50)	(52,54)	(32,52)	(12,24)	(13,26)	(32,42)	(32,39)	(13,15)
	4			(32,52,55)			(32,37,42)	(32,39,52)	(11,13,15)
	5						(32,37,42,56)		
Human percep.	2	50	52		20		42	36	30
	3	(20,50)	(50,54)	(32,52)	(20,30)	(13,26)	(32,42)	(32,39)	(18,47)
	4			(20,32,52)			(32,37,42)	(32,39,52)	(18,47,54)
	5						(32,37,42,68)		
Matching%	2	91.3%	11.7%		80.6%		40%	0%	0%
	3	50%	19%	62%	0%	100%	25%	57%	0%
	4			59%			85.7%	60%	0%
	5						75%		
Matching Status	2	VH	L		VH		M	NM	NM
	3	M	L	H	NM	VH	L	H	NM
	4			M			VH	H	NM
	5						H		

P: Number of partitions, VH: 80–100%, H: 60–80%, M: 40–60%, L: 1–40%, NM: 0%

Table 1 illustrates the comparison of human perception to identify the number of partitions between the DS/ML experts' responses and non-expert people. We received 60% responses from DS/ML experts and 40% of answers from non-expert people. We analyzed that responses from both categories were opposite for graph D8. Out of the total responses for D8, 33.3% responses of DS/ML experts marked two partitions and 66.6% responses of experts marked three partitions; however, 60% of non-experts marked two partitions, and only 30% marked three partitions. In graphs D3 and D5, we analyzed that no contributor (experts or non-experts) marked two partitions. No non-expert contributors marked three partitions for graph D6 and four partitions for graph D3; whereas 26.6% of DS/ML experts identified three partitions for D6, and 6.66% experts marked

Table 4. Analysis of unmatched datasets in regard of number of partitions for the LSQM and human perceived cut-points.

Dataset	Partitions	LSQM method		Human perception		Remarks
		LSQM cut-points	Logical correctness	Human perceived cut-points	Logical correctness	
D8	2	40	Yes	30	Yes	Matter of perception
	3	(13,15)	No	(18,47)	Yes	LSQM to be improved
	4	(11,13,15)	No	(18,47,54)	Yes	LSQM to be improved
D4	3	(12,24)	Yes	(20,30)	Yes	Matter of perception
D7	2	35	Yes	36	Yes	Matter of perception

four partitions in the graph D3. Table 2 illustrates the comparison between the results of human perception and the *LSQM* measure. Table 3 describes the similarity percentage between cut-points provided by human perceived experiment outcome and the *LSQM* measure outputs. We have mentioned the cut-points from responses near the *LSQM* provided cut-points. We determine the matching status by distributing the matching percentage into the following categories: VH (Very High), H (High), M (Medium), L (Low) and NM (No match). The distribution of ranges is mentioned at the bottom of Table 3. It is clear from Table 3 that human perceived cut-points and the cut-points identified by the proposed measure *LSQM* do not match for the datasets D8, D4 and D7. In Table 4, we present an analysis and reason for not getting similar cut-points for the datasets D4, D8 and D7. If we look at Fig. 1(D8), then it seems logical to have cut-points at the data points of 40 (LSQM cut-point) and 30 (Human perceived cut-point) for two partitions on the X-axis. Humans divided the scattered points into first partition and dense data points into the second partition. In contrast, the *LSQM* measure calculated the cut-point in the middle of the dense data points. This case can be observed as a matter of perception for human perceived cut-points, while the cut-points marked by the LSQM measure seem analytically correct. For the cases of three partitions and four partitions, human perceived cut-points $(18, 47)$ and $(18, 47, 54)$ are good, but the cut-points provided by the *LSQM*measure are not satisfactory. The cut-points provided by the *LSQM* $(12, 24)$ and human perception experiment $(20, 30)$ for D4 are also the case of matter of perception. Similarly, cut-points 35 and 36 for D7 do not match exactly. However, as the data points in the graph are scattered; therefore, the difference between the cut-points of the proposed measure and the human perceived cut-points is negligible and both can be considered the best cut-points. This case can be observed as a matter of perception. Although these cut-points do not match the *LSQM* measure cut-points, the correctness of the measure is not affected due to non-similarity.

Out of the total responses for D1 to D7, we analyzed that 25% responses were matching *Very High*, 25% responses were matching *High*, 18.7% responses were matching *Medium* and 18.7% responses were matching *Low*. By aggregating all the matching status, 68.7% responses were similar to the responses marked by the proposed *LSQM* measure. By the overall analysis, it is clear that for initial datasets (D1 to D7), the proposed measure brought approximately equivalent results to human perception. The analysis is conducted for the datasets D1 to D7 because some random cut-points were observed by the human for the dataset D8 which are difficult to match with the analytically calculated cut-points by the *LSQM*. An analysis and reason for not getting similar cut-points for the dataset D8 are given in Table 4.

7 Conclusion

This paper is the first step toward understanding the human perception of partitioning numerical attributes. We first assessed the human perception of partitioning numerical attributes by examining a series of graphs with numerical data. Furthermore, we compared the human perceived cut-points of partition with the results of the proposed *LSQM* measure. The proposed measure produces cut-points mostly close to human perceived cut-points. The overall analysis shows that the proposed measure produced results that were approximately equivalent to human perception for the datasets (D1 to D7). The present results of the proposed measure are encouraging, and it is a significant step towards the generalization of ARM by finding the partitions of numerical attributes that reflect best the impact of one independent numerical attribute on a dependent numerical attribute. In future work, we plan to implement with *inter*-measures for comparing partitions of different numbers of k-partitions.

References

1. Aupetit, M., Sedlmair, M., Abbas, M.M., Baggag, A., Bensmail, H.: Toward perception-based evaluation of clustering techniques for visual analytics. In: IEEE Visualization Conference on Proceedings of the VIS 2019, pp. 141–145 (2019)
2. Catlett, J.: On changing continuous attributes into ordered discrete attributes. In: Kodratoff, Y. (ed.) EWSL 1991. LNCS, vol. 482, pp. 164–178. Springer, Heidelberg (1991). https://doi.org/10.1007/BFb0017012
3. Demiralp, Ç., Bernstein, M.S., Heer, J.: Learning perceptual kernels for visualization design. IEEE Trans. Vis. Comput. Graph. **20**(12), 1933–1942 (2014)
4. Dougherty, J., Kohavi, R., Sahami, M.: Supervised and unsupervised discretization of continuous features. In: Machine Learning Proceedings 1995, pp. 194–202. Elsevier (1995)
5. Draheim, D.: Generalized Jeffrey Conditionalization: A Frequentist Semantics of Partial Conditionalization. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-69868-7
6. Draheim, D.: Future perspectives of association rule mining based on partial conditionalization. In: The 30th International Conference on Database and Expert Systems Applications, Proceedings of the DEXA 2019. LNCS, vol. 11706, p. xvi. Springer, Heidelberg (2019). https://doi.org/10.13140/RG.2.2.17763.48163

7. Etemadpour, R., da Motta, R.C., de Souza Paiva, J.G., Minghim, R., de Oliveira, M.C.F., Linsen, L.: Role of human perception in cluster-based visual analysis of multidimensional data projections. In: International Conference on Information Visualization Theory and Applications, Proceedings of IVAPP, pp. 276–283 (2014)
8. Fayyad, U., Irani, K.B.: Multi-interval discretization of continuous valued attributes for classification learning, 1993. In: The 13th International Joint Conference on Artificial Intelligence, Proceedings of IJCAI 1993 (1993)
9. Garcia, S., Luengo, J., Sáez, J.A., Lopez, V., Herrera, F.: A survey of discretization techniques: taxonomy and empirical analysis in supervised learning. IEEE Trans. Knowl. Data Eng. **25**(4), 734–750 (2012)
10. Kaushik, M., Sharma, R., Peious, S.A., Draheim, D.: Impact-driven discretization of numerical factors: case of two- and three-partitioning. In: Srirama, S.N., Lin, J.C.-W., Bhatnagar, R., Agarwal, S., Reddy, P.K. (eds.) BDA 2021. LNCS, vol. 13147, pp. 244–260. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-93620-4_18
11. Kaushik, M., Sharma, R., Peious, S.A., Shahin, M., Ben Yahia, S., Draheim, D.: On the potential of numerical association rule mining. In: Dang, T.K., Küng, J., Takizawa, M., Chung, T.M. (eds.) FDSE 2020. CCIS, vol. 1306, pp. 3–20. Springer, Singapore (2020). https://doi.org/10.1007/978-981-33-4370-2_1
12. Kaushik, M., Sharma, R., Peious, S.A., Shahin, M., Yahia, S.B., Draheim, D.: A systematic assessment of numerical association rule mining methods. SN Comput. Sci. **2**(5), 1–13 (2021)
13. Kotsiantis, S., Kanellopoulos, D.: Discretization techniques: a recent survey. GESTS Int. Trans. Comput. Sci. Eng. **32**(1), 47–58 (2006)
14. Liu, H., Hussain, F., Tan, C.L., Dash, M.: Discretization: an enabling technique. Data Min. Knowl. Disc. **6**(4), 393–423 (2002)
15. Liu, H., Setiono, R.: Feature selection via discretization. IEEE Trans. Knowl. Data Eng. **9**(4), 642–645 (1997)
16. Arakkal Peious, S., Sharma, R., Kaushik, M., Shah, S.A., Yahia, S.B.: Grand reports: a tool for generalizing association rule mining to numeric target values. In: Song, M., Song, I.-Y., Kotsis, G., Tjoa, A.M., Khalil, I. (eds.) DaWaK 2020. LNCS, vol. 12393, pp. 28–37. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-59065-9_3
17. Shahin, M., et al.: Big data analytics in association rule mining: a systematic literature review. In: International Conference on Big Data Engineering and Technology, Proceedings of the BDET 2021, pp. 40–49. ACM (2021)
18. Sharma, R., et al.: A novel framework for unification of association rule mining, online analytical processing and statistical reasoning. IEEE Access **10**, 12792–12813 (2022)
19. Sharma, R., Kaushik, M., Peious, S.A., Yahia, S.B., Draheim, D.: Expected vs. unexpected: selecting right measures of interestingness. In: Song, M., Song, I.-Y., Kotsis, G., Tjoa, A.M., Khalil, I. (eds.) DaWaK 2020. LNCS, vol. 12393, pp. 38–47. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-59065-9_4
20. Srikant, R., Agrawal, R.: Mining quantitative association rules in large relational tables. In: International Conference on Management of Data, Proceedings of the ACM SIGMOD 1996, pp. 1–12 (1996)

Appendix 6

[VI]

M. Kaushik, R. Sharma, I. Fister Jr.2, and D. Draheim. Numerical association rule mining: A systematic literature review, arxiv, 2307.00662, 2023

Numerical Association Rule Mining: A Systematic Literature Review

MINAKSHI KAUSHIK and RAHUL SHARMA, Tallinn University of Technology
IZTOK FISTER JR., University of Maribor
DIRK DRAHEIM, Tallinn University of Technology

Numerical association rule mining (NARM) is a widely used variant of the association rule mining (ARM) technique, and it has been extensively used in discovering patterns in numerical data. Initially, researchers and scientists incorporated numerical attributes in ARM using various discretization approaches; however, over time, a plethora of alternative methods have emerged in this field. Unfortunately, the increase of alternative methods has resulted into a significant knowledge gap in understanding diverse techniques employed in NARM – this paper attempts to bridge this knowledge gap by conducting a comprehensive systematic literature review (SLR). We provide an in-depth study of diverse methods, algorithms, metrics, and datasets derived from 1,140 scholarly articles published from the inception of NARM in the year 1996 to 2022. Out of them, 68 articles are extensively reviewed in accordance with inclusion, exclusion, and quality criteria. To the best of our knowledge, this SLR is the first of its kind to provide an exhaustive analysis of the current literature and previous surveys on NARM. The paper discusses important research issues, the current status, and the future possibilities of NARM. On the basis of this SLR, the article also presents a novel discretization measure that contributes by providing a partitioning of numerical data that meets well human perception of partitions.

CCS Concepts: • **Information systems** → **Association rules**; **Data mining**; • **Computing methodologies** → *Bio-inspired approaches*; Machine learning; • **General and reference** → *Surveys and overviews.*

Additional Key Words and Phrases: numerical association rule mining, systematic literature review, quantitative association rule mining

ACM Reference Format:
Minakshi Kaushik, Rahul Sharma, Iztok Fister Jr., and Dirk Draheim. 2023. Numerical Association Rule Mining: A Systematic Literature Review. 1, 1 (July 2023), 50 pages.

1 INTRODUCTION

Decision-makers have used a wide variety of data mining techniques to extract valuable insights from data. Out of these techniques, association rule mining (ARM) is one of the established data mining techniques. ARM was first proposed by R. Agrawal [4], and it is primarily used to identify interesting relationships between various data items, e.g., market basket analysis. Later, it has also been used in medical diagnosis and bioinformatics.

In the original settings of ARM, classical algorithms such as Apriori [5], Eclat [112] and FP-growth [43] were limited to work with boolean datasets only and do not support numerical data items like height, weight, or age. To extend the scope of ARM to support numerical items, Srinkant et al.

Authors' addresses: Minakshi Kaushik, minakshi.kaushik@taltech.ee; Rahul Sharma, rahul.sharma@taltech.ee, Tallinn University of Technology, Akadeemia tee 15a, 12618, Tallinn, Estonia; Iztok Fister Jr. University of Maribor, Koroska cesta 46, SI-2000, Maribor, Slovenia, iztok.fister1@um.si; Dirk Draheim, dirk.draheim@taltech.ee, Tallinn University of Technology, Akadeemia tee 15a, 12618, Tallinn, Estonia.

XXXX-XXXX/2023/7-ART $15.00
https://doi.org/

[96] proposed a new technique “quantitative association rule mining (QARM)”. In this technique, numerical data items are converted to categorical data through a discretization process. In literature, QARM is also referred to as “numerical association rule mining (NARM)” [11].

In the early stages of research on NARM, researchers and scientists have used various discretization approaches. However, as time progressed, a wide range of alternative methods emerged, offering novel and innovative solutions in this field. Unfortunately, the increased number of alternative methods has created a substantial knowledge gap, making it difficult to fully comprehend the diverse range of techniques utilised in NARM.

To address this knowledge gap, this paper conducts a comprehensive systematic literature review (SLR) by following one of the established research methodologies for SLR as outlined by Kitchenham and Charters's [18]. Before conducting this SLR, we thoroughly reviewed several surveys and reviews on NARM, which are listed in Table 1. However, it is important to note that these existing surveys and reviews have certain limitations. They often lack well-defined research questions, comprehensive search strategies, and rigorous research methodologies. Notably, to the best of our knowledge, no SLR of the existing literature on NARM has been conducted to date. The absence of a systematic review in the field has highlighted the need for this article and inspired us to fill this knowledge gap. Indeed, the identified limitations in previous surveys and reviews raised the importance of conducting SLR on NARM. Through this SLR, we aim to address these limitations and fulfil the need for a more comprehensive understanding of the field.

In order to provide a complete overview of the NARM literature, we conducted a systematic search across various academic databases and digital libraries to identify relevant scholarly articles. We majorly focused on articles published from the inception of NARM in 1996 up until 2022. In total, we identified 1,140 articles that met our search queries. Next, as per the research methodology, we applied a rigorous process of inclusion, exclusion, and quality assessment criteria to ensure that the selected articles were relevant to the research domain and of high quality. After the screening process, we narrowed down the initial list to a final selection of 68 articles. By following this systematic approach, we aimed to gather a comprehensive and reliable set of articles that contributes to the thorough analysis and synthesis of existing knowledge on NARM.

Based on the exhaustive analysis of 68 articles, this SLR provides an in-depth examination of diverse methods, algorithms, metrics, and datasets utilised in NARM. We thoroughly evaluate the strengths and weaknesses of these methods, algorithms, and metrics while also highlighting their outcomes and potential applications. By conducting a comprehensive analysis of the available literature, we aim to provide deep insights and understanding that can benefit researchers, practitioners, and stakeholders in the field.

As per the findings of this SLR, the article also contributes by introducing an automated novel discretization measure that addresses the human perception of partitions, providing a meaningful and accurate partitioning of numerical data. This novel measure aims to overcome the limitations of existing methods by providing a more meaningful and accurate partitioning of numerical data.

The primary contributions of this paper are as follows.

- Well-defined research questions and a methodology for extracting data for a systematic investigation in the area of mining numerical association rules.
- Detailed knowledge about NARM methods and their algorithms.
- Identified popular metrics to evaluate NARM algorithms.
- Identified the major challenges involved in generating numerical association rules, along with some probable future perspectives.

- A novel automated measure is presented for discretizing numerical attributes to contribute to NARM by providing a partitioning of numerical data that meets well human perception of partitions.
- Fills the gaps and overcomes the limitations of previous surveys.

The article is organized as follows: Section 2 presents an overview of the background and related work. In Section 3, we detail our research methodology and articulate the research questions (RQs). Section 4 presents the findings of the review. In Section 5, we address the potential threats to the validity of this research article. Section 6 delves into a comprehensive discussion of the SLR's findings. Finally, we draw our conclusions in Section 7.

Table 1. Contributions and Limitations of Previous Reviews (N.M.= Not Mentioned)

Paper	Type	Time-frame	Methodology	Contributions	Limitations
This Article	SLR	1996-2022	Kitchenham's guideline [18]	Present detailed and systematic review encompassing various aspects of NARM, including methods, algorithms, and other relevant factors.	
Kaushik et al. (2021) [53]	Review	1996-2020	Undefined	Investigated discretization techniques in various NARM methods and assessed 30 NARM algorithms.	Authors focused only on important algorithms for three methods.
Adhikary et al. (2015) [2]	Survey	N.M.	Undefined	Authors presented clustering, partitioning, and fuzzy approaches, including evolutionary, statistical and info-theoretic approaches.	This study did not present a detailed study of the discretization method.
Adhikary et al. (2015) [1]	Review	N.M.	Undefined	Discussed applications of NARM.	Approaches were not discussed in detail.
Gosain et al. (2013) [39]	Survey	N.M	Undefined	Presented a comparative study of different approaches of ARM including Quantitative data.	Approaches were not categorized and randomly presented.

2 BACKGROUND AND RELATED WORK

In this section, we provide an in-depth explanation of the background of ARM and NARM.

2.1 Association Rule Mining

In the original setting, association rules are extracted from transactional datasets composed of a set $I = \{i_1, \ldots, i_n\}$ of n binary attributes called *items* and a set $D = \{t_1, \ldots, t_n\}$, $t_k \subseteq I$, of *transactions* called database. An *association rule* is a pair of itemsets (X, Y), often denoted by an implication of the form $X \Rightarrow Y$, where X is the antecedent (or premise), Y is the consequent (or conclusion) and $X \cap Y = \emptyset$. In ARM, support and confidence measures are widely utilized and considered fundamental metrics. The support of an itemset X determines how frequently the itemset appears in a transactional database. The support of an association rule $X \Rightarrow Y$ can be defined as the percentage of transactions among the total records that contain both itemsets X and Y, shown in Eq. 1.

The confidence of an association rule $X \Rightarrow Y$ determines how frequently items in Y appear in transactions that contain X. The confidence of a rule is calculated as the percentage of transactions that contain itemset X also contain itemset Y, to the total number of records that contain X shown in Eq. 2.

$$Support(X \Rightarrow Y) = \frac{|(X \cup Y)|}{|D|} \tag{1}$$

$$Confidence(X \Rightarrow Y) = \frac{|(X \cup Y)|}{|X|} \tag{2}$$

2.2 Numerical Association Rule Mining

NARM came into the scenario to extract association rules from numerical data. Unlike the classical ARM, numerical ARM allows attributes to be categorical (e.g., gender, education) or numeric (e.g., salary, age) rather than just Boolean. A numerical association rule is an implication of the form $X \Rightarrow Y$, in which both antecedent and consequent parts are the set of attributes in the forms $A = \{v_1, v_2, \dots v_n\}$ if A is a categorical attribute, or $A\epsilon\ [v_1, v_2]$ if A is numeric attribute.

An example of a numerical association rule is given below.

$$Age \in [21, 35] \wedge Gender : [Male] \Rightarrow Salary \in [2000, 3000]$$

$$(Support = 10\%, Confidence = 80\%)$$

This rule states that those employees who are males, aged between 21 and 35 and having salaries between \$2,000 and \$3,000 form 10% of all employees; and that 80% of males aged between 21 and 35 are earning between \$2,000 and \$3,000. Here, *Age* and *Salary* are numerical attributes and *Gender* is a categorical attribute. In ARM, except for support and confidence, more than fifty measures of interestingness are available in the literature [37, 94]. The support of an association rule $X \Rightarrow Y$ determines how frequently the itemset appears in a transactional database. The confidence of an association rule determines how many transactions that contain X also contain Y.

2.3 Related Work

In recent years, there have been few surveys and studies in the literature that have focused on NARM approaches and their comparison. However, no SLR has been published to date. Our automated search identified three reviews [2, 52, 53] and a manual search found two surveys [1, 39]. While these reviews provide a contribution towards understanding the methods and algorithms for NARM, they have several limitations, as outlined in Table 1. Gosain et al. [39] presented a survey of association rules on quantitative data in 2013. The authors focused on different types of association rules but did not include NARM methods and algorithms. Adhikary and Roy[1] reviewed QARM techniques, with a focus on applications in the real world, while their 2015 survey [2] provided a classification of QARM techniques but lacked valuable information. A systematic assessment of the three popular methods for NARM with thirty algorithms was conducted in [53]. This review focused only on NARM algorithms, and the steps of systematic reviews were not followed. In contrast, our study conducted an SLR under the guidelines of Kitchenham and Charters [18] and answered the research questions in the state of the art of NARM.

Moreover, it is worth noting that some authors have made notable contributions to NARM under alternative names. For example, Telkani et al. [103] conducted an extensive survey on evolutionary computation for ARM, wherein they thoroughly examined various approaches within the realm of ARM, including NARM, and provided insights into the the classification of evolutionary algorithms in this context.

3 RESEARCH METHODOLOGY

In this work, we adopt research methodology based on Kitchenham and Charters's guidelines [18]. The main goal of this SLR is to summarize the existing evidence in the literature regarding this topic and to identify gaps in the literature. According to Kitchenham's guidelines, the process included three main phases: planning, conducting, and reporting the review. The planning phase involved identifying the need for the review and establishing a review protocol. The conducting phase involved following the review protocol, which included selecting a primary search, assessing the quality of the studies, and extracting relevant data. Finally, the reporting phase focused on formatting and evaluating the report in accordance with the guidelines.

3.1 Planning the Review

The initial phase of this study aims to justify the need for an SLR and define the research questions. Based on the objective and motivation of this study, we formulated the following research questions with the goal presented in Table 2. The primary aim of this SLR is to address these research questions, which will help us to comprehensively understand the existing research and identify gaps in the literature related to NARM.

3.2 Conducting the Review

The review phase involves a series of sequential steps, beginning with the identification of relevant research and followed by the selection of studies, study quality assessment, and data extraction. These steps are conducted systematically to ensure the comprehensive coverage of relevant studies and the extraction of accurate and reliable information for analysis.

3.2.1 Search Strategy.

Academic Databases. To conduct the review phase, we conducted a thorough search of scientific publications from relevant journals and conferences, utilizing multiple reputable digital libraries, including the ACM Digital Library, Scopus, SpringerLink, IEEE Xplore, and ScienceDirect. Additionally, we performed a manual search on Google Scholar to minimize the chance of overlooking any significant articles. The search was conducted between April and June 2022, focusing on articles published in journals and conferences. We set the time frame for articles published from 1996 to 2022, as it was in 1996 when Srikant and Agrawal [96] first presented the problem statement concerning numerical attributes.

Search Strings. For the search process, we derived the search terms from the research questions and compiled a comprehensive list of synonyms, abbreviations, and alternative words. In this study, we have also mentioned that the problem of handling numerical attributes was initially addressed as “quantitative association rule” by Srikant and Agrawal [96]. Over time, this term evolved into “numerical association rules.” Therefore, to ensure inclusivity, our search terms included variations such as "quantitative association rule mining," OR "numerical association rule mining," OR "quantitative association rules," OR "numerical association rules," OR "quantitative ARM," OR "numerical ARM," OR "QARM," OR "NARM." We targeted these terms in the abstracts, titles, and keywords of articles within the following electronic sources.

Table 2. Research Questions Together With Their Goals

ID	Research Question	Goal
RQ1	Which methods exist for solving NARM problems?	Identifying NARM methods used by researchers in the literature to solve the NARM problem.
RQ2	What are the several algorithms available for each of the existing NARM methods?	Investigating state-of-the-art algorithms proposed under different NARM methods.
RQ3	What are the advantages and limitations of the existing NARM methods?	Exploring the benefits and limitations of existing NARM methods, along with their classification.
RQ4	Which objectives are considered by the several existing multi-objective optimization NARM algorithms?	Providing an understanding of the objectives used for the multi-objective NARM algorithms under the optimization method.
RQ5	What are the metrics to evaluate the NARM algorithms?	Discussing the metrics that have been used to evaluate algorithms and which ones are the most popular.
RQ6	Which datasets are used for experiments by NARM methods?	Providing a detailed understanding of the datasets used for NARM methods.
RQ7	What are potential future perspectives for the area of NARM?	Discussing the research challenges and future prospects that will help the researchers in future investigations and perform meaningful research.
RQ8	How to automate discretization of numerical attributes for NARM in a useful (natural) manner?	Presenting an automated measure to discretize numerical attributes, which is particularly natural, i.e., which particularly well meets human perception of partitions.

- ACM Digital Library[1]
- IEEE eXplore[2]
- Scopus[3]
- SpringerLink[4]
- ScienceDirect[5]
- Google Scholar[6]

Search Process. Our search was specifically conducted for articles written in English, limited to the period between 1996 and 2022, within the subject area of Computer Science, focusing on the final publication stage. The search query and terms used in Scopus are outlined in Table 3. Through a meticulous search process, we successfully identified a total of 1,628 articles. Following the elimination of 488 redundant articles, we narrowed down the selection to 1,140 articles. Table 4 provides a breakdown of the number of articles obtained from each respective database.

3.2.2 Selection Based on Inclusion and Exclusion Criteria. To ensure the relevance of the articles, we conducted an initial screening process by carefully reviewing the abstracts and conclusions. We

[1] http://dl.acm.org

[2] http://ieeexplore.ieee.org

[3] http://www.scopus.com

[4] http://www.link.springer.com/

[5] https://www.sciencedirect.com/

[6] https://scholar.google.com/

Table 3. Search Terms

Search Term	"Quantitative Association Rule Mining" OR "Numerical Association Rule Mining" OR "Quantitative Association Rules" OR "Numerical Association Rules" OR "Quantitative ARM" OR "Numerical ARM" OR "QARM" OR "NARM"
Search String	(TITLE-ABS-KEY ("Quantitative Association Rule Mining") OR TITLE-ABS-KEY ("Numerical Association Rule mining") OR TITLE-ABS-KEY ("Numerical Association Rule") OR TITLE-ABS-KEY ("Quantitative Association Rule") OR TITLE-ABS-KEY ("Quantitative ARM") OR TITLE-ABS-KEY ("Numerical ARM") OR TITLE-ABS-KEY ("QARM") OR TITLE-ABS-KEY ("NARM")) AND PUBYEAR > 1995 AND PUBYEAR < 2023 AND LIMIT-TO (PUBSTAGE,"final") AND LIMIT-TO (SUBJAREA,"COMP") AND LIMIT-TO (LANGUAGE, "English")

Table 4. Search Results from the Digital Libraries

Digital Library	Number of Results
IEEE Xplore	102
Scopus	223
SpringerLink	618
ACM	187
ScienceDirect	148
Google Scholar	350
Total	1,628
Redundant Articles	488
Non-redundant Articles	1,140

applied the predetermined *Inclusion and Exclusion Criteria*, which are outlined in Table 5. These criteria are widely accepted and primarily focus on aligning with the scope of the study.

Non-peer-reviewed articles, such as theses and abstracts, were excluded from our analysis. Additionally, we also excluded works that combined results from both journals and conferences, such as monographs and books. Following the application of these inclusion and exclusion criteria, we were left with a final set of 96 articles that met our selection criteria. Next, to ensure a comprehensive review, we conducted a thorough examination of the references cited in the selected primary studies. This step aimed to identify any significant publications that might have been missed during the initial search. As a result, we identified 14 additional papers that fulfilled our inclusion criteria. These studies were subsequently incorporated into our list of primary studies, expanding the total number of articles to 110.

3.2.3 Selection based on Quality Assessment. The objective of the quality assessment phase is to ensure the inclusion of unbiased and relevant studies in the review. To accomplish this, we established a set of criteria to evaluate the quality of the papers, refine our search results, and assess the relevance and rigour of the included papers. Following the initial selection based on the predefined inclusion and exclusion criteria, we conducted a thorough reading of the entire article. During this phase, we utilized a quality assessment checklist comprising five criteria, as outlined in Table 6, to refine our search results. Each criterion was evaluated using “Yes,” “No,” or “Partially” responses, which corresponded to scores of 1, 0, or 0.5, respectively. Articles with scores of 2.5 or higher were selected as the final primary studies. Through this rigorous quality assessment process,

Table 5. Inclusion and Exclusion Criteria

ID	Inclusion Criteria
I1	The article discusses a novel method for mining the quantitative or numerical association rules.
I2	The article proposes a novel algorithm for mining the quantitative or numerical association rules.
I3	The article discusses an extension to the existing algorithm for mining the quantitative or numerical association rules.
I4	The article is related to at least one of the proposed research questions.
I5	The article describes the theoretical foundation of mining the association rules from numerical data sets.
ID	**Exclusion Criteria**
E1	Articles which are only application-oriented.
E2	Articles present surveys and short papers.
E3	Abstracts, editorials, thesis, monographs, panels, books.
E4	Conference version of an article whose journal version is included.

we determined a total of 68 articles that met our selection criteria and were deemed as the final primary studies. The list of final articles is available in the GitHub repository[7].

Table 6. Quality Assessment Checklist

ID	Quality Questions
QQ1	Are the proposed methods in the articles well defined?
QQ2	Are the methods/algorithms/experiments defined clearly?
QQ3	Are the results validated?
QQ4	Are their any solid finding/result and clear outcomes?
QQ5	Is the contribution of the article clearly defined?

3.2.4 Data Extraction and synthesis. In the last phase, we extracted pertinent information from the selected articles that successfully passed the quality assessment. This information was utilized to generate a comprehensive summary of our findings. Each chosen article was downloaded and thoroughly examined. Table 7 provides an overview of the extracted data from each publication, highlighting its relevance to the respective research questions. For a more in-depth analysis of the collected data and the synthesis of our findings, we encourage readers to refer to Sections 4 and 6. These sections provide a detailed presentation of the information gathered from the final set of articles, offering valuable insights into the research questions and facilitating a comprehensive understanding of our review's outcomes.

[7] https://github.com/minakshikaushik/List-of-Final-selected-articles.git

Table 7. Data Extracted from Selected Articles Based on Our Research Questions

Extracted Data	Related RQ
Article’s title	General
Author’s name	General
Source name	General
Type of publication (conference/Journal)	General
Year of publication	General
Citation count	General
Methods	RQ1, RQ3
Algorithms	RQ2
Name of datasets	RQ6
Source of datasets	RQ6
Objectives	RQ4
Metrics	RQ5

4 REPORTING THE REVIEW

The reporting phase is crucial as it involves the final presentation and evaluation of the findings obtained from the systematic review. Effectively communicating the results is essential to highlight the contribution of the review and provide valuable insights to readers. These results are derived from the studies identified during the review phase and are aligned with the pre-defined research questions. Through clear and concise reporting, the systematic review aims to enhance understanding and facilitate informed decision-making.

4.1 RQ1.Which methods exist for solving NARM problems?

The selected studies, which are reviewed to examine the existing methods in NARM, are summarized in the subsequent sub-sections. Table 8 provides an overview of the included papers pertaining to different NARM methods. Following a thorough analysis of these studies, it was determined that they could be broadly categorized into four main methods. The following subsections provide brief descriptions of these methods.

4.1.1 The Discretization Method. Classical ARM faces a significant limitation when dealing with continuous variable columns as they cannot be processed directly and must be converted into binary form first. To address this issue, researchers have turned to the discretization method [58, 70, 85]. Discretization involves dividing a column of numeric values into meaningful target groups, which facilitates the identification and generation of association rules. This approach helps to understand numeric value columns easily, but the groups are only useful if the variables in the same group do not have any objective differences. Additionally, discretization minimizes the impact of trivial variations between values. The discretization method for mining numerical association rules can be categorized into four approaches: partitioning, clustering, fuzzifying and hybrid. In this article, we have selected 28 relevant studies that focus on the discretization method.

Partitioning Approach. Srikant [96] presented a solution for mining association rules from quantitative data sets. The approach involved partitioning the numerical attributes into intervals and subsequently mapping these intervals into binary attributes. To address the information loss resulting from partitioning, the authors introduced the concept of the *partial completeness measure.* By partitioning the numerical attributes and mapping them into binary attributes, Srikant’s approach allowed for the application of traditional ARM techniques to quantitative data. This work laid the foundation for handling numerical attributes in ARM and has since influenced further developments in the field.

Clustering Approach. The clustering approach is utilized to divide a numerical column into distinct groups based on similarity among values. Various clustering techniques, including merging-based, density-based, and grid-based clustering, can be employed to achieve this goal. From the clustering approach, we identified nine relevant articles that explore this methodology.

In the merging and splitting-based concept, intervals are merged initially and then subsequently split based on specific criteria. Wang and Han proposed the notion of merging adjacent intervals in their work [104]. Li et al. [66] developed a method that identifies intervals of numeric attributes and merges adjacent intervals exhibiting similar characteristics based on predefined criteria. These studies contribute to the understanding and advancement of the merging and splitting-based approach within the context of NARM.

The density-based clustering aims to identify different dense regions within the dataset and map these regions to numeric association rules. Algorithms such as DRMiner [67], DBSMiner [40], and MQAR [109] are examples of techniques proposed within this category. Further details regarding these algorithms will be provided in response to the subsequent research question. On the other hand, grid-based clustering utilizes a bitmap grid to handle data clustering. It identifies clusters within the bitmap grid, which subsequently yield association rules. This method offers an alternative approach for extracting meaningful associations from numerical attributes.

Fuzzy Approach. The fuzzy approach is employed to tackle the issue of sharp boundaries in ARM by representing numerical values as fuzzy sets. Fuzzy sets allow for the representation of intervals with non-sharp boundaries, where an element can possess a membership value indicating its degree of belonging to a set. Hong et al. [46] applied the fuzzy concept in conjunction with the apriori algorithm to discover fuzzy association rules from a quantitative dataset. Their work demonstrated the effectiveness of combining fuzzy sets and ARM techniques for extracting valuable insights from numerical data.

Hybrid Approach. The hybrid approach for solving NARM problems is the combination of two or more methods such as clustering, partitioning, and fuzzy approaches. This method is a more flexible approach that can enhance the efficiency and accuracy of ARM. For instance, [113] combined the fuzzy approach with the partitioning method to develop an efficient algorithm for mining fuzzy association rules. On the other hand, [59, 84, 100] utilized the fuzzy approach with clustering to enhance the accuracy of ARM. The hybrid approach in NARM offers a promising direction for researchers to explore, as it allows for the utilization of complementary techniques to address the complexities of mining association rules from numerical data.

4.1.2 The Optimization Methods. In the context of NARM, the optimization method has gained significant attention, and we identified 34 papers out of the 68 studies reviewed that focused on optimization methods. These methods utilize heuristic algorithms inspired by various natural phenomena, such as animal movements and biological behavior. Generally, optimization methods fall into two categories: bio-inspired and physics-based. Depending on the optimization goals, the optimization methods can be further classified into single-objective and multi-objective approaches.

Bio-inspired optimization methods consist of approaches based on Swarm Intelligence (SI), Evolutionary algorithms, and Hybrid methods. These methods draw inspiration from the collective behavior of organisms in nature. For example, some studies have explored algorithms inspired by the movements of wolves [3], insects [88], and mining behavior in biological systems [79]. The physics-based optimization methods apply principles from physics to solve optimization problems.

Table 8. Overview of Solutions Based on NARM Methods

Methods	Approaches	# included papers	References
Discretization	Partitioning	9	[22, 23, 25, 33, 36, 66, 92, 95, 96]
	Clustering	9	[29, 40, 65, 67, 70, 81, 82, 104, 109]
	Fuzzy	6	[26, 42, 46, 62, 64, 114]
	Hybrid	4	[59, 84, 100, 113]
Optimization	Evolutionary	17	[6, 10, 14, 72–80, 83, 86, 93, 98, 108]
	Differential Evolution	3	[9, 12, 34]
	Swarm Intelligence	11	[3, 7, 8, 20, 44, 49, 61, 63, 88, 99, 107]
	Physics-based	1	[24]
	Hybrid	2	[13, 87]
Statistical		3	[17, 50, 106]
Other		3	[47, 48, 56]

These approaches offer researchers a diverse range of techniques to explore and apply in NARM, allowing for the discovery of efficient and effective association rules from numerical data.

Evolution-Based Methods. The evolutionary method in NARM is rooted in Darwin's theory of natural selection, which highlights the adaptive nature of living organisms in response to changing environments. This approach employs biological operators, including crossover, mutation, and selection, to mimic the evolutionary process in optimization algorithms [31]. By applying these principles, evolutionary methods aim to enhance the effectiveness and efficiency of NARM algorithms, allowing for the discovery of valuable association rules from numerical data.

Under the evolution-based method, the genetic algorithm (GA) and differential evolution (DE) provide detailed solutions for the NARM problem. The optimization method aims to discover association rules without the need for the prior discretization of numerical attributes. GA, a meta-heuristic inspired by natural selection and genetic structure, evolves a population of individual solutions over time [45]. It proceeds in three main steps: selection of parent individuals, crossover to combine parents for the next generation, and mutation to apply random changes to parents and form children. In 2001, the concept of genetic algorithms was successfully applied to identify numerical association rules from numerical attributes [79].

Out of the selected studies, 17 refers to the use of genetic algorithms. Initially, NARM algorithms focused solely on single-objective problems; later, multi-objective algorithms also came into the scenario [83]. Over the years, the genetic algorithm has been used with some advancement by integrating various supporting techniques, such as the binary-coded CHC algorithm [73], non-dominated sorting genetic algorithm [78], and niching genetic algorithm [76], as well as other multi-objective genetic algorithms. Genetic programming [60], which utilizes a tree structure for the genome, is another aspect of the genetic algorithm. Grammar-guided genetic programming [71, 72] also emerged with NARM in 2004.

In 1997, Storn and Price [97] introduced a global optimization meta-heuristic approach that effectively minimized non-differentiable, non-linear, and multi-modal cost functions. This approach utilized the same operator as genetic algorithms, which included crossover, mutation, and selection. To minimize the function, differential evolution (DE) employed a few control variables and parallelization techniques, which helped to decrease computing costs and quickly converge on the global minimum. Our research identified four relevant studies that used DE for NARM. One such study, proposed in 2008 by Alatas and Akin, utilized a multi-objective differential evolution algorithm [9]. Another study was conducted in 2018 and 2021 by I. Fister Jr. [34, 35], while Altay and Alatas presented a hybrid DE-based method with a sine cosine algorithm and chaos number-based encoding, respectively [12, 13].

Swarm Intelligence-Based. Swarm intelligence (SI) is a popular optimization technique inspired by the collective behavior of self-organized groups in nature, as described by Bonabeau et al. in 1999 [21]. SI algorithms emulate the behavior of swarms found in birds, fish, honey bees, and ant colonies. These algorithms consist of individuals that migrate through the search space, simulating the progression of the swarm. Various SI-based algorithms have been developed, including Particle Swarm Optimization (PSO), Bat Algorithm (BAT), Ant Colony Optimization (ACO), Cat Swarm Optimization (CSO), and others. In the context of solving NARM problems, several SI algorithms have been applied. Notable examples include PSO [7], BAT [44], Wolf Search Algorithm (WSA) [3], Crow Search Algorithm (CSA) [63], and Cuckoo Search Algorithm (CS)[49]. These SI-based algorithms have shown promise in optimizing NARM and extracting meaningful association rules.

Particle swarm optimization (PSO) is a widely used optimization technique for non-linear continuous functions inspired by the movement of bird flocks or fish schools as described in Kennedy and Eberhart [57]. PSO simulates the collective behaviour of these groups, where N particles move in a D-dimensional search space, adjusting their position iteratively by using their own best position *pbest* and the best position of the entire swarm *gbest*. The PSO algorithm finds the optimum solution by calculating the velocity and position of each particle. In the context of mining association rules with numeric attributes, Alatas and Akin introduced the application of PSO in 2008 [7]. They modified the PSO algorithm to search for numeric attribute intervals and discover numeric association rules. Seven studies have since focused on adapting PSO for NARM including the hybrid approach. These studies explore the potential of PSO to effectively mine association rules with numeric attributes and provide valuable insights into its performance and limitations.

Ant colony optimization (ACO) is another optimization technique based on the foraging behaviour of various ant species, as described in Dorigo et al.[30]. In ACO, a group of artificial ants collaborates to find solutions to an optimization problem and communicate information about the quality of these solutions using a communication mechanism similar to real ants. ACO is designed to address discrete optimization problems by selecting a solution using a discrete probability distribution. In the context of multi-objective NARM, Moslehi et al. introduced an ACO variant called ACO_R in 2011 [88]. ACO_R utilizes a Gaussian probability distribution function to handle continuous values encountered in NARM. It maintains a solution archive of size k, initially populated with k random solutions ranked by their quality. Each ant constructs its solution by probabilistically selecting a solution from the archive, allowing for the exploration of different solution possibilities. The utilization of ACO in NARM, particularly the ACO_R variant, demonstrates its potential to address the challenges posed by continuous attributes and provide effective solutions for multi-objective NARM problems.

The Cuckoo Search algorithm (CS) is an optimization algorithm introduced by Yang and Deb in 2009, inspired by the brooding parasitic behavior of cuckoo species [111]. Cuckoos lay their eggs in the nests of other bird species, mimicking the color and pattern of the host birds' eggs. Some host birds may recognize the stranger's eggs and remove them from the nest. The cuckoo search algorithm mimics this behavior by generating new solutions (cuckoo eggs) and replacing less promising solutions in the nests (solution space) with the new solutions. The algorithm operates based on three main rules: A cuckoo bird lays only one egg at a time in a randomly chosen nest (introduces a new solution to the search space). The nests with high-quality eggs are more likely to be carried over to the next generation (the better solutions have a higher chance of survival). The probability of a host bird discovering cuckoo eggs in its nest is either 0 or 1 (either the host bird finds and removes the cuckoo egg or it remains undetected). The goal of the cuckoo search algorithm is to find new and potentially better solutions to replace the existing solutions in the nests, leading to the improvement of the overall solution quality. In the context of NARM, a multi-objective cuckoo search algorithm called MOCANAR was proposed by Kahvazadeh et al. in 2015 [49]. MOCANAR applies a Pareto-based approach to solve the multi-objective NARM problem, aiming to discover association rules that optimize multiple conflicting objectives simultaneously. By employing the cuckoo search algorithm as the underlying optimization technique, MOCANAR demonstrates its effectiveness in addressing the challenges of multi-objective NARM.

In 2012, Tang et al. proposed a heuristic optimization algorithm called the Wolf Search Algorithm (WSA) that imitates how wolves hunt for food and survive in the wild by avoiding predators [102]. Unlike other bio-inspired meta-heuristics, WSA enables both individual local searching and autonomous flocking movement capabilities as wolves hunt independently in groups. WSA follows three basic rules based on wolf hunting behavior. The first rule involves a fixed visual area of each wolf with a radius v, which is calculated using Minkowski distance. The second rule pertains to the current position of the wolf, represented by the objective function's fitness, and the wolf always tries to choose the better position. The third rule concerns escaping from enemies. Agbehadji suggested WSA to develop an algorithm for searching for intervals of numeric attributes and association rules [3].

In 2010, Yang introduced the BAT algorithm (BA) as a solution to continuous constrained optimization problems inspired by the echolocation behavior of microbats [110]. Microbats use echolocation to sense distance, discover prey, avoid obstacles, and find roosting nooks in the dark. The BA algorithm is based on the velocity of a bat at a particular position, with a fixed frequency and varying wavelength and loudness. The bat adjusts its frequency and loudness to locate a new food source while changing its position in space. Heraguemi et al. [44] proposed a multi-objective version of the Bat algorithm for numerical attributes. Previously, the BA was also used for ARM to deal with categorical attributes.

The Crow Search Algorithm (CSA) is a recently developed meta-heuristic optimization technique inspired by the intelligent behaviour of crows [15]. Crows are known for their ability to store and hide food for future use while also keeping an eye on each other to steal food. The CSA is based on four principles of crow behaviour: living in flocks, memorizing the position of hiding places, following other crows to steal food, and protecting their caches from theft. In the CSA, a crow flock moves in a d-dimensional search space, with each crow having its own position and memory of its hiding place. When a crow follows another crow, it may either discover the hiding place and memorize it or be tricked by the followed crow. The CSA has been successfully applied to various optimization problems, such as image segmentation and feature selection. Recently, Makhlouf et al. (2021) [63] proposed a discrete version of CSA for NARM.

Hybrid Approach. The hybrid approach in optimization combines multiple techniques such as evolution, SI, or other approaches to leverage their respective advantages and tackle complex tasks effectively. In the context of NARM, researchers have explored the hybridization of different algorithms to enhance the performance and efficiency of association rule discovery. One study by Moslehi et al. [87] employed a hybrid approach that combined the GA and PSO. The GA facilitated the search for the best solution, while the PSO helped avoid being trapped in local optima by exploring a larger search space. By combining the strengths of both approaches, the hybrid algorithm demonstrated the ability to find high-quality solutions to complex NARM problems within a relatively short time. Another study by Altay and Alatas [13] proposed a hybrid approach that combined the DE algorithm with the sine and cosine algorithms. This hybridization aimed to leverage the exploration and exploitation capabilities of both algorithms, resulting in improved performance for NARM. The DE algorithm provided efficient search and optimization, while the sine and cosine algorithms introduced chaos-based techniques to enhance the exploration process.

Physics-Based. Physics-based meta-heuristics have emerged as a powerful approach to solving optimization problems. One such algorithm, the gravitational search algorithm (GSA) [91], is based on Newton's law of gravity, where particles attract each other with a gravitational force. The following formula defines this force:

$$F = G\frac{M_1 M_2}{R^2} \tag{3}$$

where F is the gravitational force, G is the gravitational constant, M_1 and M_2 are the mass of of two particles and R is the distance between these particles. According to Newton's second law, when a force is applied to a particle, its acceleration a depends on the force F, and it is mass M.

$$a = \frac{F}{M} \tag{4}$$

In the GSA, agents are considered as objects with masses that determine their performance. The heavier masses are better solutions and attract lighter masses, leading to an optimal solution. Each mass has a position, inertial, active, and passive gravitational mass. The position of a mass represents a problem solution, and its gravitational and inertial masses are calculated using a fitness function. While GSA has been used in various optimization problems, it has only been applied to NARM in one study, where Can and Alatas utilized it for finding intervals of numeric attributes automatically without any prior processing [24].

4.1.3 The Statistical Method. Statistics is a traditional approach for developing theories and testing hypotheses using statistical tests such as Pearson correlation, regression, ANOVA, t-test, and chi-square test, among others. Statistical inference involves inferring population properties from a sample to generate estimates and test hypotheses. Some studies have used statistical concepts such as mean, median, and standard deviation in the mining association rule. We identified three studies in this direction which suggested distribution-based interestingness measures. One such study is Kang et al. (2009) [50], which used bipartition techniques such as mean-based bipartition, median-based bipartition and standard deviation minimization for quantitative attributes in ARM.

4.1.4 Miscellaneous Other Methods. In addition to the established techniques discussed earlier, there are other alternative approaches that have been proposed to tackle the challenge of NARM. These approaches offer unique perspectives and methodologies to address the problem. One such approach is the utilization of mutual information, as presented by Yiping et al. in 2008 [56]. Mutual information is a concept from information theory that measures the dependency between two variables. In the context of NARM, mutual information is employed to generate quantitative association rules (QARs), capturing the relationships and dependencies between numerical attributes. Another approach is the use of Variable Mesh Optimization (VMO), proposed by Jaramillo et al. [48]. VMO is a population-based metaheuristic algorithm that represents solutions as nodes distributed in a mesh-like structure. Each node in the mesh represents a potential solution to the optimization problem. By leveraging the principles of VMO, the algorithm explores the solution space in a distributed and adaptive manner, facilitating the discovery of association rules. Furthermore, in 2021, Hu et al. [47] introduced a cognitive computing-based approach for NARM. Cognitive computing refers to the simulation of human thought processes by computer models. By leveraging cognitive computing techniques, the proposed approach aims to mimic the human thought process during critical situations, allowing for a more comprehensive and nuanced analysis of numerical data for ARM.

These alternative approaches demonstrate the diverse range of methodologies and concepts that researchers have explored to tackle the NARM problem. By leveraging mutual information, variable mesh optimization, and cognitive computing, these approaches offer unique perspectives and potential benefits for discovering association rules from numerical data.

4.2 RQ2 What are the several algorithms available for each of the existing NARM methods?

In response to RQ1, we have provided a comprehensive explanation of the four main methods utilized in NARM in subsection 4.1. This section further delves into a more detailed exploration of the algorithms associated with each of these methods.

4.2.1 The Discretization Method.

Partitioning Based Algorithms.

- *Qunatitative Association Rule Mining (QARM):* In 1996, Srikant and Agrawal proposed an algorithm [96] to address the use of numeric attributes in ARM, which was traditionally limited to binary attributes. One key issue was determining whether and how to partition a quantitative attribute while minimizing information loss by setting minimum support and confidence thresholds. To overcome this, the algorithm introduces a *partial completeness measure*. The algorithm converts categorical attributes to integers and partitions numerical attributes into intervals using an equi-depth discretization algorithm. Frequent itemsets are then generated by setting minimum support for each attribute and used to generate association rules. To ensure interesting and non-redundant rules, the algorithm employs an interesting measure called "greater-than-expected-values." However, setting the user-supplied threshold too high can result in missed rules, while setting it too low can generate irrelevant rules.
- *Automatic Pattern Analysis and Classification System 2 (APACS2):* To address the threshold issue, a novel algorithm named APACS2 was presented by Chan et al. [25]. This algorithm employed equal-width discretization to discover intervals of quantitative attributes without the need for user-defined thresholds. The quantitative attribute values were mapped to these intervals to obtain a new set of attributes. Each interval was described by the lower and upper bounds as $a_1 = [l_1, u_1]$. The APACS2 algorithm used *adjusted difference* analysis to identify interesting associations between items, which enabled it to generate both positive and negative association rules.
- *Q2:* Buchter and Wirth [23] proposed the *Q2* algorithm to work with multi-dimensional association rules over ordinal data. *Q2* aimed to reduce the cost of counting a large number of buckets by only counting the buckets of successful candidates. First, apriori is used to identify all frequent boolean itemsets. Then, only the items in these sets are discretized based on the user's specifications. Q2-gen technique is used to generate a prefix tree that includes only the bucket combinations that need to be counted for the discretized items. The prefix tree is then used to count these bucket combinations in a single pass through the data. Finally, the prefix tree is used to produce all R-interesting rules. Unlike the hash tree used in *QARM*, *Q2* uses a prefix tree to store quantitative itemsets.
- *Fukuda et al. Work:* Fukuda et al. presented a novel algorithm [36] that computes two optimized ranges for numeric attributes. To achieve this, the algorithm uses randomized bucketing as a preprocessing step to compute the ranges for sorted data. The focus of the algorithm

is on generating optimized rules of the format $(A[v_1, v_1]) \wedge C_1 \Rightarrow C_2$, where C_1 and C_2 are binary attributes and A is a numeric attribute. The main task of the algorithm is to generate thousands of equi-depth buckets and combine some of them to generate optimized ranges. The performance of the bucketing algorithm was compared with Naive Sort and Vertical Split Sort, and the algorithm demonstrated superior performance.

- *Brin's Algorithm:* In 1999, Brin et al. proposed an optimized algorithm for mining one and two numeric attributes [22]. The focus of the algorithm was on optimizing gain rules, where the gain of a rule R is defined by the difference between the support of $(antecedent \wedge consequent)$ and the support of $antecedent$, multiplied by the user-specified minimum confidence. To reduce the input size, a bucketing algorithm was employed. For one numeric attribute, the algorithm computes optimized gain rules, while for two numeric attributes, a dynamic programming algorithm was presented to compute approximate association rules. Although the algorithm was successful for one numeric attribute, it was not well-suited for large domain sizes in the case of two numeric attributes.
- *Numerical Attribute Merging Algorithm:* Li et al. [66] developed an algorithm that merges adjacent intervals of numeric attributes based on a merging criterion that considers value densities and distances between values. They called this the *numerical attribute merging algorithm* and used it to find suitable intervals for the QARM algorithm. After discretizing the numeric attributes, this algorithm treats each interval as a boolean attribute, allowing them to work with classical ARM.
- *Rastogi's Algorithm:* In 2002, Rastogi and Shim extended the work done by [92]. They presented efficient methods for reducing the search space during the computation of optimized association rules applicable to both categorical and numeric attributes.
- *Sliding Window Partitioning - Random Forest (SWP-RF) Algorithm:* In a related study, Guanghui Fan et al. [33] proposed a machine learning-based QARM method called *SWP-RF* to identify factors that cause network deterioration. This method uses sliding window partitioning (SWP) to discretize continuous attributes into boolean values, followed by random forest (RF) feature importance to measure the association between key performance indicator (KPI) and key quality indicator (KQI).
- *Numerical Association Rule-Discovery:* Song and Ge [95] proposed NAR-Discovery, a divide-and-conquer algorithm for mining numerical association rules. NAR-Discovery progresses in two phases. In the first phase, attributes are partitioned into a small number of large buckets, and then neighbouring buckets are mapped to an "item," and apply a classical frequent itemset mining algorithm. In the second phase, only the outermost buckets of each rule are recursively partitioned, and some bounds and filtering are used to end the process. The authors improved performance by one to two orders of magnitude using optimization techniques. They developed a search based on a tree structure to manage rule derivations, and interesting rules were selected using an optimization technique based on temporary tables. NAR-Discovery was compared with QuantMiner [93] and claimed to discover all appropriate rules.

Clustering Based Algorithms.

- *Miller's Algorithm:* Miller et al. [82] introduced a distance-based ARM approach for interval data in 1997. To handle the memory requirements, they utilized a B^+ tree data structure. The authors first used a clustering algorithm to identify intervals and then applied a standard ARM algorithm to extract association rules from these intervals.
- *Association Rule Clustering System (ARCS):* In 1997, Lent et al. [65] introduced a comprehensive framework called ARCS that focused on rules with two quantitative attributes on

the antecedent side and one categorical attribute on the consequent side. ARCS consists of four main components: binner, association rule engine, clustering, and verifier. In the binner phase, quantitative attributes are divided into bins using the equi-width binning method, and these bins are then mapped to integers. The BitOp algorithm is used to enumerate clusters from the grid and locate them within the Bitmap grid by performing bitwise operations, which results in clustered association rules. However, this method is limited to handling low-dimensional data and cannot handle high-dimensional data.

- *Interval Merger Algorithm:* In 1998, Wang and Han [104] proposed an algorithm for merging adjacent intervals of numeric attributes by evaluating merging criteria. This algorithm has two phases: initialization and bottom-up merging. They used an M-tree, which is a modified B-tree, to efficiently find the best merge during the merging phase. Additionally, two interestingness measures, J_1 and J_2, were used to evaluate the interestingness of the discovered association rules. The higher the values for both measures, the more interesting the rule was considered to be.
- *Relative Unsupervised Discretization (RUDE):* In 2000, Ludl et al. proposed the RUDE algorithm as a merging approach based on the merging and splitting technique [70]. The RUDE algorithm considers the interdependence of attributes and consists of three main steps. The first step is the pre-discretizing phase, where equal-width discretization is applied to the data. In the second step, called structure projection, the structure of each source attribute is projected onto the target attribute. This projection is then used to perform clustering on the target attribute, resulting in the gathering of split points in the split point list. Finally, in the postprocessing step, the split points are merged using predefined merging parameters. The RUDE algorithm was primarily used as a preprocessing step for the apriori algorithm. The association rules extracted from RUDE and apriori were combined to obtain the final results.
- *Dense Regions Miner (DRMiner):* In 2005, Lian et al. [67] proposed the DRMiner algorithm, which efficiently identifies dense regions and maps them to QARs. To achieve this, the authors developed a three-step approach. First, a $k-d$ tree is built to store valid cells in the space and their corresponding number of points. Second, a dense region cover set is grown inside some leaf nodes from their boundaries, and self-merging of cover sets is done across boundaries. Finally, the cells are traversed in each cover to find dense regions. The authors evaluated the complexity of DRMiner for different steps and used a synthetic data set with varying numbers of attributes and instances for evaluation.
- *Density-Based Sub-space Miner (DBSMiner):* The DBSMiner algorithm, proposed in 2008 by Guo et al. [40], aims to cluster the high-density subspace of quantitative attributes. CBSD (Clustering Based on Sorted Dense Units), a new clustering algorithm, was used to sort all subspaces with densities greater than a certain threshold in descending order. Interestingly, DBSMiner has a unique property when dealing with low-density subspaces: it only needs to verify the neighbouring cell instead of scanning the entire space. The algorithm is capable of uncovering interesting association rules.
- *Mining Quantitative Association Rule (MQAR):* Yang et al. [109] proposed the MQAR algorithm in 2010, which utilizes dense regions to generate numerical association rules. The algorithm clusters dense subspaces using the DGFP tree (dense grid frequent pattern tree) in four main steps. Firstly, the data space is partitioned into non-overlapping rectangular units by partitioning each quantitative attribute into intervals. Then, a DGFP tree is created to store dense cells in the space with a density greater than the minimal density criterion by mapping all database transactions into a high-dimensional space S and sorting units by density. The third step is to mine the DGFP tree to obtain dense subspaces, which provide information about database transactions. Finally, the dense subspaces in S are identified based on the

dense subspaces, and associated cells are found to build clusters. Association rules that are not redundant are then constructed using the clustering result.

- *Quantitative Association Rule Mining Method with Clustering Partition (QARC_Apriori):* The QARC_Apriori algorithm, proposed in 2014, aimed to analyze correlations in satellite telemetry data [29]. The algorithm involved three main steps: First, it performed dimensionality reduction to eliminate redundant attributes. Second, it discretized numeric attributes using the K-means clustering algorithm. Finally, it used the apriori algorithm for mining QARs, with frequent itemset mining and rule generation. Since satellite telemetry data has a vast amount of data, numerical attributes, and high dimensions with various attributes such as voltage, current, pressure, and temperature, the authors used the grey relational analysis method to reduce the dimensionality.
- *Graph Clustering and Quantitative Association Rules (GCQAR):* Medjadba et al. [81] proposed GCQAR, a method for discovering significant patterns in geochemical data by combining graph clustering and QARM. Identifying hidden patterns related to mineralization in geochemical data is a challenging task. The proposed method tackles this by first applying graph clustering to partition the input data into highly cohesive, sparsely connected subgraphs. This step helps to separate the relevant geochemical data from the complex background. Then, QARs are used to measure the interrelation between pairs of vertices in each subgraph. For each cluster, a set of QARs is generated by randomly selecting antecedent and consequent rules and evaluating them based on support and confidence.

Fuzzy Based Algorithms.

- *Fuzzy-Automatic Pattern Analysis and Classification System (F-APACS):* Chan extended the APACS2 algorithm for QARM by proposing the F-APACS algorithm [26], which is based on fuzzy set theory and is designed for mining association rules with numeric attributes. Instead of finding intervals for quantitative attributes as done in other methods, F-APACS uses linguistic terms to represent discovered patterns and exceptions. Similar to APACS2, F-APACS also employs the *adjusted difference analysis* technique, which eliminates the need for a user-supplied threshold and can discover both positive and negative association rules. To capture the uncertainty associated with the fuzzy association rules, F-APACS uses a weight of evidence measure to represent confidence.
- *Kuok's Approach:* Kuok et al. [62], proposed the method for mining fuzzy association rules of the form, "If X is A then Y is B." Here X, Y are attributes and A, B are fuzzy sets. This approach is important because it provides a better way of handling numeric attributes compared to existing methods. The study showed that the use of fuzzy sets helps to understand the correlation between two attributes through the significance factor and certainty factor.
- *Fuzzy Transaction Data mining Algorithm (FTDA):* Hong et al. [46] used the fuzzy concept with the apriori algorithm to discover fuzzy association rules from a quantitative data set. To overcome the limitation of the apriori algorithm in handling quantitative data, the authors introduced the FTDA (Fuzzy Transaction Data mining Algorithm), which first transformed quantitative data into linguistic terms using membership functions. Next, the scalar cardinalities of all linguistic terms were calculated, and the apriori algorithm was modified to find association rules as fuzzy sets. However, a drawback of this method is that experts need to provide the best fuzzy sets of quantitative attributes manually.
- *Gyenesei's Approach:* Gyenesei [42] addressed the limitation of expert dependency in selecting fuzzy sets for quantitative attributes by introducing a fuzzy normalization process. To obtain unbiased membership functions, the author proposed using fuzzy covariance and fuzzy correlation values. Interest measures were defined in terms of fuzzy support, fuzzy confidence,

and fuzzy correlation. The approach was evaluated using two methods: with normalization and without normalization. The non-normalized method produced the most interesting rules, while the number of rules generated by the normalized approach was comparable to the discrete method. The fuzzy normalization process helped to reduce anomalies that may arise from the arbitrary selection of fuzzy sets.

- *Generalized Fuzzy Quantitative Association Rule Mining Algorithm:* Lee [64] proposed a novel algorithm for generalized fuzzy QARM, incorporating fuzzy concept hierarchies for categorical attributes and fuzzy generalization hierarchies of linguistic terms for quantitative attributes. Unlike other methods, this approach calculates the weighted support and weighted confidence by taking into account the importance weights of attributes. To eliminate redundant rules, the *R*-interest measure is used. The algorithm converts each transaction into an augmented transaction and applies apriori [5] to generate frequent itemsets with the aid of weighted support and weighted confidence measures. It then extracts QARs by removing rules not meeting the R-interest measure's criteria.
- *Optimized Fuzzy Association Rule Mining(OFARM):* Zheng et al. [114] proposed a novel algorithm, OFARM (optimized fuzzy association rule mining), in 2014 to optimize the partition points of fuzzy sets with multiple objective functions. The frequent itemsets are generated using a two-level iteration process, and the certainty factor with confidence is used to evaluate fuzzy association rules.

Hybrid Based Algorithms .

- *Equal-Depth Partition with Fuzzy Terms (EDPFT):* Zhang [113] proposed an enhanced version of the *equi-depth partition (EDP)* algorithm that integrated fuzzy terms, called EDPFT. This algorithm was designed to identify association rules that contain intervals, crisp values, and fuzzy terms on both the left-hand and the right-hand sides. Unlike FTDA, which relies on user-supplied fuzzy sets, EDPFT utilizes equi-depth partitioning to obtain the intervals of numeric attributes. Although the author did not evaluate the algorithm using any data set, this approach shows potential in dealing with both crisp and fuzzy values in ARM.
- *Mohamadlou et al. Algorithm:* Mohamadlou et al. [84] introduced a fuzzy clustering-based algorithm for mining fuzzy association rules. The algorithm utilizes *C*-means clustering to cluster all the transactions, followed by obtaining the fuzzy partition for each attribute. It then converts the quantitative transactions into 'fuzzy discrete transactions' by mapping the quantitative data into fuzzy partitions. The algorithm mines fuzzy association rules from the 'fuzzy discrete transactions' using an ARM algorithm.
- *Fuzzy Inference Based on Quantitative Association Rule (FI-QAR):* Wang et al. [105] proposed a three-phase algorithm called FI-QAR, which integrates clustering and fuzzy techniques. In the first phase, the density-based fuzzy adaptive clustering (DFAC) [69] algorithm was applied to discretize numeric attributes into discrete intervals. The intervals were then combined with the TS fuzzy model to generate a nominal vector matrix, which was used to modify the A apriori algorithm and reduce the scanning overhead of a large database. The second phase involved mining QARs using an improved apriori algorithm. Finally, the third phase pruned the association rules. The proposed approach offers a way to mine QARs from large databases effectively.
- *Fuzzy Class Association Rule Support Vector Machine (FCARSVM:)* Kianmehr et al. [59] proposed the FCARSVM to obtain fuzzy class association rules. The authors extracted Fuzzy Class Association Rules (FCAR) using a fuzzy *C*-means clustering algorithm in the first phase, and FCARs were weighted based on the scoring metric strategy in the second phase.

4.2.2 The Optimization Method.

Evolution and DE-Based Algorithms.

- *GENetic Association Rules (GENAR):* Mata et al. [79] introduced GENAR as a genetic algorithm-based solution for NARM. GENAR is designed to identify numerical association rules with an unknown number of numeric attributes in the antecedent and a single attribute in the consequent. By utilizing genetic algorithms, GENAR offers an effective approach to discovering association rules involving numerical attributes.
- *Genetic Association Rules (GAR):* The extended version of GENAR, called GAR, was proposed by Mata et al. [80]. GAR utilizes the five fundamental phases of a genetic algorithm, namely initialization, evaluation, reproduction, crossover, and mutation, to discover intervals for numerical attributes. A key contribution of GAR is the introduction of a fitness function to determine the optimal amplitude for each numerical attribute's interval. The genes in GAR represent the upper and lower limits of the attribute intervals and are initially created randomly. Through crossover and mutation operations, a new generation of genes is generated, and the fitness function is used to evaluate the quality of the intervals. GAR provides an effective approach for identifying appropriate intervals for numerical attributes in ARM.
- *Genetic Association Rules Plus (GAR Plus):* Alvarez et al. [14] made enhancements to the GAR algorithm and introduced GAR Plus. This improved version enables the automatic extraction of intervals for numerical attributes through an evolutionary process, eliminating the need for pre-discretization. GAR Plus enhances the fitness function of GAR by incorporating additional parameters such as support, confidence, interval amplitude, and the number of attributes with a modifier. By considering these parameters, GAR Plus provides a more comprehensive evaluation of the fitness of candidate intervals, resulting in improved performance and accuracy compared to the original GAR algorithm.
- *Alatas and Akin Algorithm:* Alatas and Akin [6] have made significant contributions to the field of NARM. In one of their studies, Alatas extended the GAR algorithm to discover both positive and negative association rules. They compared the performance of their proposed algorithm with the original GAR algorithm and observed that the amplitude of the intervals generated by their approach was lower than that of GAR. This indicates that the extended algorithm by Alatas and Akin was able to identify more specific and precise intervals for numerical attributes, resulting in improved rule discovery.
- *QuantMiner:* QuantMiner [93] is a system for discovering QARs that employs a genetic algorithm. The system operates with a predefined set of rule templates, which can be either user-selected or computed by the system itself. These templates define the format of the QARs. By utilizing the genetic algorithm, QuantMiner searches for the optimal intervals for the numerical attributes specified in the rule templates. This approach allows the system to efficiently explore the search space and identify association rules that meet the desired criteria.
- *Expending Association Rule Mining with Genetic Algorithm (EARMGA):* Yan et al. [108] presented an encoding method for discovering association rules using a genetic algorithm. Their approach, named ARMGA, initially designed for boolean attributes, was extended to handle generalized association rules incorporating both categorical and quantitative attributes. The authors introduced a fitness function based on relative confidence, eliminating the need for a user-defined minimum support threshold. To handle quantitative attributes, they discretized them into intervals and integrated four genetic operators into the algorithm. The resulting enhanced version, EARMGA, successfully accommodated quantitative attributes and utilized the k-FP tree data structure for efficient rule mining.

- *Real-Coded Genetic Algorithm (RCGA):* Martinez et al. [74] introduced a real-coded genetic algorithm (RCGA) for NARM. The RCGA is a variation of the binary-coded CHC algorithm [32], known for its elitist selection mechanism that favors the best individual for the next generation. In the context of NARM, the RCGA is utilized to search for optimal intervals. By employing real-coded representations and incorporating the elitist selection feature, the RCGA aims to efficiently explore the search space and discover high-quality numerical association rules.
- *Quantitative Association Rules by Genetic Algorithm (QARGA):* Martínez et al. [73] improved the RCGA by proposing QARGA to extract QARs from real-world multidimensional time series. The QARGA method discovered significant relationships between ozone concentrations in the atmosphere and other climatological time series, including temperature, humidity, wind direction, and speed.
- *Niching Genetic Algorithm for Quantitative Association Rules (NICGAR):* NICGAR was proposed by Martin et al. [76] to prevent the generation of similar rules by reducing the set of quantitative rules, which includes positive and negative rules. The algorithm consists of three components: an external population, a punishment mechanism, and a restarting process to manage niches and avoid the same solutions. The article also proposes a new similarity measure to find the similarity between rules.
- *QAR-CIP-NSGA-II:* Martin et al. [78] presented a novel multi-objective evolutionary algorithm called QAR-CIP-NSGA-II, which extends NSGA-II to simultaneously learn the intervals of attributes and conditions for each rule in a QAR system. QAR-CIP-NSGA-II aims to discover a set of high-quality QARs that balance interpretability and accuracy by maximizing comprehensibility, interestingness, and performance objectives. The algorithm incorporates an external population and a restarting method to enhance population diversity and store discovered nondominated rules. The comprehensibility of a rule is measured by the number of attributes involved in the rule, while the product of the certainty factor and support determines the accuracy. The interestingness measure, lift, is used to determine how significant the rule is.
- *Multi-Objective Genetic algorithm Association Rule mining (MOGAR):* Minaei-Bidgoli et al. [83] proposed MOGAR algorithm for discovering association rules from numerical data. The algorithm maintains a population of candidate association rules, representing potential solutions, and applies genetic operators such as selection, crossover, and mutation to evolve the population over successive generations. The fitness of each candidate rule is evaluated based on multiple objectives, such as confidence, interestingness and comprehensibility. MOGAR employs a Pareto dominance concept to identify non-dominated solutions MOGAR has the ability to handle complex datasets with multiple conflicting objectives, providing a more comprehensive view of associations in the data.
- *Multi-Objective Positive Negative Association Rule Mining Algorithm (MOPNAR):* Martin et al. [77] proposed MOPNAR, a multi-objective algorithm that aims to achieve the same objectives as QAR-CIP-NSGA-II, including mining a reduced set of positive and negative QARs. The authors also claimed to achieve a low computational cost and good scalability, even with an increased problem size. In addition, MOPNAR was compared with other existing evolutionary algorithms such as GAR, EARMGA, GENAR, and MODENAR.
- *Multi-Objective Quantitative Association Rule Mining (MOQAR):* Martínez et al. [75] improved the multi-objective evolutionary algorithm (MOEA) non-dominated sorting genetic algorithm-II (NSGA-II) [28] by integrating it with their proposed QARGA approach. The authors used principal component analysis (PCA) to select the best subset of quality measures for the fitness function. Additionally, different distance criteria were introduced to replace the crowding

distance of solutions to obtain secondary rankings in Pareto fronts. The primary ranking was achieved through the non-dominated sorting of the solutions.

- *Multi-Objective Evolutionary Algorithm for Quantitative Association Rule Mining (MOEA-QAR):* The MOEA-QAR algorithm [86] combines a genetic algorithm with clustering to mine interesting association rules. The dataset is first clustered using K-means, and each cluster is used as input to a separate GA to extract rules for that cluster. The fitness function of each chromosome is defined by confidence, interestingness, and cosine2. The algorithm can be applied to the entire dataset or just to each cluster, and experiments show that more rules are retrieved per cluster than for the whole dataset. Notably, users do not need to specify minimum support or confidence thresholds.
- *Association Rule Mining with Differential Evolution (ARM-DE):* In 2018, Fister et al. [34] proposed a novel approach to ARM with numerical and categorical attributes based on differential evolution. Their algorithm consists of three stages: domain analysis, solution representation, and fitness function definition. In domain analysis, attribute domains are determined for numerical and categorical attributes. For numerical attributes, the minimum and maximum bounds are defined, while for categorical attributes, a set of values is enumerated. Each solution is represented mathematically using a real-valued vector. The fitness function is then calculated based on confidence and support, and optimization is achieved by maximizing the fitness function value.
- *Rare-PEAR:* The Rare-PEARs algorithm proposed by Almasi et al. [10] aims to discover various interesting and rare association rules by giving a chance to each rule with a different length and appearance. The algorithm decomposes the process of ARM into $N - 1$ sub-problems, where each sub-problem is handled by an independent sub-process during Rare-PEARs execution. N is the number of attributes, and each sub-process starts with a different initial population and explores the search space of its corresponding sub-problem to find rules with semi-optimal intervals for each attribute. This approach allows for a more comprehensive exploration of the search space, discovering more diverse and rare association rules.
- *Genetic Network Programming (GNP):* Taboada et al. [98] proposed Genetic Network Programming (GNP) as a graph-based approach to ARM with numerical attributes. GNP consists of three node types: a start node, a judgement node, and a processing node. The judgement nodes act as conditional branch decision functions, while the processing nodes act as action functions. Evolution is carried out using crossover and mutation operators, and the significance of important rules is measured using the chi-square test. Rules are stored in a pool, which is updated every generation, and the lower chi-squared value rule is exchanged with a higher chi-squared value rule. This approach effectively extracts important rules from the database.
- *Grammar-Guided Genetic Programming Association Rule Mining (G3PARM):* Luna et al. [72] applied Grammar-Guided Genetic Programming (G3P) to the task of finding QARs, building on their previous work in 2010, where they introduced G3PARM for ARM. The focus of the approach proposed in [72] was to reduce gaps in numerical intervals and emphasize the distribution of instances. To achieve this, the authors developed a self-adaptive algorithm that dynamically adjusts the number of parameters used in the evolutionary process and utilizes context-free grammar to represent solutions. The algorithm aims to identify the best rules according to a given fitness function, which are then stored in a pool and updated in each generation.
- *Multi-Objective Differential Evolution algorithm for Numeric Association Rules (MODENAR):* Alatas et al. [9] proposed a multi-objective differential evolution algorithm to discover accurate association rules from numeric attributes. The algorithm was designed to optimize

four objectives: amplitude, comprehensibility, support, and confidence, based on Pareto principles. The support and confidence of the discovered rules were required to be high. Comprehensibility was defined as the number of attributes involved in a rule, and shorter rules were preferred. The amplitudes of attribute intervals were aimed to satisfy fewer rules; hence amplitude was minimized while support, confidence, and comprehensibility were maximized.

Swarm Intelligence-Based Algorithms.

- *Rough Particle Swarm Optimization Algorithm (RPSOA):*
The RPSO algorithm was introduced as the first PSO-based algorithm for NARM with rough particles [7]. This algorithm aims to determine numeric attribute intervals and then discover association rules that conform to these intervals, where the fitness function is responsible for determining the amplitude of the intervals. Rough values of each attribute are defined by upper and lower bounds and are useful in representing an interval for an attribute. Each rough particle has decision variables representing items and intervals. It consists of three parts: the first describes the antecedent or consequent of a rule, the second represents the lower bound, and the third represents the upper bound of the interval. An item is considered an antecedent if its value is between 0 and 0.33, a consequent if it is between 0.33 and 0.66, and if it is between 0.66 and 1.0, the item would not be included in the rule. Once the RPSO algorithm completes its execution, attribute bounds refinement is performed for the covered rule. This refinement step aims to improve the quality and accuracy of the discovered association rules by further optimizing the attribute bounds.
- *Chaotically ENcoded Particle Swarm Optimization Algorithm (CENPSOA):* The CENPSOA algorithm, proposed by Alatas and Akin [8], introduced the use of chaos variables and particles in PSO for the first time. Unlike previous PSO-based methods, CENPSOA employs chaotic numbers to encode particle information. Specifically, each chaotic number mid_{rad} represents an interval with a lower bound of $mid - rad$ and an upper bound of $mid + rad$. In CENPSOA, a particle is represented as a string of chaotic parameters consisting of a midpoint and radius pair. Each decision variable consists of three parts: the first part represents the antecedent or consequent, the second part describes the midpoint and the third part represents the radius. This algorithm works similarly to RPSOA but is different only with the encoding of particles.
- *Parallel PSO for Quantitative Association Rule Mining (PPQAR):* Yan et al.[107] parallelized the PSO algorithm for ARM to increase its scalability and efficiency in dealing with large datasets in real-world applications. To evaluate each particle's quality, the suggested technique used four optimization objectives: support, confidence, comprehensibility, and interest. The parallel PSO method employs two techniques to handle distinct application scenarios: particle-oriented and data-oriented. The particle-oriented technique is well-suited for small datasets with a large number of particles, treating each particle as a separate computing unit and computing the fitness function in parallel. On the other hand, the data-oriented approach is suitable for large datasets, dividing the entire dataset into partitions and treating each partition as a computing unit. Unlike the particle-oriented method, the data-oriented method updates particle locations, velocities, and local best sets in parallel. Both methods were compared with the benchmark serial algorithm.
- *Multi-Objective Particle swarm optimization algorithm for Association Rules mining (MOPAR):* The MOPAR algorithm, proposed by Beiranvand et al. [20], is a multi-objective particle swarm optimization (MOPSO) technique based on Pareto optimality. It aims to extract numerical association rules in a single step using three objectives: confidence, comprehensibility, and

interestingness. Like RPSOA, the particle in MOPAR is represented by lower and upper bounds of intervals for each attribute. To address the problem of numerical ARM, MOPAR provides a redefinition of lbest and gbest particles and a selection procedure. The algorithm was compared with other multi-objective ARM algorithms, including MODENAR, MOGAR, RPSOA, and GAR.

- *PSO with the Cauchy Distribution (PARCD):* A method proposed by Tahyudin et al. [99] extends the MOPAR algorithm by combining PSO with the Cauchy distribution. In traditional PSO, the velocity of a particle approaches 0 after many iterations, leading to premature searching and suboptimal results. The proposed approach addresses this issue by integrating the Cauchy distribution in the velocity equation, allowing particles to continue exploring the search space. This method uses multiple objectives, including support, confidence, comprehensibility, interestingness, and amplitude functions, to extract numerical association rules in a single step. To evaluate the method's performance, it was compared with MOPAR, MODENAR, MOGAR, and RPSOA on various datasets, and it was found that the proposed method, called PARCD, outperformed MOPAR.
- *Wolf Search Algorithm (WSA):* Agbehadji [3] introduced a wolf search algorithm for NARM inspired by the hunting behaviour of wolves. The algorithm is based on three stages of wolf-preying behaviour: actively seeking prey, passively seeking prey, and escaping from predators. The algorithm generates association rules if the wolf is actively seeking prey, and no rules are generated if the wolf is passively seeking prey or escaping. The fitness function includes support, confidence, the number of attributes, and the penalization of interval frequency. The algorithm represents rules using the wolf's best position and fitness value, and each wolf's position contains decision variables for items and intervals. While this study introduces the algorithm, it has not been evaluated on datasets, and the algorithm's accuracy and efficiency will be determined in future work.
- *Multi-objective Particle Swarm Optimization (MOPSO):* The MOPSO algorithm, originally proposed by Coello [27] in 2004, utilizes Pareto dominance and an archive controller. In 2019, Kuo et al. [61] developed a MOPSO algorithm for NARM consisting of three stages: initialization, adaptive archive grid, and PSO searching. Particle representation and initialization are the same as in the RPSOA algorithm. The adaptive archive grid is a hypercube-shaped space designed to obtain non-dominated solutions by comparing all particle solutions using Pareto optimality. It contains two components: the archive controller and the grid. The external archive retains non-dominated solutions, and new solutions are added if existing ones do not dominate them or if the external archive is empty, the new solution is saved in the external archive; otherwise, it is discarded. The adaptive grid approach is used when the external population reaches its maximum capacity. The objective function space is partitioned into regions. The grid is recalculated if the external population's individual falls outside the grid's bounds, and each individual within it must be relocated. After the archive grid stage, PSO searching occurs. The algorithm also utilizes three objectives: confidence, comprehensibility, and interestingness to generate rules.
- *Ant Colony Optimization for Continous attributes (ACO_R):* The ACO_R algorithm, introduced by Moslehi and Eftekhari [88], is an ant colony optimization technique designed to discover association rules for numeric attributes without relying on minimum support and confidence thresholds. Unlike the ACO algorithm, which uses a discrete probability distribution, ACO_R employs a probability density function. It employs a solution archive size of k to describe the pheromone distribution over the search space, instead of a pheromone table. The algorithm works by having the ants move across the archive, selecting a row based on its associated weight (ω). Then a new solution is created by sampling the Gaussian function g for each

dimension's values in the selected solution. Each numeric attribute corresponds to one dimension of the solution archive, which is divided into three sections that make up a numeric association rule: the first part represents the rule's antecedent or consequence; the second part represents its value; and the third part represents its standard deviation, which is used to form numeric attribute intervals.

The algorithm uses Gaussian functions to determine the attribute intervals that correspond to interesting rules, with the function controlling the intervals' frequency and length. The objective function has four components. The first section, which can be viewed as the rule's support, measures the importance of the association rule. The second section is the confidence value of the rule. The third section is the number of attributes, while the last section penalizes the amplitude of the intervals that comply with the itemset and rules. The pheromone update technique introduces a set of new solutions, each generated by one ant, and eliminates the same number of bad solutions from the archive after ranking them to track the solutions. This ensures that the top-ranked solutions are always at the archive's top, and that the best solution in each execution of ACO_R is a rule.

- *Multi-Objective Cuckoo search Algorithm for Numerical Association Rule Mining (MOCANAR):* MOCANAR [49] is a multi-objective cuckoo search algorithm that uses Pareto principles to derive high-quality association rules from numeric attributes. The algorithm mimics the brooding parasitic behavior of cuckoo species and represents ARM using a $2D$ array. The columns of the array represent the attributes in the dataset, and the first row among three rows represents the attribute's location. The second row consists of the lower bound of the attribute, and the third row represents the upper bound of the attribute. A value of 0 in the first row indicates that the related attribute is not present in the rule, 1 shows that the attribute belongs to the antecedent part of the rule, and 2 shows that the attribute belongs to the consequent part of the rule. MOCANAR considers four objectives: support, confidence, interest, and comprehensibility. The algorithm was evaluated on three datasets and produced a small number of high-quality rules incrementally for each iteration of the method.
- *Multi-Objective Bat Algorithm for Numerical Association Rule Mining (MOB-ARM):* Heraguemi et al. [44] proposed a multi-objective bat algorithm for NARM inspired by microbats' behaviour. The algorithm uses four quality measures, namely support, confidence, comprehensibility, and interestingness, and two global objective functions to extract interesting rules. The first objective function combines support and confidence, while the second objective function considers comprehensibility and interestingness. The algorithm comprises three main steps: initialization, searching for the non-dominance solution for the Pareto point, and searching for the best solution for each bat at the Pareto point. The rule is encoded using the Michigan approach. The bats are initialized with random frequency and velocity, and the proposed algorithm is also compared with other algorithms, including MODENAR, MOGAR, and MOPAR.
- *Discrete Crow Search Algorithm for Quantitative Association Rule Mining (DCSA-QAR):* In 2021, a new algorithm called DCSA-QAR was proposed for mining numerical association rules [63]. This approach utilizes a novel discretization algorithm called Confidence-based Unsupervised Discretization Algorithm (CUDA) that employs the confidence measure to discretize numerical attributes. The CSA is then transformed from continuous to discrete using crow position encoding, and new operators are used to ensure that any position update within the search space is valid. Each crow in the flock is represented by its current position and memory positions, with each particle composed of two vectors for control and parametric attributes. The control attributes can have one of three values: 0 indicates that the attribute is not part of the rule, 1 indicates that it belongs to the antecedent, and −1 indicates that it

is part of the consequent. The fitness function is optimized by maximizing the measures of support, confidence, and gain of the rules. DCSA-QAR was compared with several mono and multi-objective algorithms, including NICGAR, MOPNAR, MODENAR, and MOEA-Ghosh.

Physics Based Algorithms.

- *Gravitational Search Algorithm for NARM (GSA-NARM):* GSA is a physics-inspired metaheuristic that leverages Newton's law of gravity. In the context of NARM, the GSA algorithm, as described by Can et al. [24], aims to discover attribute intervals simultaneously without needing a minimum support or confidence threshold. In GSA-NARM, agents are treated as objects, and their positions represent potential solutions. The objective function determines the amplitude of the intervals being explored. The algorithm identifies the position of the agent with the heaviest mass as the global solution, analogous to the gravitational force exerted by massive objects in Newton's law. During the optimization process, the fitness function is evaluated for each agent, and the gravitational constant, denoted as G, is updated based on the performance of the best and worst agents in the population. The mass M of each agent is computed, and the velocity and position are updated accordingly, mimicking the motion of celestial objects influenced by gravitational forces. The GSA-NARM algorithm continues iterating until the stopping criteria are met, such as reaching a maximum number of iterations or achieving a desired fitness value. The algorithm then returns the association rule with the best fitness value obtained during the optimization process. GSA-NARM has demonstrated promising results when compared to other state-of-the-art methods for NARM, showcasing its effectiveness in tackling the NARM problem.

Algorithm for Hybrid Based.

- *Hybrid Genetic PSO-Quantitative Association Rule Mining (HGP-QAR):* Moleshi et al. [87] introduced a hybrid approach called HGP-QAR, which combines the strengths of multi-objective GA and multi-objective PSO methods. By leveraging the advantages of both techniques, HGP-QAR aims to improve the efficiency of NARM. The hybridization of GA and PSO allows for the exploration of the search space from different perspectives. In HGP-QAR, individuals are represented as chromosomes for GA and particles for PSO. The individuals are sorted based on a fitness function that considers three metrics: confidence, interestingness, and comprehensibility. During the optimization process, the upper half of individuals follow the stages of GA, including selection, crossover, and mutation, while the lower half follows the stages of PSO, updating their velocity and positions based on the personal best (pbest) and global best (gbest) positions. This combination of GA and PSO allows for a more efficient search and exploration of the solution space. The outcomes obtained from GA and PSO are then combined to generate the next generation and form new rules. This process is repeated until the termination criteria are met, such as reaching a maximum number of iterations or achieving satisfactory results. Various experimental results show that the hybrid GA-PSO approach, HGP-QAR, outperforms other algorithms like MOPAR and PARCD in terms of efficiency, demonstrating its effectiveness in NARM.
- *Multi-objective Hybrid Differential Evolution Sine Cosine Numerical Association Rule Mining Algorithm (MOHDESCNAR):* DE has been known to suffer from premature convergence and stagnation issues in multi-modal search spaces. A recent approach called MOHDESCNAR [12] has been proposed to overcome these problems. This algorithm reduces the number of numerical association rules by adjusting the intervals of related numeric attributes. It employs hybrid sine and cosine operators with DE, which can overcome stagnation issues. The proposed algorithm balances exploration and exploitation by using global DE exploration

and local SCA exploitation during iterations to prevent premature convergence and stagnation problems. This study used three methods: using only the sine operator (MOHDESNAR), only the cosine operator(MOHDECNAR), and both the sine and cosine operators (MOHDESCNAR).

- *Quantitative Association Rule miner with Chaotically Encoded Hybrid Differential Evolution and Sine Cosine Algorithm (QARCEHDESCA):* Altay and Alatas proposed the MOHDESCNAR algorithm in 2021, which used a combination of DE and the sine and cosine algorithms. In 2022, the authors introduced a new hybrid algorithm called QARCEHDESCA [13], which employs chaos number-based encoding and HDESCA (Hybrid differential evolution sine cosine algorithm). The QARCEHDESCA algorithm dynamically discovers the ranges of quantitative attributes and association rules. It randomly initializes candidate search agents to find quantitative associations. The initial set of search agents is removed from all-dominating search agents. The remaining nondominated search agents are sent to SCA-based new operators and DE crossover. The nearest neighbour distance function is used to remove rules close to each other when the count of nondominated rules exceeds the defined threshold. For QARCEHDESCA, the best search agent and one random agent are chosen for sine and cosine operators. After that, DE's crossover operator is applied to nondominated search agents. If the trial agents dominate the target search agent, it is added to the population; otherwise, the search agent with the highest weighted sum fitness is chosen for subsequent iterations. When the maximum number of iterations is reached, QARCEHDESCA returns nondominated QARs. The fitness function of the algorithm aims to maximize support, confidence, and comprehensibility while minimizing attribute amplitudes. Each search agent represents a numerical association rule with two components: inclusion/exclusion and a chaotic number representing the center point and radius. QARCEHDESCA is compared with RPSOA and other intelligent optimized algorithms.

4.2.3 *The Statistical Method.*

- *Aumann and Lindell's Work:* Aumann and Lindell [17] introduced a new definition of QARs based on the distribution of values of quantitative attributes and presented an algorithm to mine them. To consider the distribution of continuous data, they used conventional statistical measures.
- *Webb's Work:* Aumann and Lindell's approach has the disadvantage of being impractical for generating frequent itemsets in dense data. To address this limitation, Webb proposed an efficient admissible unordered search algorithm for discovering impact rules, which capture meaningful interactions between data selectors and numeric variables in dense data [106]. Impact rules were introduced as a new name for QARs. The proposed OPUS_IR algorithm uses the OPUS framework and does not need to retain all frequent itemsets in memory during frequent itemset generation, unlike Aumann and Lindell's method [17]. It also does not require a minimum cover to be specified for the search. The OPUS_IR algorithm was compared with the frequent itemset approach in terms of performance.
- *Kang et al. Work:* The authors of the study [50] introduced a new approach to bipartition quantitative attributes called *standard deviation minimization*. This technique minimizes the standard deviation of two partitions obtained by dividing the attribute into two parts, and it outperforms existing bipartition techniques. The authors also redefined the mean-based and median-based bipartition techniques, and their experimental results confirmed the effectiveness of the proposed framework.

4.2.4 *Miscellaneous Other Methods.*

- *Mutual Information and Clique (MIC) Framework:* Yiping et al. [56] proposed a novel approach for mining QARs using an information-theoretic framework called MIC. This framework avoids the generation of excessive itemsets by investigating the relationship between attributes. The approach comprises three phases: 1) discretization, which partitions numeric attributes into intervals; 2) MI graph construction, which computes the normalized mutual information of attributes and represents their strong relationships using a MI (mutual information) graph; and 3) clique computation and QAR generation, which computes frequent itemsets using cliques and generates QARs. The experiments demonstrate the effectiveness of the MIC framework in reducing the number of generated itemsets and improving the efficiency of QAR mining.
- *Generalized One-sided Quantitative Association Rule mining (GOQR) and Non-redundant Generalized One-sided Quantitative Association Rule mining (NGOQR):* Zhiyanget al.[47] proposed a cognitive computing-based method for NARM, consisting of two algorithms: *GOQR* and *NGOQR*. These algorithms consider the order relation of attribute values when mining rules. The first phase of the *GOQR* algorithm generates frequent itemsets, while the second phase extracts generalized one-sided QARs. To enhance efficiency, the rules are reduced using a generalized one-sided concept lattice. For non-redundant rule extraction, the *NGOQR* algorithm first executes the minimal generator of a target itemset algorithm and then continues with rule mining.
- *Quantitative Miner with the VMO algorithm (QM_VMO):* The Quantitative Miner with the VMO algorithm *(QM_VMO)* [48] utilizes the Variable Mesh Optimization algorithm [90], a population-based meta-heuristic. The algorithm represents the population P as a mesh of n nodes $P = n_1, n_2, ..., n_n$, where each node corresponds to a possible solution and consists of an m-dimensional vector $n_i = (v_1^i, v_2^i, ..., v_m^i)$. The algorithm primarily operates through expansion and contraction processes. *QM_VMO* is executed in three stages: (i) defining a rule template, (ii) generating the rule population, and (iii) optimizing the numerical attributes of the rule by optimizing intervals. Compared with QuantMiner, *QM_VMO* is found to be less sensitive to changes in the dataset.

4.3 RQ3 What are the advantages and limitations of the existing NARM methods?

There are strengths and limitations associated with each method for NARM. These advantages and limitations of each approach are summarized in Tables 9, 10, and 11.

Discretization methods are advantageous in terms of simplicity, interpretability, and flexibility. They allow for the handling of both categorical and numerical attributes, and the resulting discrete intervals can be easily understood and applied. However, these methods require the specification of a user-defined threshold, which can be subjective and may affect the quality of the discovered rules. Discretization can also lead to information loss and may not capture the true underlying patterns in the data.

Optimization methods excel in their ability to discover relationships and patterns in high-dimensional data without the need for user-defined thresholds or discretization steps. They can handle both categorical and numerical attributes and are often more robust to noise and missing data. However, these methods can suffer from issues such as convergence problems, finding only local optima, high computational complexity, and the need for a large amount of computational resources.

Statistical methods are advantageous in their ability to handle missing data and noise. They are also well-suited for analyzing categorical data and can provide statistical significance measures for the discovered rules. However, these methods are typically designed for categorical data and may not be suitable for numerical attributes. They often assume linear relationships and may not capture more complex patterns present in the data. Overall, each approach has its strengths and limitations, and the choice of method depends on the specific requirements and characteristics of the dataset being analyzed.

Table 9. Advantages and Limitations of Discretization Method

Approaches	Reference	Advantages	Limitations
Partitioning	[96]	Simple and easy to implement.	Adjusting minimum support and minimum confidence.
	[25]	Discover both +ve and -ve rules. Avoid user-specified threshold.	need to adjusted difference analysis
Clustering	[65]	The ARCS system scales better than linearly with data size.	Algorithm is sensitive to noise. Not used for high dimensional data.
	[104]	Scalable for very large databases. Generate only non-overlapping intervals.	Only generate the rules where consequent should be categorical attribute
	[70]	Reflects all the possible interdependencies between attributes in data sets	Requirement of the user-specified threshold.
	[67]	Efficiently identify a small set of subspaces for finding dense regions. In each cover searching is limited.	Need to specify many thresholds. Performance is poor on more than 10 dimensions. The dimensionality curse problem is unsolved. The algorithm does not perform well for the data set with uniform density.
	[40]	Effective and scale up linearly with an increased number of attributes.	Minimum threshold is needed.
	[109]	There is no need to scan the database many times. Do not generate many candidate units. Histogram H' saves the calculation time of support of each grid.	As the number of transactions increases, the run time also increases.
Fuzzy	[113]	Prune less interesting rules.	
	[46]	Accuracy increased as the number of transactions increased.	Membership functions should be known in advance.
	[114]	Optimized the fuzzy sets. The frequent itemsets are created through a two-level iteration process. Flexible membership function.	
	[105]	Provide better clustering than other methods.	

Table 10. Advantages and Limitations of Optimization Method

Approaches	Reference	Advantages	Limitations
Evolution and DE	[79]	Find association rules from numeric dataset without discretization	
	[80]	Find the amplitude of the intervals by fitness function.	Only frequent itemsets are generated.
	[14]	Find association rules from numeric and categorical without discretization	
	[6]	discover both positive and negative rules.	
	[108]	High-performance association rule mining, System automation, no need for user-specified minimum support threshold	
	[76]	Low run time, discover diverse, both positive and negative rules.	
	[75]	Low computational cost and good scalability	
	[86]	No need to determine the minimum support and minimum confidence.	
	[34]	Capable of dealing with numerical and categorical attributes.	The algorithm is unable to shrink the lower and upper borders of the numerical attributes.
	[9]	association rules are mined without generating frequent itemsets. The algorithm is easy to implement and independent from the requirement of minimum support and minimum confidence threshold.	DE suffers from stagnation and premature convergence problem and its local exploitation capability is weak.
Swarm-Intelligence	[107]	Efficient and scalable to process huge dataset.	PSO trap in local optima.
	[20]	Prevent generation of huge useles rules. No requirement for minimum support and minimum confidence threshold.	Low support values for association rules.
	[99]	Increase the global optimal value of expanded search space.	
	[88]	No need for minimum support and minimum confidence threshold.	Variable correlations are not differentiated by ant algorithms.
	[49]	Provide better support and confidence.	Higher number of extracted rules decreases the interpretability of the results.
	[44]	Reduces computation time.	

Table 10. Advantages and Limitations of Optimization Method

Approaches	Reference	Advantages	Limitations
Physics-based	[24]	The confidence and support values of the automatically mined rules are very high. No prior requirement for minimum support and confidence threshold. The problem of attribute interactions has been solved.	Not very efficient in searching.
Hybrid	[13]	Efficient with respect to the mean number of rules, mean confidence, and mean size metrics.	Does not provide the higher mean support value.

Table 11. Advantages and Limitations of Statistical Method

Reference	Advantages	Limitations
[17]	Able to handle large datasets. Adaptable and may be used for various types of data and user needs.	Computationally expensive. Assumptions-dependent and limited interpretability.
[106]	Its unordered search quality makes it handle different types of data. High performance. Identify high-impact patterns and relationships.	High computation cost for large datasets. Limited interpretability.
[50]	Accuracy of ARM can be improved by identifying patterns and relationships within particular subsets of the data. Able to handle large datasets.	Difficult to choose the right threshold. Loss of information could be possible.

4.4 RQ4 Which objectives are considered by the several existing multi-objective optimization NARM algorithms?

Optimization problems are prevalent and important in scientific research. They can be categorized into two types based on the number of objective functions: single-objective and multi-objective optimization problems. In NARM, the most commonly used parameters are support and confidence, making NARM algorithms single-objective optimization methods where a single solution is selected based on the user's requirements. On the other hand, multi-objective optimization problems involve computing multiple objective functions simultaneously, which can conflict with each other. A solution that works well for one function may be ineffective for another. This makes finding a single solution that satisfies all objectives difficult, and instead, a set of Pareto-optimal solutions is obtained that trade-off between the competing objectives. Table 12 lists the objectives considered in multi-objective optimization NARM studies, and Table 13 provides the names of algorithms that utilize these objectives. A detailed explanation of all these objectives is given via Eq. 5–13.

Support: The number of records with both X and Y itemsets determines the rule's support count. |D| is the total number of records in a dataset.

Table 12. List of Objectives for Multi-objective Optimization Algorithm for NARM

Objectives	References
Confidence	[9, 10, 12, 13, 20, 44, 49, 61, 63, 75, 83, 86–88, 99, 107]
Support	[9, 12, 13, 44, 49, 63, 88, 99, 107]
Interestingness	[10, 20, 44, 49, 61, 77, 78, 83, 86–88, 99, 107]
Comprehensibility	[9, 12, 13, 20, 44, 49, 61, 77, 78, 83, 87, 99, 107]
Amplitude	[9, 12, 13, 88, 99]
Performance	[77, 78]
Accuracy	[10, 75]
Leverage	[75]
Gain	[63]
Cosine	[86]

Table 13. List of Multi-Objective Algorithms for NARM

Algorithms	Reference	Objectives
MOGAR	[83]	Confidence, Comprehensibility, Interestingness
QAR-CIP-NSGA-II	[78]	Comprehensibility, Interestingness, Performance
MOPNAR	[77]	Comprehensibility, Interestingness, Performance
MOQAR	[75]	Accuracy, Leverage, Confidence
MOEA-QAR	[86]	Confidence, Interestingness, Cosine
Rare-PEAR	[10]	Interestingness, Accuracy, Confidence
MODENAR	[9]	Support, Comprehensibility, Confidence, Amplitude
MOHDESCNAR	[12]	Support, Comprehensibility, Confidence, Amplitude
PPQAR	[107]	Support, Confidence, Comprehensibility, Interestingness
MOPAR	[20]	Confidence, Comprehensibility, Interestingness
PARCD	[99]	Support, Confidence, Comprehensibility, Interestingness, Amplitude
MOPSO	[61]	Confidence, Comprehensibility, Interestingness
ACO_R	[88]	Support, Confidence, Interestingness, Amplitude
MOCANAR	[49]	Support, Confidence, Interestingness, Amplitude
MOB-ARM	[44]	Support, Confidence, Comprehensibility, Interestingness
DCSA-QAR	[63]	Support, Confidence, Gain
QARCEHDESCA	[13]	Support, Comprehensibility, Confidence, Amplitude
HGP-QAR	[87]	Confidence, Comprehensibility, Interestingness

$$Support(X \Rightarrow Y) = \frac{|(X \cup Y)|}{|D|} \tag{5}$$

Confidence: The confidence metric assesses the quality of a rule by counting the number of times an AR appears in the entire dataset. The following equation 6 is used to compute the confidence of the rule $X \Rightarrow Y$. Furthermore, these parameters do not ensure that significant rules will be generated.

$$Confidence(X \Rightarrow Y) = \frac{|(X \cup Y)|}{|X|} \tag{6}$$

Interestingness: The interestingness of a rule is a metric for determining how surprising a rule is to users, not just all possible rules. The first component of Eq.(7) relates to the probability of producing the rule based on the antecedent part, the second part relates to the probability of producing rules based on the consequent part, and the last component is the probability of producing rules based on the overall dataset.

$$Interestingness = \frac{Support|(X \cup Y)|}{Support|X|} \cdot \frac{Support|(X \cup Y)|}{Support|Y|} \cdot \left(1 - \frac{Support(X \cup Y)}{Support(X)}\right) \tag{7}$$

Comprehensibility: The number of attributes included in both the antecedent and consequent parts of the rule is measured by comprehensibility [38]. If the generated rules contain more attributes, then the rules will be difficult to comprehend. The rule is more comprehensible if the number of conditions in the antecedent part is less than that in the consequent part. The following expression measures the comprehensibility of an association rule:

$$Comprehensibility = \frac{\log(1 + |Y|)}{\log(1 + |X \cup Y|)} \tag{8}$$

Where $|Y|$ and $|X \cup Y|)$ represent the number of attributes in the consequent part and both parts.

Amplitude: The intervals in each attribute that comply with interesting rules must have smaller amplitudes. If two rules have the same number of rows and attributes, the one with smaller intervals will provide more information. Amplitude is a minimization function; however support, confidence and comprehensibility are maximization functions [9].

$$Amplitude\ of\ the\ Intervals = 1 - \frac{1}{m} \sum_{i=1}^{m} \frac{u_i - l_i}{max(A_i) - min(A_i)} \tag{9}$$

Performance: The product of support and CF is performance. Performance enables the ability to mine accurate rules with a suitable trade-off between local and general rules. This measure has a range of values between 0 and 1. The user may find a rule with a performance value close to 1 more useful.

Accuracy: Accuracy represents the veracity of the rule [37].

$$Accuracy(X \Rightarrow Y) = Support(X \Rightarrow Y) + Support(\neg X \Rightarrow \neg Y) \tag{10}$$

Leverage: Leverage is the difference between the frequency with which the antecedent and the consequent are identified together and the frequency with which they would be expected to be observed together, given their individual support [89]. It represents the strength of the rule.

$$Leverage(X \Rightarrow Y) = Support(X \cup Y) - Support(X) \cdot Support(Y) \tag{11}$$

Gain: Gain is the difference between the confidence of both the antecedent and consequent part and the support of the consequent part [37].

$$Gain(X \Rightarrow Y) = confidence(X \Rightarrow Y) - Support(Y) \tag{12}$$

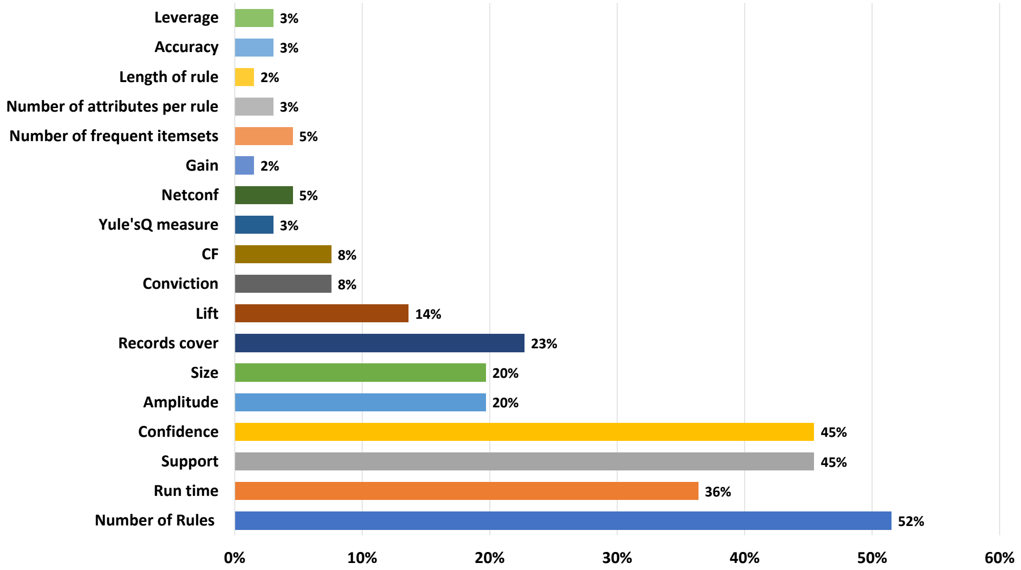


Fig. 1. Metrics Used to Evaluate NARM Algorithms.

Cosine: The cosine measure considers both the pattern's interest and its significance [101].

$$Cosine(X \Rightarrow Y) = \frac{Support(X \cup Y)}{\sqrt{Support(X) \cdot Support(Y)}} \tag{13}$$

4.5 RQ5 What are the metrics to evaluate the NARM algorithms?

This RQ aims to identify the commonly used evaluation metrics in NARM algorithms. As shown in Figure 1, the number of rules is the most commonly used metric, followed by support, confidence, and run time. Only a few algorithms use other metrics, such as Yule'sQ measure (3%), leverage (3%), accuracy (3%), gain (2%), length of rule (2%), and the number of attributes per rule (3%). Interestingly, all methods use the number of rules, run time, support, and confidence as evaluation metrics, while only the optimization method employs all metrics (see Figure 2). Some papers on discretization methods use the number of frequent itemsets as a metric [29, 42]. The discretization method primarily uses run time metric with different parameters, such as over the number of records or buckets [23, 36, 40, 42, 62, 65, 67, 82, 109], minimum support, and minimum confidence [22, 23, 29, 36, 95, 114] over the number of buckets [23], number of sparse points, number of dense regions and number of attributes [67]. On the other hand, the statistical method primarily uses run time [17, 106] and a number of rules [17, 50] as evaluation metrics. However, other methods also use minimum support, and confidence except for run time and number of rules [47, 48, 56]. Mean of interest of missing QARs and variance of interest of missing QARs, the maximum interest of missing QARs were also used for performance evaluation in [56]. Of the 34 papers (52%) that employ the number of rules as an evaluation metric, 24 papers (36%) belong to the optimization method. However, 24 publications (36%) in all have evaluated the NARM algorithms regarding the execution time, among which 15 (23%) articles are from the discretization method.

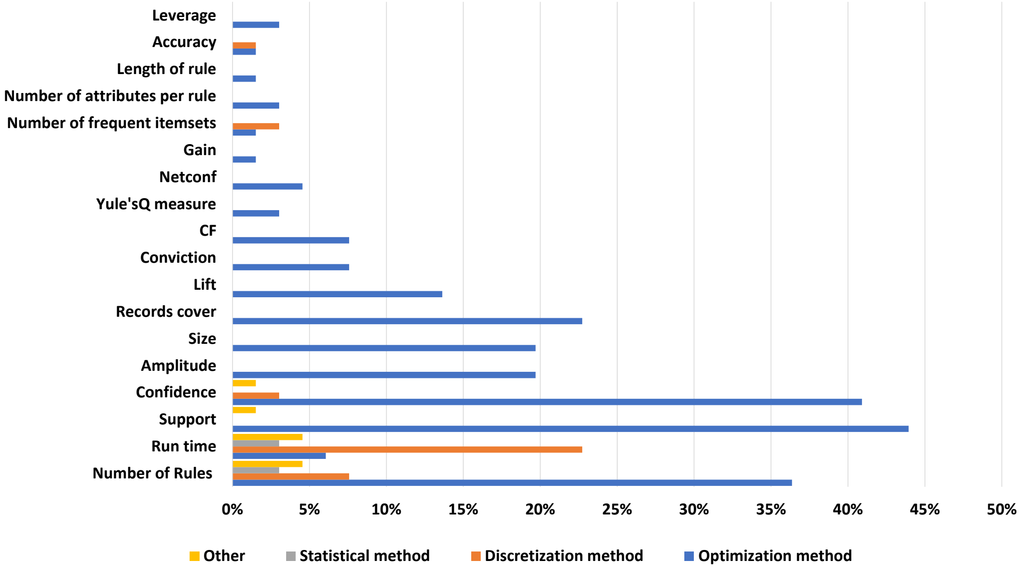


Fig. 2. Distribution of Metrics Used in NARM Methods.

4.6 RQ6 Which datasets are used for experiments by NARM methods?

Different NARM methods may use different datasets for their experiments, depending on the method type and application domain. Table 14 presents the datasets that were most commonly used by different NARM methods, excluding those that were used in only one or two articles. In total, we considered twenty-two datasets, including both real-world and synthetic ones. Fifteen of these datasets were sourced from the Bilkent University Function Approximation Repository (BUFA) [41], while seven were from the University of California Irvine machine learning repository (UCI) [68]. Figure 3 shows the datasets used by NARM methods.

The *Quake* dataset was used more frequently than any other dataset, followed by *Basketball*, *Bodyfat*, *Bolts*, and *Stock Price*. Synthetic datasets were also commonly used. Table 15 lists the datasets that were used specifically for the discretization method, which were mostly different from those used by other methods. In total, we considered seventeen datasets, including both real-world and synthetic ones. Most articles on the discretization method used various real-world datasets. As shown in Figure 4, most of these datasets were used in only one article. We also observed that the optimization method articles tended to use datasets from the BUFA repository, while the discretization method articles tended to use datasets from the UCI repository.

4.7 RQ7 What are the challenges and potential future perspectives for the area of NARM?

To address this research question, a manual identification of the existing research challenges in NARM was conducted. Additionally, the focus was placed on identifying future directions for NARM research.

4.7.1 Research Challenges. After a comprehensive analysis of various NARM methods in both static and dynamic settings, we have identified several issues that NARM needs to address.

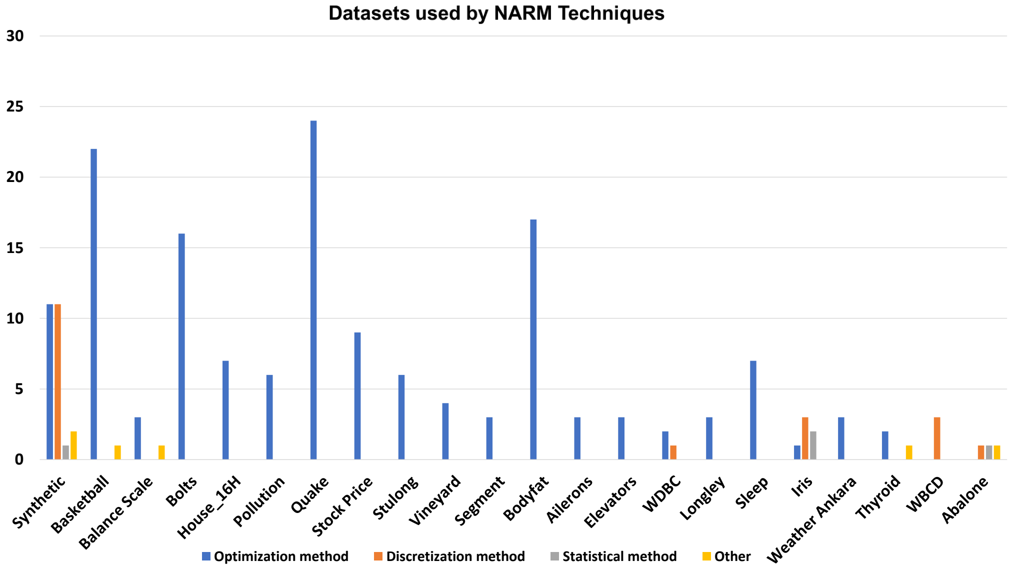


Fig. 3. Datasets Used by NARM Techniques.

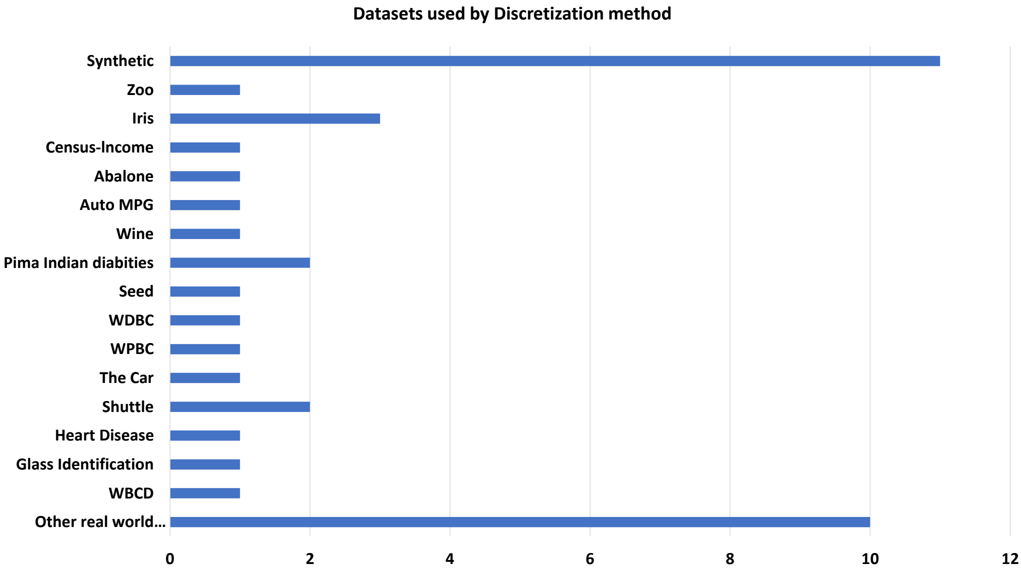


Fig. 4. Datasets Used by Discretization Method.

- *Handling Skewed Data:* NARM faces challenges when dealing with skewed data, where the data distribution is uneven. Finding associations between numerical variables in such datasets can be difficult and lead to biased results, as well as a high number of irrelevant rules. Moreover, skewed data can have a negative impact on the accuracy and reliability of the

Table 14. Data-sets Used by NARM Algorithms (WBC=Wisconsin Breast Cancer Data; WDBC = Wisconsin Diagnostic Breast Cancer) According to Ulitized Methods (Opt.=Optimization; Disc.=Discretization; Stat.=Statistical, Etc.=Other).

Source	Dataset	Methods	References
BUFA	Basketball	Opt. Etc.	[6–10, 13, 20, 24, 44, 48, 49, 61, 63, 73, 75, 76, 78, 80, 83, 86–88, 99]
	Balance Scale	Opt. Etc.	[10, 47, 76, 78]
	Bolts	Opt.	[6, 7, 9, 10, 12, 13, 63, 73, 75–78, 80, 87, 88, 99]
	House_16H	Opt.	[10, 12, 72, 75–78]
	Pollution	Opt.	[6, 7, 9, 10, 12, 13, 63, 73, 75–78, 80, 87, 88, 99]
	Quake	Opt.	[6–10, 12, 13, 20, 24, 44, 49, 61, 63, 73, 75–78, 80, 83, 86–88, 99]
	Stock Price	Opt.	[9, 10, 24, 73, 75–78, 80]
	Stulong	Opt.	[10, 12, 76–78, 93]
	Vineyard	Opt.	[12, 73, 75, 80]
	Segment	Opt.	[72, 76, 77]
	Bodyfat	Opt.	[7–9, 12, 13, 20, 44, 49, 61, 63, 73, 75, 80, 83, 86, 87, 99]
	Ailerons	Opt.	[12, 63, 75]
	Elevators	Opt.	[12, 63, 75]
	Longley	Opt.	[12, 73, 75]
	Sleep	Opt.	[6, 7, 9, 73, 75, 80, 88]
UCI	Iris	Opt. Disc. etc.	[47, 48, 66, 69, 93, 95]
	Weather Ankara	Opt.	[63, 72, 75]
	Thyroid	Opt. Etc.	[24, 56, 76]
	WBC	Disc.	[66, 69, 82]
	WDBC	Opt. Etc.	[72, 76, 114]
	Abalone	Disc. Stat. Etc.	[48, 95, 106]
	Synthetic	Opt. Stat. Etc.	[6–9, 14, 22, 36, 48, 50, 56, 62, 64, 65, 67, 70, 74, 79, 80, 88, 92, 93, 96, 98, 104, 113]

analysis, potentially resulting in biased conclusions. Calculation of support and confidence measures can be particularly affected, leading to inaccurate values and erroneous assessments of rule strength. Furthermore, processing skewed data can also impact the speed and efficiency of the algorithms, as they may need to handle a large number of extraneous rules.

- *Handling a Large Number of Rules:* The main objective of mining numerical association rules is to discover relationships between numerical variables in large datasets. However, this often results in a vast number of association rules, which can make the process computationally expensive, time-consuming, and difficult to sift through to identify the most relevant or interesting rules. To address this challenge, several techniques have been developed to simplify the process and make it more manageable. Some of these techniques include data sampling, the use of efficient algorithms, parallel and distributed computing, dimensionality reduction, and pruning methods. By implementing these techniques, it becomes easier to extract useful association rules from large datasets, reduce the size of the dataset, simplify the mining process, and speed up computations.
- *Quality of Association Rules:* Extracting high-quality rules is also a challenge in NARM due to the potential for redundancy, irrelevance, and conflicts in the rules. The large and complex

Table 15. Data-sets Used by Discretization Method for NARM Algorithms

Datasets	References
Wisconsin Breast Cancer Data	[66, 69, 82]
Wisconsin Prognostic Breast Cancer	[114]
Wisconsin Diagnostic Breast Cancer	[114]
Glass Identification	[66]
Heart Disease	[66]
Shuttle	[40, 109]
The Car	[40]
Seed	[69]
Pima Indian diabities	[95, 114]
Wine	[69]
Auto MPG	[95]
Abalone	[95]
Census-lncome	[25]
Iris	[66, 69, 95]
Zoo	[25]
Synthetic	[22, 36, 62, 64, 65, 67, 70, 92, 96, 104, 113]
Other real-world datasets	[23, 26, 29, 33, 42, 46, 70, 81, 84]

nature of the datasets used in NARM can lead to a large number of rules, making it difficult to identify the most relevant ones. Additionally, skewed data can impact the reliability of support and confidence measures, further affecting the quality of the rules. The rules generated by NARM algorithms may also be challenging to interpret and understand. To address these issues, data pre-processing, the use of alternative metrics, the selection of appropriate algorithms, and the application of ensemble methods can be helpful in improving the quality of association rules.

- *Complex Relationship:* Numerical data often contain intricate relationships, such as non-linear or multi-dimensional relationships, which can be difficult to represent and analyze using traditional ARM algorithms. This may result in inaccurate or incomplete rules, which can impact the reliability and accuracy of the analysis. To address this challenge, advanced algorithms such as decision trees, artificial neural networks, or support vector machines can be utilized in NARM. Ensemble approaches like gradient boosting or random forests can also be helpful in addressing complex relationships by combining the output of multiple algorithms to produce more accurate results. However, these techniques may increase computational complexity and require more data and computing resources to be effective.
- *Handling Outliers:* Outliers are extreme values that differ significantly from the majority of values in the dataset and can impact the accuracy and reliability of the results of ARM. Outliers may indicate genuine data variances, or they could result from measurement errors or data input issues. Several methods can address this problem, including outlier detection, data cleaning, data transformation, and robust algorithms. These methods can help remove or mitigate the effect of outliers, ensuring that the mining process yields more accurate and reliable results.

4.7.2 Future directions.

- *Handling Big Data:* Despite conducting a thorough SLR, we were unable to identify any studies that focus on retrieving numerical association rules from big data. However, the rise of big data will undoubtedly have a significant impact on the future of NARM. Developing more efficient algorithms that can handle vast amounts of numerical data will be essential as big data continues to become increasingly common. This will likely lead to the development of new algorithms specifically designed for big data that are optimized for scalability, speed, and accuracy. Additionally, advanced data cleanings and preprocessing techniques, such as outlier detection, imputation of missing values, and feature selection, will become increasingly important to ensure the quality of results.
- *Explainable AI:* Improving the interpretability and explainability of NARM results is a critical research direction. Explainable AI [16, 19] can enhance transparency and comprehension, which is essential for non-experts to validate the findings and ensure their alignment with the intended objectives. By revealing the underlying reasoning behind the results, Explainable AI can assist users in making better decisions and recognizing any inherent biases or limitations in the outcomes. Therefore, developing NARM techniques that provide transparent and comprehensible results is a vital area of research.
- *Hybrid Approach:* A promising future direction in NARM is to leverage the strengths of various methods and techniques through hybrid approaches. Some studies, such as [12, 13, 59, 84, 87, 105, 113], have attempted to combine different approaches to improve the results of NARM. Combining NARM with deep learning, integrating rule-based and distance-based approaches, and combining unsupervised and supervised learning are all hybrid approaches that show potential in this field. By integrating these techniques, the limitations of individual methods can be addressed, resulting in a more accurate and thorough analysis of the relationships between variables in numerical data. Hybrid approaches can lead to valuable insights and more reliable results.
- *Handling Streaming Data:* To keep up with the increasing demand for real-time analysis, developing NARM algorithms that can handle streaming data and update association rules in near real-time is crucial. In applications where timely and accurate decisions are critical, streaming data enables real-time analysis of numerical data, which allows organizations to make informed decisions based on up-to-date information. The ability to analyze a larger volume of numerical data in real time will lead to more comprehensive and accurate results. Moreover, streaming data enables dynamic updates to the results of NARM as new data becomes available, providing a more accurate and comprehensive view over time. Therefore, developing NARM algorithms that can handle streaming data is an important future direction.
- *Incorporating Machine Learning Techniques:* The integration of machine learning techniques, such as deep learning, into NARM, has the potential to revolutionize the field. With the ability to automatically detect patterns and relationships in the data, which may not be immediately apparent to human analysts, machine learning algorithms can significantly enhance the accuracy of the results. Moreover, this approach can reduce the time and effort required to identify such patterns in the data. The utilization of machine learning can also expand the scope of NARM applications across various industries, as these algorithms can handle complex data more efficiently, including high-dimensional data or data with non-linear relationships.
- *Privacy and Security:* The importance of privacy and security in NARM is increasing, and it is imperative to protect and use data ethically. However, the existing studies in this SLR did not address these issues. To ensure data protection, sensitive information can be removed or masked using anonymization techniques while preserving the necessary data for ARM. Furthermore, to reduce the risk of unauthorized access, the data can be partitioned into

smaller subsets, and access control methods can be developed to control who has access to the data and the association rules generated from it. Incorporating these privacy and security measures will safeguard the data and ensure its ethical use.

4.8 RQ8 How to automate discretization of numerical attributes for NARM in a useful (natural) manner?

Developing novel methods and techniques for NARM is a continuous area of research, and the discretization method serves as the foundation for NARM [96]. However, selecting the best partitions for discretizing complex real-world datasets still lacks a benchmark method. None of the discretization methods that we figured out with the review, see Sects. 4.1.1 in 4.2.1, explicitly addresses human perception of partitions. Therefore, we proposed a novel discretization technique in our research [51], which utilizes two measures for order-preserving partitioning of numerical factors: the *Least Squared Ordinate-Directed Impact Measure* (LSQM) and the *Least Absolute-Difference Ordinate-Directed Impact Measure* (LADM).

These proposed measures offer a straightforward method for finding partitions of numerical attributes that reflect best the impact of one independent numerical attribute on a dependent numerical attribute. We thoroughly experimented with these measures and compared the outcomes with human perceptions of partitioning a numerical attribute [55]. To develop an automated measure for discretizing numerical attributes, understanding perceptual conception is crucial. To achieve this, we investigated the impact of data points' features on human perception when partitioning numerical attributes [54]. These efforts have contributed to the development of more accurate and efficient methods for NARM.

5 THREATS TO VALIDITY

This section outlines potential threats to the validity of this SLR that might bias the outcomes of our in-depth investigation. The first threat pertains to defining the search string. We make no claims regarding the perfection of the search string used in the process. While we included all relevant search terms related to NARM, it is possible that the search terms may not have captured all relevant NARM-related work. To mitigate this risk, we included synonyms for "numerical association rule mining" and abbreviations such as "NARM" and "QARM" in the search term. The second threat pertains to the selection of digital libraries to search for articles. Although we searched five digital libraries for computer science, it is possible that additional sources may have produced different outcomes. To minimize bias, we manually searched Google Scholar and also looked through the list of references for the selected primary studies to identify significant publications. We are confident that the majority of published research on NARM is covered in this study. The third threat is the inclusion and exclusion of articles. To determine whether a paper should be included or excluded, we first reviewed the title, abstract, and keywords according to the inclusion and exclusion criteria. Then, we manually checked for references to ensure we did not miss any relevant papers. Additionally, we evaluated the selected studies using a quality assessment procedure. The fourth threat is about the time frame. Since the search process was conducted in early 2022, only articles published between 1996 and the beginning of 2022 were included. It is possible that we may have missed some articles published after the specified time frame. The final threat concerns article selection. The authors of this article may have been biased in their choice and categorization of publications that were included. Two authors chose articles based on their personal experiences. Although the final selection was made by a single author, all studies were verified by other authors to minimize bias.

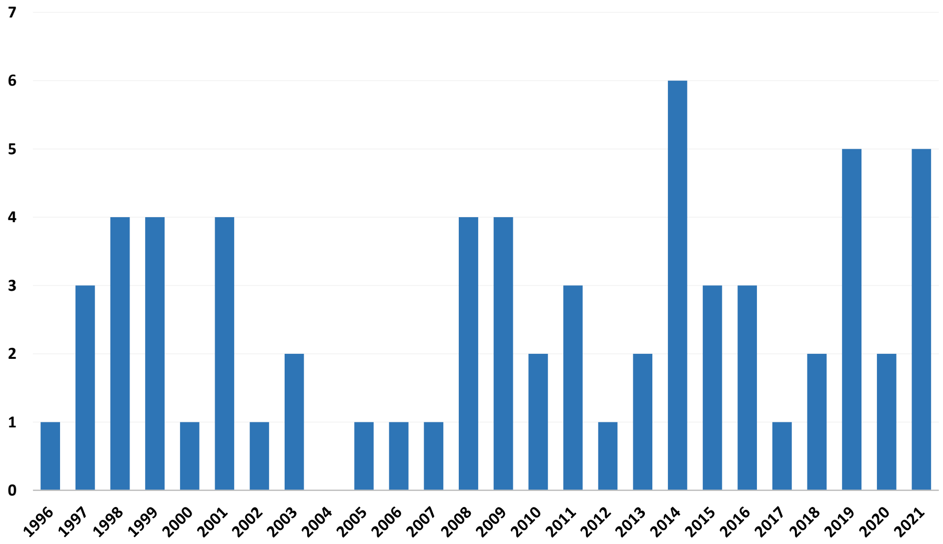
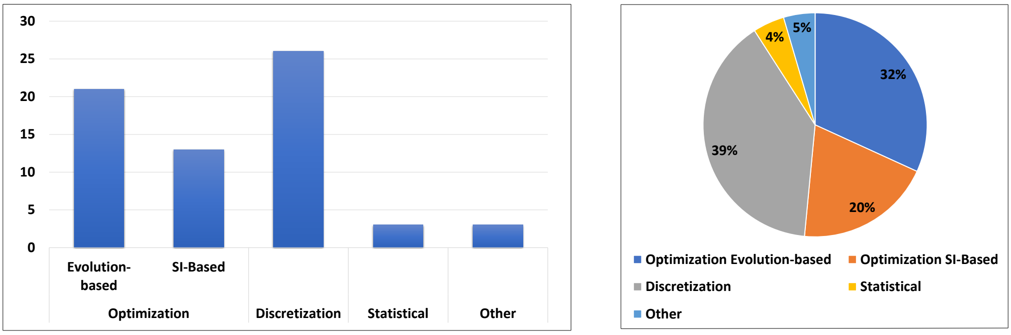


Fig. 5. Distribution of Articles by Publication Year.



(a) Distribution of the Articles by Method.

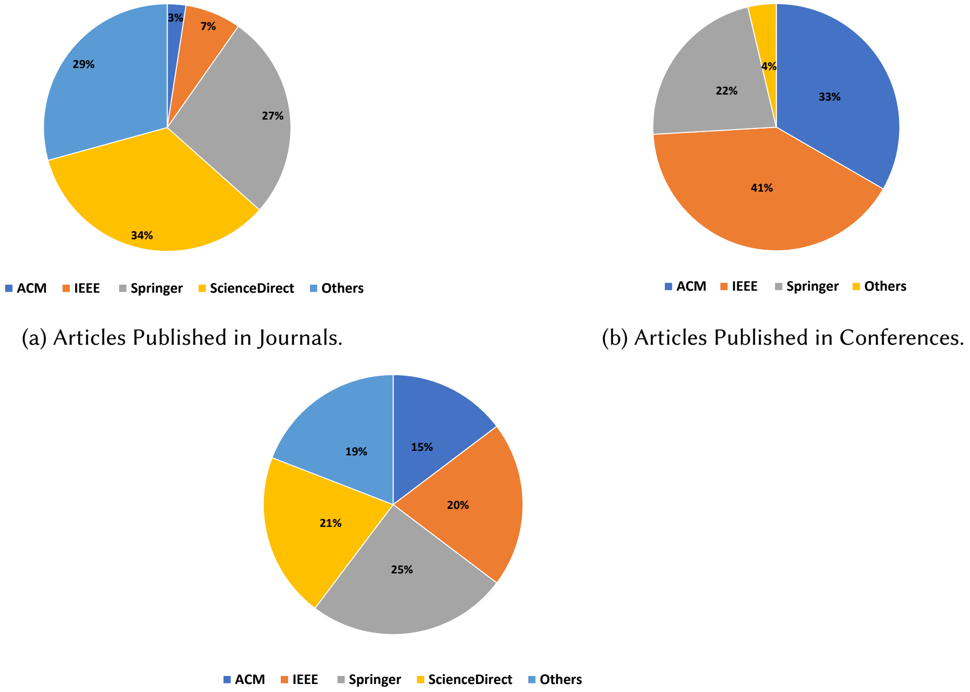
(b) Proportions of the Articles.

Fig. 6. Distribution of the Primary Study.

6 DISCUSSION AND FINDING

In this SLR, we analyzed a total of 68 studies on NARM published between 1996 and 2022. Our analysis revealed several significant findings and trends. Figure 5 illustrates the distribution of selected articles by publication year, with a notable concentration in 2014 and strong NARM trends in 2019 and 2021.

The distribution of primary studies by the method is presented in Figure 6. The majority of articles focus on the discretization method, followed by the evolution-based optimization method (Figure 6a). Statistical and other techniques make up a smaller portion. We further detail the partition of articles by the method as a percentage in Figure 6b. Table 16 provides the number of articles published in different journals and conferences. Overall, 60% of the selected articles were published in journals



(a) Articles Published in Journals.

(b) Articles Published in Conferences.

(c) Total Articles Published in Different Sources.

Fig. 7. Publication Source Distribution: 1996-2022.

Table 16. Type of Publications in the Area of NARM

Publication Source	Type of Publication	Number of Articles
IEEE	Journal	3
Springer	Journal	11
ACM	Journal	1
ScienceDirect	Journal	14
Other	Journal	12
IEEE	Conference	11
Springer	Conference	6
ACM	Conference	9
Other	Conference	1

and 40% in conferences. Figure 7 illustrates the percentage of articles published in journals and conferences. ScienceDirect published 34% journal articles (Figure 7a) however, IEEE published 41% conference articles (Figure 7b), which is the highest number and Springer published 25% overall articles (Figure 7c). We identified several methods for solving NARM, including the discretization method, optimization methods using evolutionary and bio-inspired approaches, the statistical method, and other methods. The discretization method was the most widely used, accounting for 39% of the total articles (Figure 6b). Evolution-based and SI-based optimization methods covered

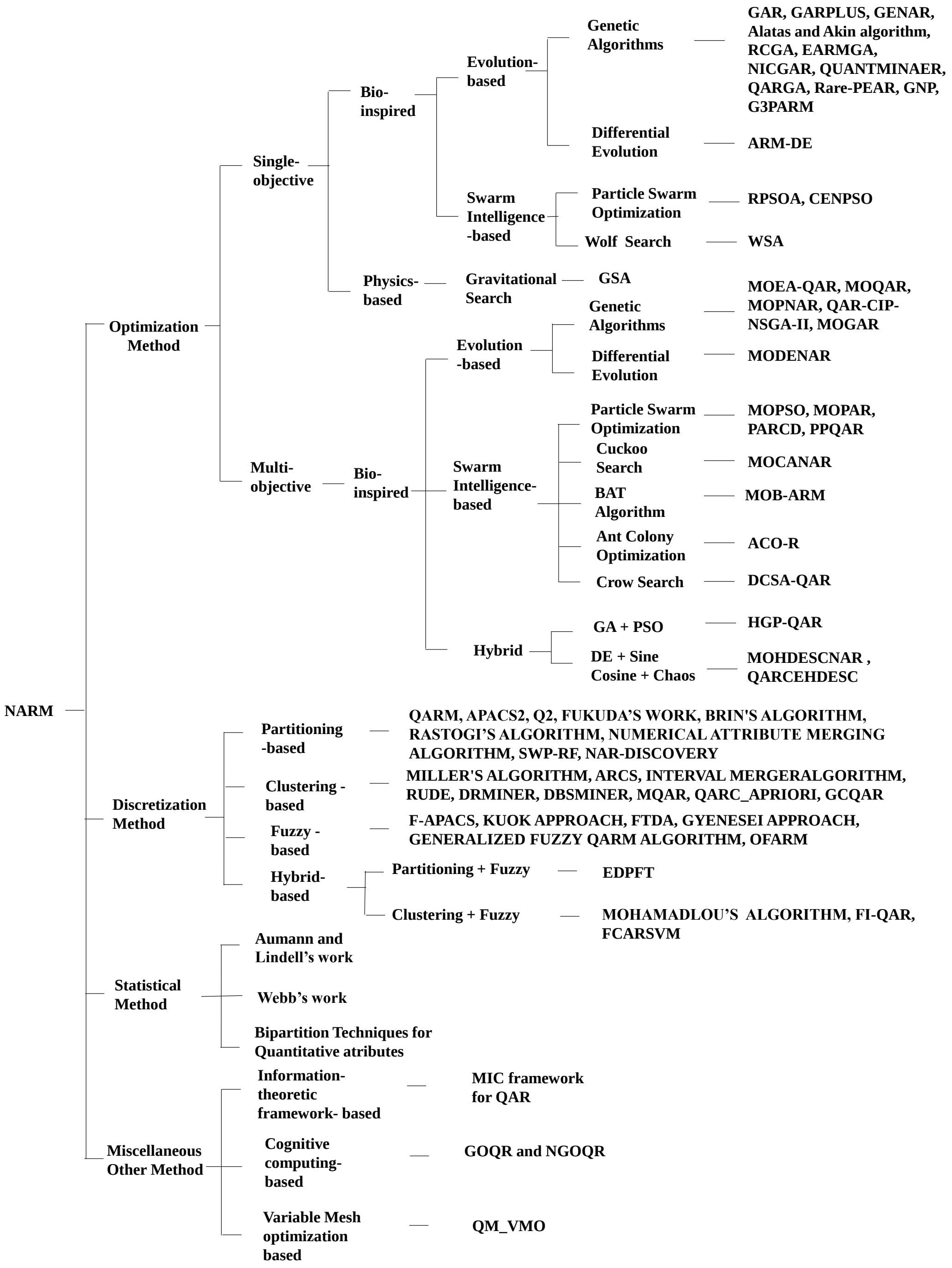

NARM
Optimization Method
Single-objective
Bio-inspired
Evolution-based
Genetic Algorithms
GAR, GARPLUS, GENAR, Alatas and Akin algorithm, RCGA, EARMGA, NICGAR, QUANTMINAER, QARGA, Rare-PEAR, GNP, G3PARM
Differential Evolution
ARM-DE
Swarm Intelligence-based
Particle Swarm Optimization
RPSOA, CENPSO
Wolf Search
WSA
Physics-based
Gravitational Search
GSA
Multi-objective
Bio-inspired
Evolution-based
Genetic Algorithms
MOEA-QAR, MOQAR, MOPNAR, QAR-CIP-NSGA-II, MOGAR
Differential Evolution
MODENAR
Swarm Intelligence-based
Particle Swarm Optimization
MOPSO, MOPAR, PARCD, PPQAR
Cuckoo Search
MOCANAR
BAT Algorithm
MOB-ARM
Ant Colony Optimization
ACO-R
Crow Search
DCSA-QAR
Hybrid
GA + PSO
HGP-QAR
DE + Sine Cosine + Chaos
MOHDESCNAR , QARCEHDESC
Discretization Method
Partitioning-based
QARM, APACS2, Q2, FUKUDA'S WORK, BRIN'S ALGORITHM, RASTOGI'S ALGORITHM, NUMERICAL ATTRIBUTE MERGING ALGORITHM, SWP-RF, NAR-DISCOVERY
Clustering -based
MILLER'S ALGORITHM, ARCS, INTERVAL MERGERALGORITHM, RUDE, DRMINER, DBSMINER, MQAR, QARC_APRIORI, GCQAR
Fuzzy -based
F-APACS, KUOK APPROACH, FTDA, GYENESEI APPROACH, GENERALIZED FUZZY QARM ALGORITHM, OFARM
Hybrid-based
Partitioning + Fuzzy
EDPFT
Clustering + Fuzzy
MOHAMADLOU'S ALGORITHM, FI-QAR, FCARSVM
Statistical Method
Aumann and Lindell's work
Webb's work
Bipartition Techniques for Quantitative atributes
Miscellaneous Other Method
Information-theoretic framework- based
MIC framework for QAR
Cognitive computing-based
GOQR and NGOQR
Variable Mesh optimization based
QM_VMO

Fig. 8. Visual Exploration and Illustration of NARM Methods and Algorithms: A Comprehensive Overview.

Table 17. Advantages and Limitations of NARM Methods

NARM Method	Advantages	Limitations
The Discretization method	Easier to interpret. Efficient and scalable to process huge datasets. Able to deal with continuous variables. Highly flexible.	Requirement of a user-specified threshold. Information loss. Unable to deal with high dimensional data. Discretization bias can lead to inaccurate or unreliable results. Membership functions should be known in advance.
The Optimization method	No need to determine the minimum support and minimum confidence. No need for a prior discretization step. High scalability.	Higher computational cost. Low search capability in the local area. Stuck in local minima. Convergence issues.
The Statistical method	Measuring significance to identify meaningful association rules. Can handle noise and missing data. Provide quantifiable results.	Lack of scalability. Not able to detect complex relationships in data. Not suitable for handling high-dimensional data. Not well-suited for mining rules in numerical data.

32% and 20% of the total articles, respectively (Figure 6b). The optimization and discretization methods had a greater impact compared to the statistical and other methods. Each method also employs various approaches to address NARM problems. For example, the discretization method includes the clustering approach, which encompasses density-based and grid-based techniques. The partition approach, which involves converting continuous numerical data into discrete values by grouping them into intervals or bins, was found to be simple and easy to implement. However, the choice of interval or bin size can affect result accuracy, and it may not be suitable for datasets with a large number of variables. The optimization method encompasses a range of approaches, such as genetic algorithms, grammar-guided genetic programming, differential evolution, particle swarm optimization, gravity-based algorithms, swarm-based algorithms, Cauchy distribution, and hybrid-based methods. The statistical method is limited and utilizes various distribution scales, including mean, median, variance, and standard deviation. Some studies did not fit into any specific method and were categorized as miscellaneous other methods based on the information-theoretic approach, cognitive computing, and variable mesh.

Many algorithms [46, 65, 70, 96, 105, 113, 114] based on the discretization method use apriori algorithm for generating association rules. However, the evolution and SI-based algorithms do not use the apriori algorithm. Additionally, certain algorithms under the discretization method have employed new measures, including density measure [40, 67], R-measure [113], Certainty Factor [114], and adjusted difference measure [25]. Figure 8 demonstrates the visual presentation of NARM methods and their algorithms. In Table 17, we have summarized the advantages and limitations of each method.

Our analysis found that most studies based on the discretization method used synthetic and real-world data to evaluate the effectiveness of NARM algorithms. However, evolutionary and SI-based algorithms mostly used common datasets, such as *Quake*, *Basketball*, *Bolt*, *Bodyfat*. Furthermore, the *Iris* dataset was the only one commonly used by discretization, optimization, and statistical methods-based algorithms. It is crucial to note that the choice of the dataset may impact the performance of the methods, and further studies are needed to evaluate the algorithms' practical applicability on real-world datasets.

In our extensive review of the literature on NARM, we analyzed various metrics used to evaluate the performance and effectiveness of different algorithms and models. Our study focused on important metrics such as generated number of rules, run time, and value of support and confidence. Support measures the frequency of a specific item set or rules in the dataset, frequently used in conjunction with confidence, which quantifies how likely a certain outcome is given an antecedent. However, it is crucial to carefully interpret and evaluate the reliability and validity of the metrics used, as they can lead to spurious or irrelevant associations if not used properly. Our SLR sheds light on the most commonly used metrics in NARM, including those used by multi-objective algorithms.

Multi-objective NARM algorithms consider different objectives simultaneously to generate a set of Pareto-optimal solutions that balance competing objectives. The choice of objective for multi-objective NARM algorithms depends on the research question and data characteristics. Some common objectives include maximizing support or confidence while minimizing the number of rules generated.

As a rapidly evolving research area, NARM presents numerous potential future directions for research. These include exploring new scalable optimization algorithms, addressing Big Data challenges, incorporating explainable AI into the mining process, integrating machine learning techniques, addressing security concerns, and using hybrid approaches. By pursuing these directions, researchers can advance the state of the art in NARM and develop more effective and practical solutions for real-world applications.

7 CONCLUSION

This article addressed a significant research gap in the field of NARM and provided readers with a comprehensive understanding of the state-of-the-art methodologies and developments in the domain. Moreover, this study serves as a foundation for future research and offers comprehensive insights for researchers working on NARM-related problems. To achieve this, a comprehensive SLR is conducted based on the guidelines set forth by Kitchenham and Charter. We conducted a detailed examination of a wide range of methods, algorithms, metrics, and datasets sourced from 1,140 scholarly articles spanning the period from the introduction of NARM in 1996 to 2022. Eventually, through a rigorous selection process, including several inclusion, exclusion and quality assessment criteria, 68 articles were selected for this SLR. By providing an exhaustive understanding of the existing NARM methods, highlighting their strengths and limitations, as well as identifying research challenges and future directions, we aim to stimulate innovative thinking and encourage the exploration of novel approaches in NARM. These perspectives include exploring new scalable optimization algorithms, analyzing NARM methods with big data, incorporating explainable AI into the mining process, incorporating machine learning techniques, addressing security concerns, and using hybrid approaches. Subsequently, based on the finding of this SLR, a novel discretization measure is presented to aid in NARM that explicitly addresses the human perception of partitions. The ultimate goal of this review is to inspire and guide researchers in developing more effective and practical solutions for real-world NARM applications.

ACKNOWLEDGMENTS

This work has been partially conducted in the project "ICT programme" which was supported by the European Union through the European Social Fund.

DECLARATIONS

Conflict of interest Authors declare that they have no conflict of interest.

REFERENCES

[1] Dhrubajit Adhikary and Swarup Roy. 2015. Mining quantitative association rules in real-world databases: A review. In *2015 1st International Conference on Computing and Communication Systems (I3CS)*, Vol. 1. IGI Global, India, 87–92.

[2] Dhrubajit Adhikary and Swarup Roy. 2015. Trends in quantitative association rule mining techniques. In *2015 IEEE 2nd International Conference on Recent Trends in Information Systems (ReTIS)*. IEEE, India, 126–131.

[3] Israel Edem Agbehadji, Simon Fong, and Richard Millham. 2016. Wolf search algorithm for numeric association rule mining. In *2016 IEEE International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*. IEEE, China, 146–151. https://doi.org/10.1109/ICCCBDA.2016.7529549

[4] Rakesh Agrawal, Tomasz Imieliński, and Arun Swami. 1993. Mining Association Rules Between Sets of Items in Large Databases. *ACM SIGMOD Record* 22, 2 (1993), 207–216. https://doi.org/10.1145/170036.170072

[5] Rakesh Agrawal and Ramakrishnan Srikant. 1994. Fast Algorithms for Mining Association Rules in Large Databases. In *Proceedings of VLDB'1994 – the 20th International Conference on Very Large Data Bases*. Morgan Kaufmann, Chile, 487–499.

[6] Bilal Alataş and Erhan Akin. 2006. An efficient genetic algorithm for automated mining of both positive and negative quantitative association rules. *Soft Computing* 10, 3 (2006), 230–237.

[7] Bilal Alatas and Erhan Akin. 2008. Rough particle swarm optimization and its applications in data mining. *Soft Computing* 12, 12 (2008), 1205–1218.

[8] Bilal Alatas and Erhan Akin. 2009. Chaotically encoded particle swarm optimization algorithm and its applications. *Chaos, Solitons & Fractals* 41, 2 (2009), 939–950. https://doi.org/10.1016/j.chaos.2008.04.024

[9] Bilal Alatas, Erhan Akin, and Ali Karci. 2008. MODENAR: Multi-objective differential evolution algorithm for mining numeric association rules. *Applied Soft Computing* 8, 1 (2008), 646–656. https://doi.org/10.1016/j.asoc.2007.05.003

[10] Mehrdad Almasi and Mohammad Saniee Abadeh. 2015. Rare-PEARs: A new multi objective evolutionary algorithm to mine rare and non-redundant quantitative association rules. *Knowledge-Based Systems* 89 (2015), 366–384. https://doi.org/10.1016/j.knosys.2015.07.016

[11] Elif Varol Altay and Bilal Alatas. 2020. Intelligent optimization algorithms for the problem of mining numerical association rules. *Physica A: Statistical Mechanics and its Applications* 540 (2020), 123142.

[12] Elif Varol Altay and Bilal Alatas. 2021. Differential evolution and sine cosine algorithm based novel hybrid multi-objective approaches for numerical association rule mining. *Information Sciences* 554 (2021), 198–221. https://doi.org/10.1016/j.ins.2020.12.055

[13] Elif Varol Altay and Bilal Alatas. 2022. Chaos numbers based a new representation scheme for evolutionary computation: Applications in evolutionary association rule mining. *Concurrency and Computation: Practice and Experience* 34, 5 (2022), e6744.

[14] Victoria Pachón Álvarez and Jacinto Mata Vázquez. 2012. An evolutionary algorithm to discover quantitative association rules from huge databases without the need for an a priori discretization. *Expert Systems with Applications* 39, 1 (2012), 585–593.

[15] Alireza Askarzadeh. 2016. A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm. *Computers & Structures* 169 (2016), 1–12. https://doi.org/10.1016/j.compstruc.2016.03.001

[16] Iztok Fister Jr. au2, Iztok Fister, Dušan Fister, Vili Podgorelec, and Sancho Salcedo-Sanz. 2023. A comprehensive review of visualization methods for association rule mining: Taxonomy, Challenges, Open problems and Future ideas. arXiv:cs.DB/2302.12594

[17] Yonatan Aumann and Yehuda Lindell. 2003. A statistical theory for quantitative association rules. *Journal of Intelligent Information Systems* 20, 3 (2003), 255–283.

[18] Kitchenham BA and Stuart Charters. 2007. Guidelines for performing Systematic Literature Reviews in Software Engineering. 2 (01 2007).

[19] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador Garcia, Sergio Gil-Lopez, Daniel Molina, Richard Benjamins, Raja Chatila, and Francisco Herrera. 2020. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion* 58 (2020), 82–115. https://doi.org/10.1016/j.inffus.2019.12.012

[20] Vahid Beiranvand, Mohamad Mobasher-Kashani, and Azuraliza Abu Bakar. 2014. Multi-objective PSO algorithm for mining numerical association rules without a priori discretization. *Expert systems with applications* 41, 9 (2014), 4259–4273.

[21] Eric Bonabeau, Marco Dorigo, and Guy Theraulaz. 1999. *Swarm intelligence: from natural to artificial systems*. Number 1. Oxford university press.

[22] Sergey Brin, Rajeev Rastogi, and Kyuseok Shim. 1999. Mining optimized gain rules for numeric attributes. In *Proceedings of KDD'99 – the 5th ACM SIGKDD international conference on knowledge discovery and data mining*. ACM, New York, 135–144.

[23] Oliver Büchter and Rüdiger Wirth. 1998. Discovery of association rules over ordinal data: A new and faster algorithm and its application to basket analysis. In *Research and Development in Knowledge Discovery and Data Mining*, Xindong Wu, Ramamohanarao Kotagiri, and Kevin B. Korb (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 36–47.

[24] Umit Can and Bilal Alatas. 2017. Automatic mining of quantitative association rules with gravitational search algorithm. *International Journal of Software Engineering and Knowledge Engineering* 27, 03 (2017), 343–372.

[25] Keith CC Chan and Wai-Ho Au. 1997. An effective algorithm for mining interesting quantitative association rules. In *Proceedings of the 1997 ACM symposium on Applied computing*. ACM, San Jose, CA, United States, 88–90.

[26] Keith CC Chan and Wai-Ho Au. 1997. Mining fuzzy association rules. In *Proceedings of the sixth international conference on information and knowledge management*. ACM, Las Vegas Nevada USA, 209–215.

[27] C.A.C. Coello, G.T. Pulido, and M.S. Lechuga. 2004. Handling multiple objectives with particle swarm optimization. *IEEE Transactions on Evolutionary Computation* 8, 3 (2004), 256–279. https://doi.org/10.1109/TEVC.2004.826067

[28] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* 6, 2 (2002), 182–197. https://doi.org/10.1109/4235.996017

[29] Xin Dong and Dechang Pi. 2014. An Effective Method for Mining Quantitative Association Rules with Clustering Partition in Satellite Telemetry Data. In *2014 Second International Conference on Advanced Cloud and Big Data*. IEEE, Huangshan, China, 26–33. https://doi.org/10.1109/CBD.2014.12

[30] Marco Dorigo, Mauro Birattari, and Thomas Stutzle. 2006. Ant colony optimization. *IEEE Computational Intelligence Magazine* 1, 4 (2006), 28–39. https://doi.org/10.1109/MCI.2006.329691

[31] Agoston E Eiben and James E Smith. 2015. *Introduction to evolutionary computing*. Springer, Berlin Heidelberg New York.

[32] Larry J. Eshelman. 1991. The CHC Adaptive Search Algorithm: How to Have Safe Search When Engaging in Nontraditional Genetic Recombination. In *Foundations of Genetic Algorithms*, GREGORY J.E. RAWLINS (Ed.). Vol. 1. Elsevier, 265–283. https://doi.org/10.1016/B978-0-08-050684-5.50020-3

[33] Guanghui Fan, Wenjuan Shi, Liang Guo, Jun Zeng, Kaixuan Zhang, and Guan Gui. 2019. Machine Learning Based Quantitative Association Rule Mining Method for Evaluating Cellular Network Performance. *IEEE Access* 7 (2019), 166815–166822. https://doi.org/10.1109/ACCESS.2019.2953943

[34] Iztok Fister, Andres Iglesias, Akemi Galvez, Javier Del Ser, Eneko Osaba, and Iztok Fister. 2018. Differential Evolution for Association Rule Mining Using Categorical and Numerical Attributes. In *Intelligent Data Engineering and Automated Learning – IDEAL 2018*, Hujun Yin, David Camacho, Paulo Novais, and Antonio J. Tallón-Ballesteros (Eds.). Springer International Publishing, Cham, 79–88.

[35] Iztok Fister Jr., Vili Podgorelec, and Iztok Fister. 2021. Improved Nature-Inspired Algorithms for Numeric Association Rule Mining. In *Intelligent Computing and Optimization*, Pandian Vasant, Ivan Zelinka, and Gerhard-Wilhelm Weber (Eds.). Springer International Publishing, Cham, 187–195.

[36] Takeshi Fukuda, Yasuhiko Morimoto, Shinichi Morishita, and Takeshi Tokuyama. 1999. Mining optimized association rules for numeric attributes. *J. Comput. System Sci.* 58, 1 (1999), 1–12.

[37] Liqiang Geng and Howard J. Hamilton. 2006. Interestingness Measures for Data Mining: A Survey. *ACM Comput. Surv.* 38, 3 (sep 2006), 9–es. https://doi.org/10.1145/1132960.1132963

[38] Ashish Ghosh and Bhabesh Nath. 2004. Multi-objective rule mining using genetic algorithms. *Information Sciences* 163, 1 (2004), 123–133. https://doi.org/10.1016/j.ins.2003.03.021 Soft Computing Data Mining.

[39] Anjana Gosain and Maneela Bhugra. 2013. A comprehensive survey of association rules on quantitative data in data mining. In *2013 IEEE Conference on Information & Communication Technologies*. IEEE, India, 1003–1008.

[40] Yunkai Guo, Junrui Yang, and Yulei Huang. 2008. An Effective Algorithm for Mining Quantitative Association Rules Based on High Dimension Cluster. In *2008 4th International Conference on Wireless Communications, Networking and Mobile Computing*. IEEE, Dalian, China, 1–4. https://doi.org/10.1109/WiCom.2008.2663

[41] H Altay Guvenir, Ilhan Uysal, and Function Approximation Repositor. 2000. Function approximation repository.

[42] Attila Gyenesei. 2001. A Fuzzy Approach for Mining Quantitative Association Rules. *Acta Cybern.* 15 (2001), 305–320.

[43] Jiawei Han, Jian Pei, Yiwen Yin, and Runying Mao. 2004. Mining frequent patterns without candidate generation: A frequent-pattern tree approach. *Data mining and knowledge discovery* 8, 1 (2004), 53–87.

[44] Kamel Eddine Heraguemi, Nadjet Kamel, and Habiba Drias. 2018. Multi-objective bat algorithm for mining numerical association rules. *International Journal of Bio-Inspired Computation* 11, 4 (2018), 239–248.

[45] Jhon H HOLLAND. 1992. *Adaption in Natural and Artificial Systems:an introductory analysis with applications to biology, control, and artificial intelligence.* MIT press, USA.

[46] Tzung-Pei Hong, Chan-Sheng Kuo, and Sheng-Chai Chi. 1999. Mining association rules from quantitative data. *Intelligent data analysis* 3, 5 (1999), 363–376.

[47] Zhiyong Hu, Mingwen Shao, Huan Liu, and Jvsheng Mi. 2022. Cognitive computing and rule extraction in generalized one-sided formal contexts. *Cognitive Computation* 14, 6 (2022), 2087–2107.

[48] Iván Fredy Jaramillo, Javier Garzás, and Andrés Redchuk. 2021. Numerical Association Rule Mining from a Defined Schema Using the VMO Algorithm. *Applied Sciences* 11, 13 (2021), 21. https://doi.org/10.3390/app11136154

[49] Irene Kahvazadeh and Mohammad Saniee Abadeh. 2015. MOCANAR: a multi-objective cuckoo search algorithm for numeric association rule discovery. *Computer Science & Information Technology* 99 (2015), 113.

[50] Gong-Mi Kang, Yang-Sae Moon, Hun-Young Choi, and Jinho Kim. 2009. Bipartition techniques for quantitative attributes in association rule mining. In *TENCON 2009-2009 IEEE Region 10 Conference*. IEEE, Singapore, 1–6.

[51] Minakshi Kaushik, Rahul Sharma, Sijo Arakkal Peious, and Dirk Draheim. 2021. Impact-Driven Discretization of Numerical Factors: Case of Two- and Three-Partitioning. In *Big Data Analytics*. Springer International Publishing, Cham, 244–260.

[52] Minakshi Kaushik, Rahul Sharma, Sijo Arakkal Peious, Mahtab Shahin, Sadok Ben Yahia, and Dirk Draheim. 2020. On the Potential of Numerical Association Rule Mining. In *International Conference on Future Data and Security Engineering*. Springer, Vietnam, 3–20.

[53] Minakshi Kaushik, Rahul Sharma, Sijo Arakkal Peious, Mahtab Shahin, Sadok Ben Yahia, and Dirk Draheim. 2021. A Systematic Assessment of Numerical Association Rule Mining Methods. *SN Computer Science* 2, 5 (2021), 1–13.

[54] Minakshi Kaushik, Rahul Sharma, Mahtab Shahin, Sijo Arakkal Peious, and Dirk Draheim. 2022. An Analysis of Human Perception of Partitions of Numerical Factor Domains. In *Information Integration and Web Intelligence*, Eric Pardede, Pari Delir Haghighi, Ismail Khalil, and Gabriele Kotsis (Eds.). Springer Nature Switzerland, Cham, 137–144.

[55] Minakshi Kaushik, Rahul Sharma, Ankit Vidyarthi, and Dirk Draheim. 2022. Discretizing Numerical Attributes: An Analysis of Human Perceptions. In *New Trends in Database and Information Systems*. Springer International Publishing, Cham, 188–197.

[56] Yiping Ke, James Cheng, and Wilfred Ng. 2008. An information-theoretic approach to quantitative association rule mining. *Knowledge and Information Systems* 16, 2 (2008), 213–244.

[57] James Kennedy and Russell Eberhart. 1995. Particle swarm optimization. In *Proceedings of ICNN'95-international conference on neural networks*, Vol. 4. IEEE, Australia, 1942–1948.

[58] Rohan Khade, Nital Patel, and Jessica Lin. 2015. Supervised Dynamic and Adaptive Discretization for Rule Mining. In *SDM Workshop on Big Data and Stream Analytics, 2015*.

[59] Keivan Kianmehr, Mohammed Alshalalfa, and Reda Alhajj. 2010. Fuzzy clustering-based discretization for gene expression classification. *Knowledge and Information Systems* 24, 3 (2010), 441–465.

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

[61] Ren-Jieh Kuo, Monalisa Gosumolo, and Ferani E Zulvia. 2019. Multi-objective particle swarm optimization algorithm using adaptive archive grid for numerical association rule mining. *Neural Computing and Applications* 31, 8 (2019), 3559–3572.

[62] Chan Man Kuok, Ada Fu, and Man Hon Wong. 1998. Mining Fuzzy Association Rules in Databases. *SIGMOD Rec.* 27, 1 (mar 1998), 41–46. https://doi.org/10.1145/273244.273257

[63] Makhlouf Ledmi, Hamouma Moumen, Abderrahim Siam, Hichem Haouassi, and Nabil Azizi. 2021. A Discrete Crow Search Algorithm for Mining Quantitative Association Rules. *International Journal of Swarm Intelligence Research (IJSIR)* 12, 4 (2021), 101–124.

[64] Keon-Myung Lee. 2001. Mining generalized fuzzy quantitative association rules with fuzzy generalization hierarchies. In *Proceedings Joint 9th IFSA World Congress and 20th NAFIPS International Conference (Cat. No. 01TH8569)*, Vol. 5. IEEE, Canada, 2977–2982. https://doi.org/10.1109/NAFIPS.2001.943701

[65] Brian Lent, Arun Swami, and Jennifer Widom. 1997. Clustering association rules. In *Proceedings 13th International Conference on Data Engineering*. IEEE, Birmingham, UK, 220–231.

[66] Jiuyong Li, Hong Shen, and Rodney Topor. 1999. An Adaptive Method of Numerical Attribute Merging for Quantitative Association Rule Mining. In *Internet Applications*, Lucas Chi Kwong Hui and Dik-Lun Lee (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 41–50.

[67] Wang Lian, David W Cheung, and SM Yiu. 2005. An efficient algorithm for finding dense regions for mining quantitative association rules. *Computers & Mathematics with Applications* 50, 3-4 (2005), 471–490.

[68] M. Lichman. 2013. UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. http://archive.ics.uci.edu/ml Publication Title: UCI Machine Learning Repository.

[69] WANG Ling, WU Lu-lu, and FU Dong-mei. 2014. A density-based fuzzy adaptive clustering algorithm. *Chinese Journal of Engineering* 36, 20141120 (2014), 1560. https://doi.org/10.13374/j.issn1001-053x.2014.11.020

[70] Marcus-Christopher Lud and Gerhard Widmer. 2000. Relative Unsupervised Discretization for Association Rule Mining. In *Principles of Data Mining and Knowledge Discovery*. Springer Berlin Heidelberg, Berlin, Heidelberg, 148–158.

[71] Jose Maria Luna, Mykola Pechenizkiy, and Sebastian Ventura. 2016. Mining exceptional relationships with grammar-guided genetic programming. *Knowledge and Information Systems* 47, 3 (2016), 571–594.

[72] José María Luna, José Raúl Romero, Cristóbal Romero, and Sebastián Ventura. 2014. Reducing gaps in quantitative association rules: A genetic programming free-parameter algorithm. *Integrated Computer-Aided Engineering* 21, 4 (2014), 321–337.

[73] María Martínez-Ballesteros, Francisco Martínez-Álvarez, Alicia Troncoso, and José C Riquelme. 2011. An evolutionary algorithm to discover quantitative association rules in multidimensional time series. *Soft Computing* 15, 10 (2011), 2065–2084.

[74] María Martínez-Ballesteros, A Troncoso, Francisco Martínez-Álvarez, and José C Riquelme. 2010. Mining quantitative association rules based on evolutionary computation and its application to atmospheric pollution. *Integrated Computer-Aided Engineering* 17, 3 (2010), 227–242.

[75] María Martínez-Ballesteros, A Troncoso, Francisco Martínez-Álvarez, and José C Riquelme. 2016. Improving a multi-objective evolutionary algorithm to discover quantitative association rules. *Knowledge and Information Systems* 49, 2 (2016), 481–509.

[76] D. Martín, J. Alcalá-Fdez, A. Rosete, and F. Herrera. 2016. NICGAR: A Niching Genetic Algorithm to mine a diverse set of interesting quantitative association rules. *Information Sciences* 355-356 (2016), 208–228. https://doi.org/10.1016/j.ins.2016.03.039

[77] Diana Martín, Alejandro Rosete, Jess Alcalá-Fdez, and Francisco Herrera. 2014. A New Multiobjective Evolutionary Algorithm for Mining a Reduced Set of Interesting Positive and Negative Quantitative Association Rules. *IEEE Transactions on Evolutionary Computation* 18, 1 (2014), 54–69. https://doi.org/10.1109/TEVC.2013.2285016

[78] D. Martín, A. Rosete, J. Alcalá-Fdez, and F. Herrera. 2014. QAR-CIP-NSGA-II: A new multi-objective evolutionary algorithm to mine quantitative association rules. *Information Sciences* 258 (2014), 1–28. https://doi.org/10.1016/j.ins.2013.09.009

[79] Jacinto Mata, JL Alvarez, and JC Riquelme. 2001. Mining numeric association rules with genetic algorithms. In *Artificial neural nets and genetic algorithms*. Springer, Czech Republic, 264–267.

[80] Jacinto Mata, José-Luis Alvarez, and José-Cristobal Riquelme. 2002. Discovering numeric association rules via evolutionary algorithm. In *Pacific-Asia conference on knowledge discovery and data mining*. Springer, Taiwan, 40–51.

[81] Yasmina Medjadba, Dan Hu, Wei Liu, and Xianchuan Yu. 2020. Combining Graph Clustering and Quantitative Association Rules for Knowledge Discovery in Geochemical Data Problem. *IEEE Access* 8 (2020), 40453–40473. https://doi.org/10.1109/ACCESS.2019.2948800

[82] R. J. Miller and Y. Yang. 1997. Association Rules over Interval Data. In *SIGMOD '97*. Association for Computing Machinery, New York, NY, USA, 452–461. https://doi.org/10.1145/253260.253361

[83] B. Minaei-Bidgoli, R. Barmaki, and M. Nasiri. 2013. Mining numerical association rules via multi-objective genetic algorithms. *Information Sciences* 233 (2013), 15–24. https://doi.org/10.1016/j.ins.2013.01.028

[84] Hamid Mohamadlou, Reza Ghodsi, Jafar Razmi, and Abbas Keramati. 2009. A method for mining association rules in quantitative and fuzzy data. In *2009 International Conference on Computers Industrial Engineering*. IEEE, France, 453–458. https://doi.org/10.1109/ICCIE.2009.5223873

[85] Katherine Moreland and Klaus Truemper. 2009. Discretization of target attributes for subgroup discovery. In *International Workshop on Machine Learning and Data Mining in Pattern Recognition*. Springer, Germany, 44–52.

[86] F. Moslehi and A. Haeri. 2020. A genetic algorithm-based framework for mining quantitative association rules without specifying minimum support and minimum confidence. *Scientia Iranica* 27, 3 (2020), 1316–1332. https://doi.org/10.24200/sci.2019.51030.1969

[87] Fateme Moslehi, Abdorrahman Haeri, and Francisco Martínez-Álvarez. 2020. A novel hybrid GA-PSO framework for mining quantitative association rules. *soft computing* 24, 6 (2020), 4645–4666.

[88] Parisa Moslehi, Behrouz Minaei Bidgoli, Mahdi Nasiri, and Afshin Salajegheh. 2011. Multi-objective numeric association rules mining via ant colony optimization for continuous domains without specifying minimum support and minimum confidence. *International Journal of Computer Science Issues (IJCSI)* 8, 5 (2011), 34.

[89] Gregory Piatetsky-Shapiro. 1991. Discovery, Analysis, and Presentation of Strong Rules. In *Knowledge Discovery in Databases*, Gregory Piatetsky-Shapiro and William J. Frawley (Eds.). AAAI/MIT Press, 229–248.

[90] Amilkar Puris, Rafael Bello, Daniel Molina, and Francisco Herrera. 2012. Variable mesh optimization for continuous optimization problems. *Soft Computing* 16, 3 (2012), 511–525.

[91] Esmat Rashedi, Hossein Nezamabadi-Pour, and Saeid Saryazdi. 2009. GSA: a gravitational search algorithm. *Information sciences* 179, 13 (2009), 2232–2248.

[92] R. Rastogi and Kyuseok Shim. 2002. Mining optimized association rules with categorical and numeric attributes. *IEEE Transactions on Knowledge and Data Engineering* 14, 1 (2002), 29–50. https://doi.org/10.1109/69.979971

[93] Ansaf Salleb-Aouissi, Christel Vrain, and Cyril Nortet. 2007. QuantMiner: A Genetic Algorithm for Mining Quantitative Association Rules. In *IJCAI*, Vol. 7. Morgan Kaufmann Publishers Inc., India, 1035–1040.

[94] Rahul Sharma, Minakshi Kaushik, Sijo Arakkal Peious, Sadok Ben Yahia, and Dirk Draheim. 2020. Expected vs. unexpected: selecting right measures of interestingness. In *Big Data Analytics and Knowledge Discovery*, Vol. 12393.

Springer International Publishing, Cham, 38–47.

[95] Chunyao Song and Tingjian Ge. 2013. Discovering and Managing Quantitative Association Rules. In *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management (CIKM '13)*. Association for Computing Machinery, New York, NY, USA, 2429–2434. https://doi.org/10.1145/2505515.2505611

[96] Ramakrishnan Srikant and Rakesh Agrawal. 1996. Mining quantitative association rules in large relational tables. In *Proceedings of the 1996 ACM SIGMOD international conference on Management of data*. ACM, Canada, 1–12.

[97] Rainer Storn and Kenneth Price. 1997. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization* 11, 4 (1997), 341–359.

[98] Karla Taboada, Eloy Gonzales, Kaoru Shimada, Shingo Mabu, Kotaro Hirasawa, and Jinglu Hu. 2008. Association rule mining for continuous attributes using genetic network programming. *IEEJ transactions on electrical and electronic engineering* 3, 2 (2008), 199–211.

[99] Imam Tahyudin and Hidetaka Nambo. 2019. Improved optimization of numerical association rule mining using hybrid particle swarm optimization and cauchy distribution. *International Journal of Electrical and Computer Engineering* 9, 2 (2019), 1359.

[100] Tomohiro Takagi and Michio Sugeno. 1985. Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man, and Cybernetics* SMC-15, 1 (1985), 116–132. https://doi.org/10.1109/TSMC.1985.6313399

[101] Pang-Ning Tan, Vipin Kumar, and Jaideep Srivastava. 2004. Selecting the right objective measure for association analysis. *Information Systems* 29, 4 (2004), 293–313.

[102] Rui Tang, Simon Fong, Xin-She Yang, and Suash Deb. 2012. Wolf search algorithm with ephemeral memory. In *Seventh International Conference on Digital Information Management (ICDIM 2012)*. IEEE, Macao, 165–172.

[103] Akbar Telikani, Amir H. Gandomi, and Asadollah Shahbahrami. 2020. A survey of evolutionary computation for association rule mining. *Information Sciences* 524 (2020), 318–352. https://doi.org/10.1016/j.ins.2020.02.073

[104] Ke Wang, Soon Hock William Tay, and Bing Liu. 1998. Interestingness-Based Interval Merger for Numeric Association Rules. In *KDD*, Vol. 98. AAAI Press, New York, 121–128.

[105] Ling Wang, Ji-Yuan Dong, and Shu-Lin Li. 2015. Fuzzy Inference Algorithm based on Quantitative Association Rules. *Procedia Computer Science* 61 (12 2015), 388–394. https://doi.org/10.1016/j.procs.2015.09.166

[106] Geoffrey I Webb. 2001. Discovering associations with numeric variables. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, San Francisco California, 383–388.

[107] Danfeng Yan, Xuan Zhao, Rongheng Lin, and Demeng Bai. 2019. PPQAR: parallel PSO for quantitative association rule mining. *Peer-to-Peer Networking and Applications* 12, 5 (2019), 1433–1444.

[108] Xiaowei Yan, Chengqi Zhang, and Shichao Zhang. 2009. Genetic algorithm-based strategy for identifying association rules without specifying actual minimum support. *Expert Systems with Applications* 36, 2 (2009), 3066–3076.

[109] Junrui Yang and Zhang Feng. 2010. An effective algorithm for mining quantitative associations based on subspace clustering. In *2010 International Conference on Networking and Digital Society*, Vol. 1. IEEE, China, 175–178. https://doi.org/10.1109/ICNDS.2010.5479600

[110] Xin-She Yang. 2010. A new metaheuristic bat-inspired algorithm. In *Nature inspired cooperative strategies for optimization (NICSO 2010)*. Springer, 65–74.

[111] Xin-She Yang and Suash Deb. 2009. Cuckoo search via Lévy flights. In *2009 World congress on nature & biologically inspired computing (NaBIC)*. IEEE, India, 210–214.

[112] Mohammed Javeed Zaki. 2000. Scalable algorithms for association mining. *IEEE transactions on knowledge and data engineering* 12, 3 (2000), 372–390. https://doi.org/10.1109/69.846291

[113] Weining Zhang. 1999. Mining fuzzy quantitative association rules. In *Proceedings 11th International Conference on Tools with Artificial Intelligence*. IEEE, USA, 99–102.

[114] Hui Zheng, Jing He, Guangyan Huang, and Yanchun Zhang. 2014. Optimized fuzzy association rule mining for quantitative data. In *2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. IEEE, China, 396–403.

Appendix 7

[VII]

M. Kaushik. Swarm-intelligence algorithms for mining numerical association rules: An exhaustive multi-aspect analysis of performance assessment data. *SSRN Electronic Journal*, 2023

Highlights

Swarm-Intelligence Algorithms for Mining Numerical Association Rules: An Exhaustive Multi-Aspect Analysis of Performance Assessment Data

Minakshi Kaushik, Pilleriin Kõiva, Rahul Sharma, Iztok Fister Jr., Dirk Draheim

- Investigating the role of multi-objective optimization algorithms, especially swarm-intelligence algorithms for NARM.
- Analyzing four swarm-intelligence algorithms (MOPAR, MOCANAR, ACO-R and MOB-ARM) for NARM.
- Presenting an exhaustive multi-aspect analysis of performance assessment data.

Swarm-Intelligence Algorithms for Mining Numerical Association Rules: An Exhaustive Multi-Aspect Analysis of Performance Assessment Data

Minakshi Kaushik[a,*], Pilleriin Kõiva[b], Rahul Sharma[a], Iztok Fister Jr.[c], Dirk Draheim[a]

[a]*Information Systems Group,Tallinn University of Technology, Akadeemia tee 15a, Tallinn, 12618, Estonia*

[b]*School of Information Technologies, Tallinn University of Technology, Akadeemia tee 15a, Tallinn, 12618, Estonia*

[c]*Faculty of Electrical Engineering and Computer Science University of Maribor Koroska cesta 46 SI-2000 Maribor Slovenia*

Abstract

Numerical association rule mining (NARM) is an extended version of association rule mining that determines association rules in numerical data items, primarily via distribution, discretization and optimization techniques. Under the umbrella of optimization techniques, several evolutionary and swarm intelligence-based algorithms have been proposed to extract association rules from a numeric dataset. However, a sufficient understanding of the performance of swarm intelligence-based algorithms, especially for NARM, is still missing. In state-of-the-art, various swarm intelligence-based optimization algorithms are claimed to be better based on their arbitrary comparisons with different algorithms in different classes, e.g., swarm intelligence-based algorithms are compared with genetic algorithms. Unfortunately, they are not compared within their own class algorithms. Therefore, it is challenging to select an appropriate swarm intelligence-based algorithm for NARM. This article aims at filling this gap by conducting an exhaustive multi-aspect analysis of four popular swarm intelligence-based optimization algorithms (MOPAR, MOCANAR, ACO-R and MOB-ARM) with four real-world datasets and six major metrics and objectives: performance time, the number of rules, sup-

[*]Corresponding author

Email address: `minakshi.kaushik@taltech.ee` (Minakshi Kaushik)

Preprint submitted to Swarm and Evolutionary Computation *March 23, 2023*

port, confidence, comprehensibility, and interestingness. In our analysis, the MOPAR algorithm produces a low number of rules and shows high values of confidence, comprehensibility, and interestingness. The MOCANAR algorithm provides satisfactory results with respect to all six parameters across all the data sets. The ACO-R algorithm produces high-quality rules but needs parameter modification for a large number of attributes in datasets, and the MOB-ARM algorithm is way slower than the other three algorithms.

Keywords: swarm intelligence optimization, association rule mining, machine learning, numerical association rule mining

1. Introduction

Numerical association rule mining (NARM) is an extended version of classical association rule mining. It is used to mine association rules from continuous values or datasets consisting of numeric attributes, which makes it highly relevant for a plenty of today's data analysis tasks. Several methods, such as optimization, discretization, and distribution, are proposed in the literature to solve the problem of NARM [6, 28]. Out of them, the optimization method is one of the potential solutions to deal with such complex problems, which consists of Evolutionary-based, Swarm Intelligence (SI)-based and Physics-based algorithms [27]. Various SI-based optimization algorithms [10] consisting of animal, insect movements and the biological behaviour of natural objects are proposed in the literature [17, 33]. Primarily, these optimization algorithms are helpful in mining association rules from numeric datasets without discretization. However, it is still unclear which algorithms perform better for efficient NARM.

In the state of the art [9, 34, 24, 26], SI-based optimization algorithms are compared randomly with different algorithms in different classes; however, they are not compared in their own classes. Therefore, it is challenging to select the most suitable SI-based algorithm for NARM. This research conducts an exhaustive multi-aspect analysis of four popular SI-based optimization algorithms, i.e., MOPAR [9], ACO-R [34], MOB-ARM [24] and MOCANAR [26].

MOCANAR, MOB-ARM, MOPAR and ACO-R algorithms have been shown efficient in solving multi-objective optimization problems in various domains, including numerical association rule mining and continuous optimization. Moreover, these algorithms are relatively new algorithms that have been proposed in recent years, and their performance comparison is not yet available in the state of the art. Therefore, these algorithms are selected for their performance comparison to find a set of optimal solutions that trade-off between multiple objectives simultaneously. The performance evaluation of these algorithms will surely shed light on their potential advantages and limitations.

We first discuss the usefulness of these algorithms in NARM and then experiment with four real-world datasets. The results are then compared with a set of six metrics and objectives, i.e., the average number of rules mined, the average values of confidence, support, comprehensibility, interestingness of the rules and the time efficiency, i.e., average time spent in running the algorithms.

In our assessments, the MOPAR algorithm produces a low number of rules that show high confidence, comprehensibility, and interestingness; however, it requires modification of parameters under a large number of dimensions in datasets. The MOCANAR algorithm has produced reliable results with respect to all six parameters across all the data sets. The ACO-R algorithm produces high-quality rules but needs parameter modification for a large number of attributes in datasets and the MOB-ARM algorithm performs several times slower. Based on this analysis, we conclude that different SI-based NARM optimization algorithms best suit different needs. The investigations in this article are built on data collected by a preliminary study on the performance of SI-based NARM algorithms [25]. This analysis is valuable for bridging the artificial gaps between the optimization algorithms and developing the advanced framework for generalized association rule mining [38].

The following are the key contributions of this article.

- Investigating the role of multi-objective optimization algorithms, especially SI-based optimization algorithms for NARM.

- Presenting an exhaustive multi-aspect analysis of SI-based algorithms with four real-world datasets and six major metrics and objectives (performance time, the number of rules, support, confidence, comprehensibility, and interestingness).

- Providing efficient utilization of four popular SI-based optimization algorithms(MOPAR, MOCANAR, ACO-R and MOB-ARM) for NARM and discussing challenges associated with them.

The paper is structured as follows. In Sect. 2, related work is given. Sect. 3 highlights the background information to understand the subject. We discuss the SI-based algorithms in Sect. 4. Sect. 5 outlines the experimental results and multi-aspect analysis of the four SI-based algorithms. Sect. 6 provides the challenges and future directions for the algorithms. The conclusion is given in Sect. 7.

2. Related Work

In data mining [1], association rule mining (ARM) is a well-known technique to find interesting relations among various data items. Agrawal [2] introduced ARM in 1993 to discover the associations between data items in market basket analysis. Later, some essential algorithms, such as Apriori [3] and FP-Growth [22], were proposed. These algorithms were suitable for binary data but could not deal with numerical data. In 1996, Srikant introduced the concept of quantitative association rule mining (QARM) [41] to deal with numerical data. Further, this technique is also known as NARM [6]. Several methods, such as optimization, discretization, and distribution, are available in the literature to solve the problem of NARM [6, 28]. The optimization method seems to be a potential solution to deal with such complex problems. Evolutionary-based and SI-based algorithms come under the optimization method [27]. Recent NARM optimization algorithms also cover SI-based algorithms, which are based on animal, and insect movements and the biological behaviour of natural objects [17]. In recent decades, bio-inspired computation [14] has been one of the most researched subfields of artificial intelligence. SI-based algorithms are the subcategory of nature-inspired algorithms. Particle swarm optimization (PSO) [29], ant colony optimization (ACO) [15], cuckoo search [46], bat-inspired algorithm [45], crow search [8],and wolf search [43] are some examples of various SI algorithms. The variants of these algorithms were used for solving NARM problems. Such as in 2008, Alatas and Akin [4] used the PSO algorithm for mining the association rule with numeric attributes. The PSO was modified to search the numeric attributes' intervals and discover the numeric association rules. Further, Coello et al. [12] extended PSO to handle multi-objective issues. In the same way Makhlouf et al. [31] used the crow search-based algorithm for NARM. A multi-objective PSO technique using an adaptive archive grid for NARM was proposed by Kuo et al. [30]. It is based on the Pareto-optimal technique as well. Recently Stupan and Fister [42] presented a minimalistic framework *NiaARM* for NARM which is the extended version of the ARM-DE [16] algorithm. Users can preprocess their data using the *NiaARM* framework and use a variety of interest measures. In the literature, a performance analysis of several NARM algorithms was conducted. Altay et al. [6] analyzed the performance of seven evolutionary and fuzzy evolutionary NARM algorithms. The chosen algorithms were also compared against the Apriori algorithm. A comparative analysis was done in terms of support, confidence, the number of rules mined, the number of records covered, and time spent using eleven real-world datasets. This research found that the evolutionary algorithms have better results in terms of support, confidence, and time metrics. The authors also performed a performance analysis of multi-objective evolutionary NARM algorithms in [44]. In this research, six multi-objective and four single-objective optimization algorithms were chosen to be compared. The number of rules, coverage percentage, support, confidence, conviction, lift, netconf, ylesQ and certain factor measures were used for comparative analysis. Ten real-world datasets were used. This research found that multi-objective algorithms outperformed single-objective algorithms in terms of support, lift, certain factors, netconf, and yulesQ metrics. An example of using NARM for real-world problems was presented in [5], which performed an association analysis of multi-objective NARM algorithms using data about Parkinson's disease. This research used numerical data consisting of speech samples related to Parkinson's disease. This data was used on three multi-objective NARM algorithms to find association rules related to healthy individuals and patients with Parkinson's disease. The number of rules, coverage percentage, support, confidence, conviction, lift, netconfylesQ and certain factor measures were used for comparative analysis. Another example of using NARM for real-world problems was presented in [7], which presented an association analysis of multi-objective NARM algorithms using data about liver fibrosis. This data was used on two multi-objective NARM algorithms to find association rules related to liver fibrosis. The number of rules, coverage percentage, support, confidence, conviction, lift, netconf, ylesQ and certain factor measures were used for comparative analysis. After that, a sensitivity analysis was done to find the best parameters for this problem. A recent exhaustive review of more than five hundred nature-inspired metaheuristic algorithms and a performance assessment of fifteen algorithms has been conducted in [32].

3. Background

3.1. Association Rule Mining

ARM aims to extract interesting correlations, frequent patterns, or associations among sets of items in mainly transactional databases. One application of ARM is to find out what products are bought together from a store [2]. The discovered association rules can help determine how to boost the sales of a product, what products may be impacted by the discontinuation of another product, and the best locations for the products. Let $I = \{i_1, i_2, i_3, \dots i_m\}$ be a set of different m data items and D be a set of transactions where each transaction T contains a non-empty set of items, $T \subseteq I$. A transaction T contains X which is a set of some items in I if $X \subseteq T$. An association rule is an if-then relationship and denoted by $X \Rightarrow Y$ that has an antecedent X, and a consequent part Y, where $X \subset I$, $Y \subset I$ and $X \cap Y = \phi$ [3]. Support and confidence are the most commonly used measures in ARM. The support is calculated as the percentage of transactions of the total records containing both itemsets X and Y. The confidence of a rule is described as the percentage of transactions that contain itemset X also contain itemset Y.

3.2. Numerical Association Rule Mining

NARM came into the scenario to extract association rules from numerical data. Unlike a classical ARM, a numerical ARM allows attributes to be either categorical (e.g., gender, education) or numeric (e.g., salary, age) rather than just Boolean. A Numerical association rule is an implication of the form $X \Rightarrow Y$, in which both antecedent and consequent parts are the set of attributes in the forms $A = \{v_1, v_2, \dots v_n\}$ if A is a categorical attribute, or $A \, \epsilon \, [v_1, v_2]$ if A is numeric attribute.

An example of a numerical association rule is given below.

$$Age \; \epsilon \; [25, 35] \wedge Gender : [Male] \Rightarrow Salary \; \epsilon \; [2000, 2500]$$

$$(Support = 10\%, Confidence = 70\%)$$

This rule states that “10% of the employee are males aged between 25 and 35, and their salary would be between \$2,000 and \$2,500,” while “70% of males aged between 25 and 35 are earning between \$2,000 and \$2,500.” Here, *Age* and *Salary* are numerical attributes and *Gender* is a categorical attribute. In ARM, except for support and confidence, more than fifty measures of interestingness are available in the literature [39, 19]. This article mainly uses support, confidence, comprehensibility, and interestingness measures.

The support of an association rule $X \Rightarrow Y$ determines how frequently the itemset appears in a transactional database, shown in Eq. 1.

$$Supp(X \Rightarrow Y) = \frac{|(X \cup Y)|}{|D|} \tag{1}$$

The confidence of an association rule, shown in Eq. 2, determines how many transactions that contain X, also contain Y.

$$Conf(X \Rightarrow Y) = \frac{Supp(X \cup Y)}{Supp(X)} \tag{2}$$

According to [20], if the number of conditions involved in the antecedent part is less than the consequent part, the rule is more comprehensible. Eq. 3 is used to calculate the comprehensibility of an association rule. Here, $|Y|$ represents the number of attributes in the consequent part of the rule, and $|X \cup Y|$ shows the number of attributes in both the antecedent and consequent parts of the rule.

$$Comp(X \Rightarrow Y) = \frac{\log(1 + |Y|)}{\log(1 + |X \cup Y|)} \tag{3}$$

The interestingness measure is focused on discovering hidden information by extracting interesting rules. The Eq. 4 consists of three parts; the first part shows the probability of generating the rule based on the antecedent part, the second part shows the probability based on the consequent part and the third part shows the probability of not generating the rule based on the whole dataset.

$$interest(X \Rightarrow Y) = \frac{Supp(X \cup Y)}{Supp(X)} \frac{Supp(X \cup Y)}{Supp(Y)} \left(1 - \frac{Supp(X \cup Y)}{|D|}\right) \tag{4}$$

3.2.1. Multi-objective NARM

A single objective optimization problem has just one objective function; however, when many objective functions are used, the process is referred to as multi-objective [13]. Multi-objective optimization aims to balance several conflicting performance measures by using a set of non-dominated solutions [35]. The weighted sum and Pareto dominance are two methods for solving multi-objective optimization problems. The weighted sum method is a classical multi-objective method that summarizes multiple objectives into a single objective by multiplying each objective with a pre-defined weight. Traditional evolutionary algorithms optimize the resulting single-objective function. It is the simplest multi-objective method, but finding suitable multipliers can be challenging. However, in the Pareto dominance method, all the objectives are evaluated simultaneously. One solution dominates another if it improves one objective without causing a worse outcome for all the other objectives. Using this dominance criterion, non-dominated solutions can be defined.

4. Swarm Intelligence Optimization Algorithms

Optimization methods provide a robust and effective approach for massive search spaces and they are divided into biology-inspired and physics-based methods. Biology-based algorithms are further divided into SI and evolution-based, which is one of the widespread optimization methods [28, 6]. However, SI-based algorithms belong to the subset of bio-inspired algorithms [18], which, in turn, belong to the subcategory of nature-inspired algorithms.

According to [11], SI-based optimization methods are based on the collective intelligence of self-organized groups and the group behaviour of swarms, such as birds, fish, honey bees, and ant colonies. These algorithms are comprised of individuals who migrate throughout the search space over the simulated progression. Different SI-based algorithms are popular for different optimization problems. Some advanced SI algorithms have been developed recently for solving NARM problems. The most popular SI-based algorithms for NARM are Particle Swarm Optimization and Ant Colony Optimization. The *Bat Algorithm* and the *Cuckoo Search Algorithm* are also part of the family. The pseudocode of a nature-inspired meta-heuristic algorithm is given below in List 1. First, a population of agents is initialized with random solutions. The solutions are evaluated in terms of the used objectives. After that, each agent modifies its solution until a stopping criterion is met and the best-generated solutions are returned.

Listing 1: Pseudo code of nature-inspired meta heuristic algorithm

```
Step 1: Initialize the population
Step 2: Evaluate solutions
Step 3: For iteration in max iterations:
```

```
Step 4:      Modify solutions
Step 5:      Evaluate modified solutions
Step 6:      Select the best solutions
Step 7: Return the best solutions
```

Two key elements must be addressed when employing nature-inspired population-based algorithms to solve the ARM: (1) the representation of solutions in the search space and (2) fitness function assessment. The former describes the solution's encoding in the search space, whereas the latter is concerned with the quality of solutions. A solution must be encoded to mine numerical association rules in the search space. There are two well-known approaches to representing individuals: Michigan and Pittsburgh. When using the Michigan approach for representing individuals, each individual encodes a single association rule, while in the Pittsburgh approach, each individual encodes a set of association rules [17]. The Michigan approach is comparatively better than the Pittsburgh approach for finding high-quality rules. In the Michigan approach, different types of individual representation are identified. The first representation of encoding the association rules in NARM is shown in Eq. 5. The rule is encoded as a vector of attributes with n number of triplets, where n is the total number of attributes in the transactional database. Each triplet consists of three elements. ACN determines whether the attribute is present in the rule. ACN stands for antecedent, consequent and not present. LB determines the lower bound of the attribute and UB determines the upper bound of the attribute [17].

$$((ACN_1, LB_1, UB_1), ..., (ACN_n, LB_n, UB_n)) \tag{5}$$

Another way to represent a rule as a vector is shown in Eq. 6. Here, s shows the value and δ shows the standard deviation of the attribute [34].

$$((ACN_1, s_1, \delta_1), ..., (ACN_n, s_n, \delta_n)) \tag{6}$$

The ACN element can be encoded in two different ways. In a first way, shown in Eq. 7, if ACN value is less than or equal to 1/3, then the attribute is in the antecedent part of the rule. If the value is greater than 1/3 and smaller than or equal to 2/3, then the attribute is in the consequent part. If the value is greater than 2/3, then the attribute is not present in the rule.

$$j = \begin{cases} ACN_j \leq \frac{1}{3}, antecedent \\ \frac{1}{3} < ACN_j \leq \frac{2}{3}, consequent \\ ACN_j > \frac{2}{3}, not\ present \end{cases} \tag{7}$$

The second way to encode ACN is shown in Eq. 8. Here, if ACN is 1, then the attribute is in the antecedent part, if it is 2, then it is in the consequent part; and if it is 0, then the attribute is not present in the rule [26].

$$j = \begin{cases} ACN_j = 1, & antecedent \\ ACN_j = 2, & consequent \\ ACN_j = 0, & not\ present \end{cases} \tag{8}$$

Another way to represent an association rule is shown in Eq. 9, where n is the number of attributes in the database. Here, cp_i defines the cutting point between the antecedent and consequent attributes. If the $o_{i..n}$ value is zero, then the attribute is omitted from the rule; otherwise, it represents the id of the interval of the attribute. In this case, the database is discretized into intervals.

$$(cp_i, o_{i_1} ... o_{i,n}) \tag{9}$$

4.1. Multi-Objective Particle Swarm Optimization Algorithm

Particle swarm optimization (PSO) is the most popular optimization method for continuous non-linear functions, which simulates the movement of bird flocks or fish schools [29, 36]. As bird flocks move around in search of food in the sky and change their speed and position according to the group's direction and food availability, PSO simulates this behaviour artificially. A swarm is made up of N particles that move across in D dimensional search space. While searching, particles adjust their position by using the best position of their own *pbest* and by using the best position of the whole swarm *gbest*. The velocity and position of the particles are calculated iteratively and find the optimum solution.

4.1.1. MOPAR

The MOPAR is a multi-objective PSO (MOPSO) algorithm based on Pareto optimality for extracting numerical association rules in one step. The algorithm used three objectives: confidence, comprehensibility, and interestingness. For rule encoding, Eq. 5 and Eq. 7 are used.

Steps of the algorithm based on [9] are given in algorithm 1. The population consists of particles, the external repository consists of the mined rules, and the global best (the best particle) is initialized. In each iteration, the particle population is updated. After that, the best solutions from the population are added to the external repository, and the global best solution is updated. Finally, after the iterations, the external repository is returned. To update particles, Eq. 10 and Eq. 11 are used, which update the velocities and positions of a particle. After that, the particle's objectives are evaluated. Finally, the local best solution of each particle is updated using Pareto dominance.

$$v_{i,k}(t{+}1) = w(t)v_{i,k}(t){+}c_1R_1\big(lbest_{i,k}(t){-}x_{i,k}(t)\big){+}c_2R_2\big(gbest_{i,k}(t){-}x_{i,k}(t) \tag{10}$$

$$x_{i,k}(t+1) = x_{i,k}(t) + v_{i,k}(t+1) \tag{11}$$

To find the global best solution, roulette wheel selection is used. The roulette wheel first assigns a rank to each particle using Eq. 12, in which *xRank* is a user-specified parameter and local dominated count is the number of a particle's local best solutions that the current solution dominates. After that, each particle is assigned a probability based on Eq. 13. Based on these probabilities, a particle is chosen.

$$rank_i(t) = \frac{xRank}{\text{local dominated\ \ count}} \tag{12}$$

$$Prob_i(t) = \frac{rank_i(t)}{\sum_{k=1}^{n} rank_k(t)} \tag{13}$$

The MOPAR develops a MOPSO that provides a redefinition of *lbest* and *gbest* particles and a selection procedure to handle the problem of numerical ARM. In this algorithm, the particle has the same representation as $RPSOA$ and has lower and upper bounds of intervals.

4.2. Cuckoo Search Algorithm

The Cuckoo search algorithm (CSA) was proposed by Yang and Deb in 2009 [46]. CSA is inspired by the brooding parasitic behaviour of cuckoo species. Cuckoo birds do not build nests and instead lay eggs in the nests

Algorithm 1 MOPAR algorithm steps.

```
Input: Data, population size, maximum iterations, external repository size,
c1, c2, inertia weight, xRank
Output: External repository
 1: Initialize population.
 2: Evaluate the objectives of the generated rule.
 3: Initialize external repository.
 4: Initialize global best.
 5: Update the velocities of particles.
 6: Update the positions of particles and evaluate the objectives of the new
    rules.
 7: Update the non-dominated local set of each particle. Update local best.
 8: Update external repository. If the size of the external repository is bigger
    than the external repository size, then particles that dominate more rules
    are removed.
 9: Update global best by using the roulette wheel selection. If the maximum
    number of iterations is not reached, go to step 2.
10: return the external repository.
```

of other bird species. The cuckoo bird has the special ability to mimic the colour and pattern of other birds' eggs. It is possible that some of the host birds know about the stranger's eggs and throws them, or they can leave their nest. The three rules are followed for describing the cuckoo search algorithm. The first rule is that cuckoo birds lay only one egg at a time in randomly chosen nests. According to the second rule, the nest with high-quality eggs will carry over to the next generation. In the third rule, the number of host nests is fixed, and the probability of discovering the cuckoo eggs by the host bird is either 0 or 1. If the host finds so, it can destroy the egg or quit the nest. Each egg in the nest represents a solution, and the egg laid by the cuckoo denotes a new solution, and the goal is to use the new and possibly better solution to replace the less interesting solution in the nest. In the direction of ARM, k used the concept of a multi-objective cuckoo search algorithm for NARM using a Pareto-based approach [26].

4.2.1. MOCANAR

The MOCANAR [26] is a multi-objective cuckoo search algorithm based on Pareto principles that derive high-quality association rules from numeric attributes. This algorithm mimics the brooding parasitic behaviour of cuckoo species. A 2D array for ARM represents the cuckoos. The columns represent the attributes in the dataset, and the number of rows is three. The first row represents the attribute's location; the second row consists of the lower bound of the attribute, and the third row represents the upper bound of the attribute. The 0 value of the first row indicates that the related attribute is not present in the rule. In contrast, value 1 shows that the related attribute belongs to the antecedent part of the rule, and value 2 shows that the concerned attribute belongs to the consequent part of the rule. The MOCANAR considers the following objectives: support, confidence, interest, and comprehensibility. The rules were retrieved incrementally, with a small number of high-quality rules being produced for each iteration of the method.

For rule encoding, Eq. 5 and Eq. 8 are used. Pareto optimality is used for extracting non-dominated rules. Algorithm 2 shows the steps of the algorithm and is based on [26]. In each increment, the population, which consists of cuckoos, and current non-dominated rules, are initialized. In each generation, random cuckoos are generated and directed towards the best solution, using levy flight policy [46] and replaced with the worst cuckoos in the population. After that, each cuckoo generates an egg using levy flight. At the end of each generation, current non-dominated rules are updated. At the end of each increment, the final non-dominated rules are updated. Finally, the final non-dominated rules are returned. A tournament is used to choose the best solution when generating eggs. For this, a number of tournament cuckoos are selected randomly from the population, and a random non-dominated solution from this selection is returned.

A levy flight policy is used to direct cuckoos towards the best cuckoo. For each attribute of a source cuckoo's rule, three-step sizes are calculated using levy distribution and a target cuckoo. Based on these step sizes, the rule of a source cuckoo is modified. To generate a new population, first, a percentage of eggs that have the worst support measure is eliminated. After that, the eggs and cuckoo population is merged into a temporary population. The temporary population is sorted in terms of support measure, and 1/4 of the highest-ranking solutions are added to the new population. The same is done for the rest of the measures, after which a new population has been formed.

Algorithm 2 MOCANAR algorithm steps.

```
Input: Data, population size, number of increments, maximum generations,
pa, pmut, number of tournaments, number of random cuckoos, w1, w2, w3
Output: Final non-dominated rules

1: Initialize population and cuckoo eggs. Evaluate the objectives of generated rules.
2: Generate random cuckoos and direct them toward the best cuckoo in the
   population. Replace the worst cuckoo in the population with the directed cuckoo.
3: Generate cuckoo eggs by directing all cuckoos in the population toward
   the best cuckoo. The best cuckoo is chosen for the tournament.
4: A percentage of the worst eggs in terms of the support measure is eliminated.
   A new population is formed by choosing the cuckoos with the best objective measures.
5: Population and non-dominated lists are merged, and duplicated rules are
   deleted. Non-dominated rules from the merged list are assigned to the
   non-dominated list.
6: If the maximum number of generations is not reached, go to step 2.
7: Rules from the non-dominated list are added to the final non-dominated list.
8: If the maximum number of increments is not reached, go to step 1.
9: Duplicated and dominated rules are removed from the final non-dominated.
10: return the final non-dominated list.
```

4.3. Ant Colony Optimization Algorithm

Ant colony optimization (ACO) is based on the foraging behaviour of various ant species. Ants begin to investigate the area around the nest at random and eventually find some food sources. Based on the quantity and quality of food, these ants deposit chemical pheromones on the ground to suggest the desired path for colony members to follow on their return trip [15]. In ACO, a group of artificial ants develops solutions to the optimization problem and communicates information about the quality using a communication mechanism similar to real ants.

4.3.1. ACO-R

The ACO-R algorithm is an ant colony optimization for numeric values and retrieves association rules for numeric attributes without minimum support and minimum confidence thresholds. ACO uses a discrete probability distribution, while ACO-R uses a probability density function. ACO stores pheromone information in the pheromone table, whereas ACO-R, describes the pheromone distribution over the search space using a solution archive size of k. The ants in ACO-R move across the archive, selecting one row depending on its associated weight (ω). Then a new solution is created by sampling the Gaussian function g of the values of each dimension in the selected solution. Each numeric attribute is one dimension of the solution archive, divided into three sections, with each complete solution regarded as a numeric association rule. The first part of the solution reflects the rule's antecedent or consequence. The second part represents the solution's value. The third part shows the solution's standard deviation, which is used to construct numeric attribute intervals. This algorithm uses Gaussian functions to identify attribute intervals that correspond to an interesting rule, with the function determining the frequency and length of the intervals. There are four components to this objective function as given in Eq. 16. The first section can be seen as support for the rule, which is the importance of an association rule. The second part is known as the confidence value. The rule's third section is the number of attributes. The amplitude of the intervals that adhere to the itemset and rules is penalized using the last part of the objective function. The pheromone update technique adds a number of new solutions, each made by one ant, and removes the same number of bad solutions from the archive after ranking the solutions to keep track of the solutions. As a result, the best solutions are always at the top of the solution archive, and the best solution in each ACO-R execution can be thought of as a rule.

For rule encoding, Eq. 6 and Eq. 8 are used. Algorithm 3 shows the steps of the algorithm and is based on [34]. First, the archive, which consists of solutions, is initialized, and solutions are ranked. In each iteration, the weights and probabilities of solutions are calculated. Each ant chooses a solution based on the assigned probabilities and generates a new solution by sampling a Gaussian function. At the end of each iteration, the solutions in the archive are ranked and the worst solutions are removed. After the iterations, the archive is returned.

Algorithm 3 ACO-R algorithm steps.

Input: Data, archive size, ant colony size, maximum iterations, alpha1, alpha2, alpha3, alpha4, alpha5, q, e
Output: Archive

```
Initialize and sort the archive.
Initialize weights, probabilities, and ants.
Ant chooses a solution and generates a new solution by sampling a Gaussian function. The objectives of the new solution are evaluated.
If the number of ants that have generated a new rule is not reached, go to step 3.
Add ant-generated solutions to archive, sort and cut off
If the number of iterations is not reached, go to step 2.
Delete duplicated rules from the archive.
return non-dominated rules from the archive.
```

The interval objective, shown in Eq. 14, favours rules with smaller intervals. Here, n is the number of attributes, max bound and min bound is the maximum and minimum values for the attribute in the database. UB_i and LB_i are the upper and lower bounds of an attribute in the rule. The upper and lower bound of the intervals can be calculated by adding a coefficient of a standard deviation to the value of solution s_j^i using Eq. 15.

$$int = \sum_{i=0}^{n} \frac{(UB_i - LB_i)}{maxbound_i - minbound_i} \tag{14}$$

$$UB_i = s_j^i + \alpha_5\sigma \qquad and \qquad LB_i = s_j^i - \alpha_5\sigma \tag{15}$$

All the mentioned objectives are put together into a single objective function, shown in Eq. 16. Here α_1, α_2, α_3, and α_4 are input parameters of the algorithm.

$$objective = \alpha_1 \cdot supp + \alpha_2 \cdot conf - \alpha_3 \cdot inter - \alpha_4 \cdot int \tag{16}$$

$$\omega_j = \frac{1}{qk\sqrt{2\pi}} e^{\frac{-(j-1)^2}{2q^2k^2}} \tag{17}$$

To calculate the weight ω_j of a solution S_j, Eq. 17 is used, where k is the number of solutions in the archive, j is the rank of the solution, and q is a user-specified parameter. If q is small, best-ranked solutions are more preferred, and if it is large, the probability is more uniform [40].

To calculate the probability of choosing solution S_j, Eq. 18 is used, where the weight of a solution is divided by the sum of all the weights of the solutions.

$$p_j = \frac{\omega_j}{\sum_{r=1}^{k} \omega_r} \tag{18}$$

After an ant chooses a solution based on the probabilities, a new solution is sampled using Eq. 19. Here, δ is calculated using Eq. 20 and μ is the value of the chosen solution.

$$P(x) = g(x, \mu, \delta) = \frac{1}{\delta\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\delta^2}} \tag{19}$$

In Eq. 20, ξ is a user-specified parameter. The higher it is, the lower the convergence speed of the algorithm. Parameter k is the number of solutions in the archive, and s_j^i is the value of the chosen solution.

$$\delta = \xi \sum_{r=1}^{k} \frac{s_r^i - s_j^i}{k-1} \tag{20}$$

4.4. BAT Algorithm

Yang (2010) introduced the BAT algorithm (BA) to address continuous constrained optimization problems based on the echolocation behaviour of microbats [45]. Bats have the special characteristics to use echolocation to sense distance. Microbats use echolocation to discover prey, avoid obstacles, and find roosting nooks in the dark. These bats produce an extremely loud sound pulse and listen for the echo reflected back from the objects in their surroundings. The BA is based on the velocity of a bat at a particular position, with a fixed frequency and varying wavelength and loudness. BA was used for ARM for dealing with categorical attributes [23].

4.4.1. MOB-ARM

A multi-objective bat algorithm for NARM was proposed by Heraguemi et al. [24]. This algorithm is based on the behaviour of microbats. The authors used four quality measures: support, confidence, comprehensibility, and interestingness, and defined two global objective functions for optimization to extract interesting rules. The first objective function consists of support and

Algorithm 4 MOB-ARM algorithm steps.

```
Input: Data, population size, iterations, Pareto points, alpha, beta, gamma, delta, minimum support, minimum confidence
Output: Non-dominated solutions
Initialize population. Sort population. Initialize global best. Initialize the non-dominated solutions list.
Initialize weights.
Update every bat's frequency, velocity and generate a new rule.
If the random number is bigger than the bat's rate, change one attribute in the new rule.
Check and fix rule. Evaluate fitness.
If the new objective is bigger than the old objective, then accept the new rule, increase rate, and decrease the loudness of the bat.
Sort population. Update global best.
If the number of iterations is not reached, go to step 3.
Add best solutions to non-dominated solutions
If the number of Pareto points is not reached, go to step 2.
return non-duplicated rules from the non-dominated solutions list.
```

confidence (Eq. 23), and the second objective function consists of comprehensibility and interestingness (Eq. 24). The algorithm flows in three main steps: initialization, searching for the non-dominance solution for the Pareto point, and searching for the best solution for each bat at the Pareto point. This algorithm uses the Michigan approach for encoding the rule. The bats are initialized with a random frequency and velocity.

For rule encoding, Eq. 9 is used. Prior to using the algorithm, data is discretized into intervals. The weighted sum is used to determine the best solutions. Algorithm 4 presents the steps and is based on [24]. First, the population, which consists of bats, is initialized. In each iteration, objective weights are generated, and each bat's frequency, velocity and rules are updated. At the end of each iteration, the bats are ranked, and a new global best solution is chosen. After each iteration, each bat's best solution is recorded as a non-dominated solution. Finally, the non-dominated solutions are returned. To generate weights, Eq. 21 is used. Here, k is the number of objectives used, which in MOB-ARM is two. The weights are used for calculating an objective measure shown in Eq. 22, which uses two objectives.

$$\sum_{k\,=\,1}^{k} \; w_k = 1 \tag{21}$$

$$Obj(R) = w_1 \; Obj_1(R) + w_2 \; Obj_2(R) \tag{22}$$

The objective measure uses two separate objectives, which are calculated using Eq. 23 and Eq. 24. The equations use user-specified parameters α, β, γ and δ as weights for the support, confidence, comprehensibility, and interestingness measures.

$$Obj_1(R) = \alpha conf(R) + \frac{\beta supp(R)}{\alpha} + \beta \tag{23}$$

$$Obj_2(R) = \gamma Comp(R) + \frac{\delta Inter(R)}{\gamma} + \delta \tag{24}$$

To update a bat's frequency and velocity, Eq. 25 and Eq. 26 are used. First, the new frequency is calculated using a maximum frequency, which is the number of attributes in the dataset. After that, a new velocity is calculated using the maximum frequency, new frequency, and the previous velocity.

$$f_i^t = 1 + (f_{max})\beta \tag{25}$$

$$v_i^t = f_{max} - f_i^t - v_i^{t-1} \tag{26}$$

$$A_i^{t+1} = \alpha A_i^t \tag{27}$$

$$r_i^{t+1} = r_i^0[1 - \exp{(-\gamma t)}] \tag{28}$$

A new rule is generated using an algorithm proposed in [23]. The rules are generated based on the velocity, frequency and loudness of the bat. Velocity determines the starting position of the change in the rule, and frequency determines how many attributes are changed. If the loudness of the bat is less than a random number, the attribute value at index velocity is increased; otherwise, it is decreased. If the value goes out of bounds, it is set to zero. After the rule is generated, one item in the rule is changed if a random number is bigger than the rate.

If the new rule's objective is better than the old objective, the rule is accepted, and the loudness and rate of the bat are updated. Loudness is decreased by using Eq. 27. The rate is increased using Eq. 28, where r_i^0 is the initial rate of the bat and t is the current iteration [45].

Table 1: Datasets used in the experiments

Dataset	#records	#attributes	Description
Basketball	96	5	This dataset includes a variety of numerical attributes related to the performance of basketball teams and players to identify patterns and relationships.
Quake	2178	4	This dataset is used to demonstrate the use of various smoothing techniques in statistics. The dataset contains a time series of the number of earthquakes that occurred in California between the years 1980 and 1984.
Fat	225	18	This dataset contains the percentage of body fat, age, weight, height, and 10 measurements of body circumference (such as the abdomen) for a total of 252 men.
Longley	16	7	The Longley dataset comprises a number of strongly collinear US macroeconomic indicators. It has been used to assess the precision of least squares methods.

5. Experimental Results

In experimentation and to evaluate the performance of the MOPAR, MOCANAR, ACO-R, and MOB-ARM algorithms, four real-world datasets are selected from Guvenir et al. [21]. A detailed description of these datasets is given in Table 1. These datasets have different numbers of records and attributes, which helps in providing a more accurate evaluation of how well the implementations perform when dealing with various characteristics. All

Table 2: Algorithmic parameters used in the experiment

Algorithms	Parameters
MOPAR [9]	Population size: 50, iterations: 200, external repository size: 50, inertia weight: 0.63, velocity: 3.83, xRank: 13.3, c1:2, c2: 2
MOCANAR [26]	Population size: 50, generations: 200, increments: 1, randomcuckoo: 1, tournament: 30, Pa: 0.3, P_mut: 0.05, w1: 0.2, w2: 0.5, w3: 0.3
MOB-ARM [24]	Population size: 50, iterations: 40, Pareto points: 5, alpha: 0.4, beta: 0.3, gamma: 0.2, delta: 0.1, minsupp: 0.2, minconf: 0.5
ACO-R [34]	Ant colony size: 50, iterations: 200, archive size: 50, alpha1, alpha2: 4, alpha3, alpha5: 1, alpha4: 0.001, Q: 0.1, E: 0.85

Table 3: Average support of the MOPAR, MOCANAR, ACO-R and MOB-ARM algorithms with respect to the 'Basketball', 'Quake', 'Fat' and 'Longley' datasets

Datasets	MOPAR [9]	MOCANAR [26]	ACO-R [34]	MOB-ARM [24]
Basketball	0.13	0.49	0.41	0.28
Fat	0.08	0.63	0.01	0.34
Quake	0.22	0.51	0.57	0.45
Longley	0.10	0.29	0.35	0.28

the experiments are performed using an Intel Core i7-10510U machine with 16 GB of memory and running Windows 10. Table 2 lists the parameters used in this experiment for the four algorithms. However, to evaluate the algorithms under equal conditions, the population size is fixed at 50, and the number of iterations is set at 200 for all the algorithms. For the MOPAR algorithm, the external repository size is 50 and the other parameters are taken from Beiranvand et al. [9]. The parameters used for ACO-R are decided via testing because the author of the algorithm, Moslehi et al. [34], did not specify the best parameters. For the MOB-ARM algorithm, the number

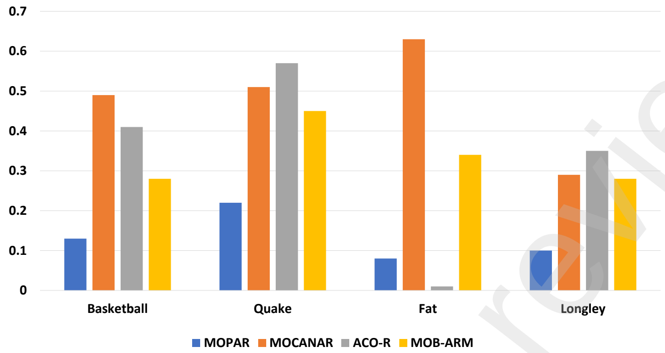


Figure 1: Average support of MOPAR, MOCANAR, ACO-R and MOB-ARM algorithms with respect to the 'Basketball', 'Quake', 'Fat' and 'Longley' datasets

Table 4: Average confidence of the MOPAR, MOCANAR, ACO-R and MOB-ARM algorithms with respect to the 'Basketball', 'Quake', 'Fat' and 'Longley' datasets

Datasets	MOPAR [9]	MOCANAR [26]	ACO-R [34]	MOB-ARM [24]
Basketball	0.78	0.80	0.80	0.63
Fat	0.48	0.87	0.86	0.72
Quake	0.71	0.84	0.87	0.72
Longley	0.94	0.93	0.99	0.92

Table 5: Average generated rules by the MOPAR, MOCANAR, ACO-R and MOB-ARM algorithms with respect to the 'Basketball', 'Quake', 'Fat' and 'Longley' datasets

Datasets	MOPAR [9]	MOCANAR [26]	ACO-R [34]	MOB-ARM [24]
Basketball	11.2	32.8	40.4	8.4
Fat	10.4	54.6	13.8	7.8
Quake	18.6	22.2	47	8.4
Longley	16.2	8.8	8.4	20.6

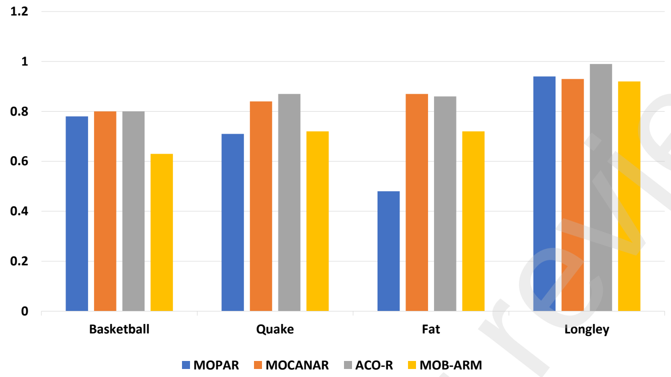


Figure 2: Average confidence of the MOPAR, MOCANAR, ACO-R and MOB-ARM algorithms with respect to the 'Basketball', 'Quake', 'Fat' and 'Longley' datasets

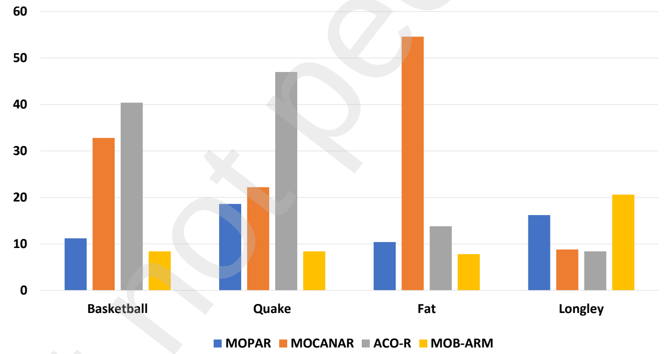


Figure 3: Average generated rules by the MOPAR, MOCANAR, ACO-R and MOB-ARM algorithms with respect to the 'Basketball', 'Quake', 'Fat' and 'Longley' datasets

of iterations and Pareto points is set as 40 and 5, respectively. For each dataset, all the algorithms are tested five times. The programming code is available in the GitHub[1] repository.

[1] https://github.com/rahul-sharmaa/Performance-Analysis-of-SI-based-Algorithms.git

Table 6: Average time (in seconds) spent by the MOPAR, MOCANAR, ACO-R and MOB-ARM algorithms with respect to the 'Basketball', 'Quake', 'Fat' and 'Longley' datasets

Datasets	MOPAR [9]	MOCANAR [26]	ACO-R [34]	MOB-ARM [24]
Basketball	455.4	404.9	442	1181.92
Fat	1259.28	1469.22	1173.26	3345.4
Quake	361	424.04	402.16	1253.42
Longley	500.28	545.08	604.52	1539.3

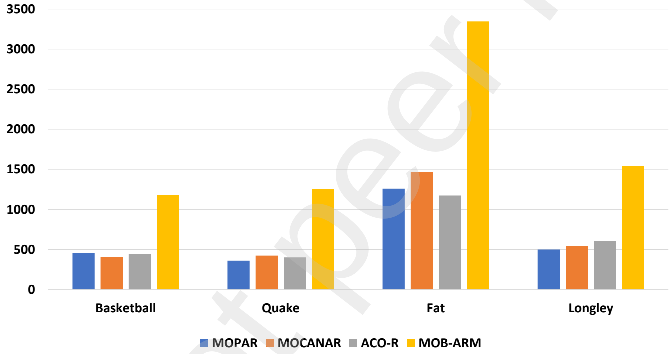


Figure 4: Average time (in seconds) spent by the MOPAR, MOCANAR, ACO-R and MOB-ARM algorithms with respect to the 'Basketball', 'Quake', 'Fat' and 'Longley' datasets

Table 3 demonstrates the average support values from all four SI algorithms over four datasets. Fig. 1 shows the average support values of the rules mined by the algorithms. The MOCANAR produced high support rules in most of the datasets; however, ACO-R produced rules with high support in the *Basketball*, *Quake*, and *Longley* datasets but underperformed in the *Fat* dataset. MOB-ARM had average support measures in all datasets, while MOPAR had the overall lowest support values.

The average confidence values obtained from the SI algorithms within four datasets are given in Table 4. Fig. 2 represents the average confidence values of the rules mined by the algorithms. MOCANAR and ACO-R had the

Table 7: Average comprehensibility of the MOPAR, MOCANAR, ACO-R and MOB-ARM algorithms with respect to the 'Basketball', 'Quake', 'Fat' and 'Longley' datasets

Datasets	MOPAR [9]	MOCANAR [26]	ACO-R [34]	MOB-ARM [24]
Basketball	0.82	0.67	0.62	0.62
Fat	0.83	0.69	0.75	0.62
Quake	0.71	0.66	0.63	0.64
Longley	0.90	0.75	0.55	0.70

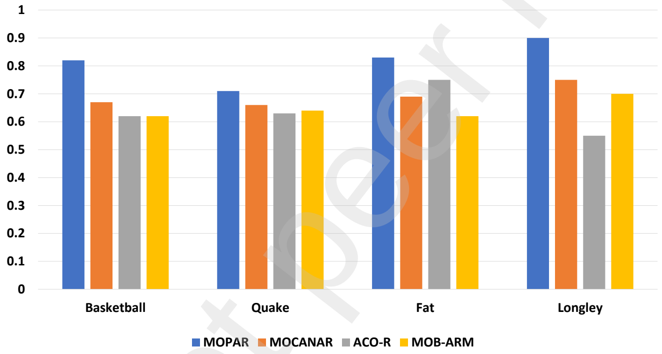


Figure 5: Average comprehensibility of the MOPAR, MOCANAR, ACO-R and MOB-ARM algorithms with respect to the 'Basketball', 'Quake', 'Fat' and 'Longley' datasets

Table 8: Average interestingness of the MOPAR, MOCANAR, ACO-R and MOB-ARM algorithms with respect to the 'Basketball', 'Quake', 'Fat' and 'Longley' datasets

Datasets	MOPAR [9]	MOCANAR [26]	ACO-R [34]	MOB-ARM [24]
Basketball	0.43	0.24	0.25	0.24
Fat	0.15	0.21	0.54	0.27
Quake	0.16	0.24	0.24	0.22
Longley	0.84	0.65	0.56	0.59

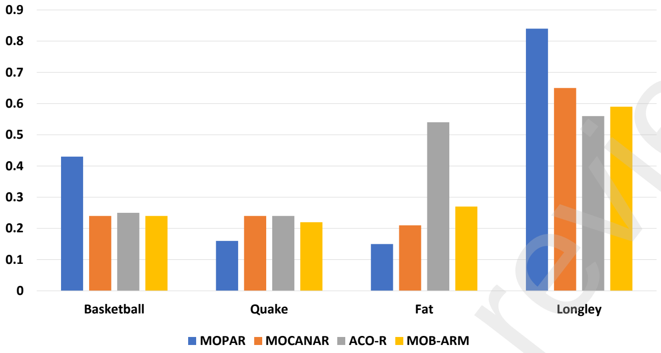


Figure 6: Average interestingness of the MOPAR, MOCANAR, ACO-R and MOB-ARM algorithms with respect to the 'Basketball', 'Quake', 'Fat' and 'Longley' datasets

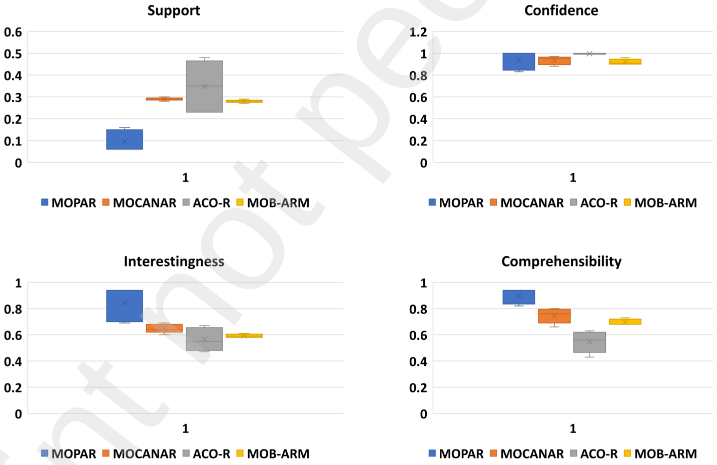


Figure 7: Boxplots of the support, confidence, comprehensibility and interestingness values for the 'Longley' dataset

highest results across all datasets. Compared to other algorithms, MOPAR and MOB-ARM have average results but produced the lowest confidence in the *Fat* and *Basketball* datasets, respectively.

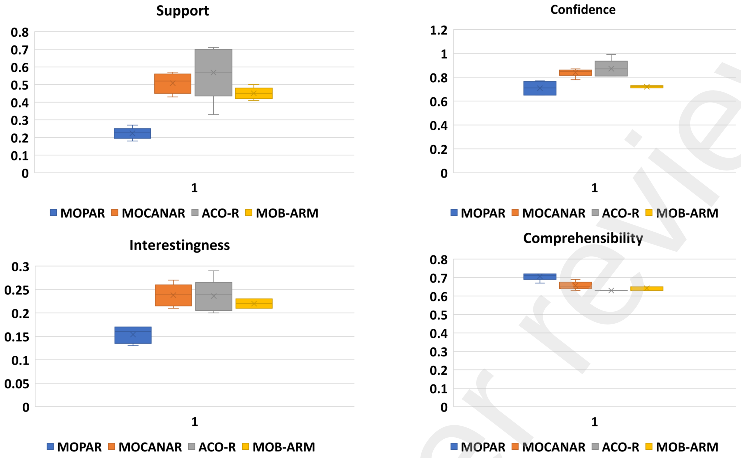


Figure 8: Boxplots of the support, confidence, comprehensibility and interestingness values for the 'Quake' dataset

The average generated association rules by algorithms for each dataset have been presented in Table 5. It is clear from the table that MOCANAR and ACO-R mined the most rules across all datasets. MOPAR and MOB-ARM mined the least rules across all datasets except *Longley* dataset. Fig. 3 shows the average number of rules mined by the algorithms.

Table 6, 7, and 8 demonstrates the average time spent by the algorithms, the average comprehensibility values, and the average interestingness values of the rules mined by the algorithms, respectively. However, Table 9 presents the comparative analysis of all four algorithms with four datasets. Fig. 4 showcases that MOPAR, MOCANAR and ACO-R had approximately similar results for all datasets. MOB-ARM stood multiple times slower than the other algorithms. MOPAR produced the highest comprehensibility measures for all datasets, which is shown in Fig. 5. MOCANAR, ACO-R and MOB-ARM had similarly average results across all datasets. Fig. 6 demonstrates the average interestingness values of the rules mined by the algorithms. MOPAR produced the highest results for *Basketball* and *Longley* datasets but the lowest results for *Quake* and *Fat* datasets. ACO-R has the highest interestingness measure for *Fat* dataset. MOCANAR and MOB-ARM produced average interestingness results for all the datasets.

Table 9: Comparative experimental results for the 'Basketball', 'Quake', 'Fat' and 'Longley' datasets

Datasets	Algorithms	Time(sec)	Avg. rules	Avg. Supp.	Avg. Conf.	Avg. Comp.	Avg. Int.
Basketball	MOPAR	455.4	11.2	0.13	0.78	0.82	0.43
	MOCANAR	404.9	32.8	0.49	0.80	0.67	0.24
	ACO-R	442	40.4	0.41	0.80	0.62	0.25
	MOB-ARM	1181.92	8.4	0.28	0.63	0.62	0.24
Quake	MOPAR	361	18.6	0.22	0.71	0.71	0.16
	MOCANAR	424.04	22.2	0.51	0.84	0.66	0.24
	ACO-R	402.16	47	0.57	0.87	0.63	0.24
	MOB-ARM	1253.42	8.4	0.45	0.72	0.64	0.22
Fat	MOPAR	1259.28	10.4	0.08	0.48	0.83	0.15
	MOCANAR	1469.22	54.6	0.63	0.87	0.69	0.21
	ACO-R	1173.26	13.8	0.01	0.86	0.75	0.54
	MOB-ARM	3345.4	7.8	0.34	0.72	0.62	0.27
Longley	MOPAR	500.28	16.2	0.10	0.94	0.90	0.84
	MOCANAR	545.08	8.8	0.29	0.93	0.75	0.65
	ACO-R	604.52	8.4	0.35	0.99	0.55	0.56
	MOB-ARM	1539.3	20.6	0.28	0.92	0.70	0.59

Table 10: The Average values of six measures across all datasets

Algorithms	Time(sec)	Avg. rules	Avg. Supp.	Avg. Conf.	Avg. Comp.	Avg. Int.
MOPAR	644	14	0.13	0.72	0.81	0.39
MOCANAR	710	29	0.48	0.86	0.69	0.33
ACO-R	655	27	0.33	0.88	0.64	0.40
MOB-ARM	1830	11	0.34	0.75	0.64	0.33

Boxplots for the confidence, support, interestingness and comprehensibility for *Longley* and *Quake* datasets are given in Fig. 7 and Fig. 8. We have considered these datasets because, out of the four datasets, *Longley* has the lowest number of records, and *Quake* has the highest number of records. For *Longley* dataset, MOPAR has given the best results under comprehensibility and interestingness measures. The algorithms performed very closer to each other in terms of confidence, but ACO-R gave the best result. It is observed that algorithms do not perform well in terms of support, although ACO-R has a better average support value in comparison to other algorithms.

When comparing the algorithms for interestingness measure with *Quake* dataset, MOCANAR, ACO-R and MOB-ARM have given nearly similar values. However, the MOPAR has given significantly different results for interestingness and achieved a better result in terms of comprehensibility. The ACO-R has given the best results in terms of confidence and support values.

In terms of measures, Table 10 presents the average of the results of the four SI-based NARM algorithms across four data sets. After the evaluation of Table 10, it can be observed that none of the algorithms gave the best results in terms of all six measures. However, MOPAR performed best in terms of average time, average comprehensibility, and average interestingness, and ACO-R performed best in terms of average confidence and average interestingness measures. MOCANAR generated the best average support value and the highest average number of rules.

6. Future Directions

SI-based algorithms have been used in NARM to optimize traditional data mining algorithms. However, several issues and challenges need to be addressed to make SI-based algorithms more effective. For a fair comparison among different algorithms, stopping criteria is also one of the important factors to be considered. Ravber et al. [37] have raised this issue and concluded that the maximum number of generations as a stopping criterion is harmful and not it is not recommended for a fair comparison of the optimization algorithms. However, In this paper, we have also used the maximum generation as a stopping criterion, as per the original settings proposed under the algorithms. Therefore, for future direction, this is one of the important factors to be considered.

Next, scalability and premature convergence are also the main issues with optimization algorithms. These algorithms require significant computational resources and memory, which makes them less practical for larger datasets. Premature convergence can occur when the algorithm's parameters are not set correctly or the search space is too small, leading to sub-optimal solutions.

Parameter tuning is another challenge, as it can be time-consuming and requires expertise in the field. Additionally, SI-based algorithms are often robust to noisy and incomplete data; they can still be sensitive to certain types of noise or outliers in the data. Incorporating domain knowledge into the algorithm can improve performance and interpretability, but this requires additional efforts and expertise in the field.

In future, addressing these important issues and challenges will be an important factor for the continued development of SI-based algorithms for NARM.

7. Conclusion

This paper presents an exhaustive multi-aspect analysis of four SI-based algorithms for NARM. The algorithms are experimented with four real-world datasets and analysed with six major parameters, i.e., time, average support, average confidence, the average number of rules, average comprehensibility, and average interestingness. The experiments and the analysis demonstrate that the MOB-ARM algorithm yields the worst results in terms of average time spent on all datasets, but MOPAR, MOCANAR and ACO-R algorithms performed well. However, when the average comprehensibility value of the rules produced by the algorithms is examined, MOPAR provides the best result across all the data sets. The MOPAR algorithm has a low support value and can produce a low number of rules with high confidence and interestingness measures. Still, it needs parameter modification for datasets with a larger number of attributes or instances. The MOCANAR algorithm can be used to generate rules with consistent outcomes in all metrics across all datasets. The ACO-R generates high-quality rules, but it underperforms for support under the *Fat* dataset and requires parameter modification when applied to datasets with more attributes. MOB-ARM produced a few rules with average results across all datasets, but it was much slower than the other three algorithms. The performance of the MOB-ARM algorithm can be improved by eliminating the discretization step to produce more rules which is also helpful to reduce the time complexity of the algorithm. The overall results demonstrated that no single SI-based algorithm is a perfect fit for efficient NARM, and each SI-based algorithm has its own drawback, therefore, a combination of algorithms for different metrics and objectives is suggested to be utilized for efficient NARM.

Acknowledgments

This work has been partially conducted in the project “ICT programme” which was supported by the European Union through the European Social Fund.

Declarations

Conflict of interest Authors declare that they have no conflict of interest.

References

[1] Agrawal, R., Imielinski, T., Swami, A., 1993a. Database mining: a performance perspective. IEEE Transactions on Knowledge and Data Engineering 5, 914–925.

[2] Agrawal, R., Imieliński, T., Swami, A., 1993b. Mining association rules between sets of items in large databases. ACM SIGMOD Record 22, 207–216.

[3] Agrawal, R., Srikant, R., 1994. Fast algorithms for mining association rules in large databases, in: Proceedings of VLDB'1994 – the 20th International Conference on Very Large Data Bases, Morgan Kaufmann, San Francisco, CA, USA. p. 487–499.

[4] Alatas, B., Akin, E., 2008. Rough particle swarm optimization and its applications in data mining. Soft Computing 12, 1205–1218.

[5] Altay, E.V., Alatas, B., 2020a. Association analysis of Parkinson disease with vocal change characteristics using multi-objective metaheuristic optimization. Medical Hypotheses 141, 109722.

[6] Altay, E.V., Alatas, B., 2020b. Intelligent optimization algorithms for the problem of mining numerical association rules. Physica A: Statistical Mechanics and its Applications 540, 123142.

[7] Altay, E.V., Alatas, B., 2020c. A novel clinical decision support system for liver fibrosis using evolutionary multi-objective method based numerical association analysis. Medical Hypotheses 144, 110028.

[8] Askarzadeh, A., 2016. A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm. Computers & Structures 169, 1–12.

[9] Beiranvand, V., Mobasher-Kashani, M., Bakar, A.A., 2014. Multi-objective pso algorithm for mining numerical association rules without a priori discretization. Expert systems with applications 41, 4259–4273.

[10] Blum, C., Li, X., 2008. Swarm intelligence in optimization, in: Swarm intelligence. Springer, pp. 43–85.

[11] Bonabeau, E., Dorigo, M., Theraulaz, G., 1999. Swarm Intelligence: From Natural to Artificial Systems. Oxford University Press.

[12] Coello, C.A.C., Pulido, G.T., Lechuga, M.S., 2004. Handling multiple objectives with particle swarm optimization. IEEE Transactions on evolutionary computation 8, 256–279.

[13] Deb, K., 2011. Multi-objective Optimisation Using Evolutionary Algorithms: An Introduction. Springer London, London. chapter 3. pp. 3–34.

[14] Del Ser, J., Osaba, E., Molina, D., Yang, X.S., Salcedo-Sanz, S., Camacho, D., Das, S., Suganthan, P.N., Coello Coello, C.A., Herrera, F., 2019. Bio-inspired computation: Where we stand and what's next. Swarm and Evolutionary Computation 48, 220–250.

[15] Dorigo, M., Birattari, M., Stutzle, T., 2006. Ant colony optimization. IEEE Computational Intelligence Magazine 1, 28–39.

[16] Fister, I., Iglesias, A., Galvez, A., Del Ser, J., Osaba, E., Fister, I., 2018. Differential evolution for association rule mining using categorical and numerical attributes, in: Yin, H., Camacho, D., Novais, P., Tallón-Ballesteros, A.J. (Eds.), Intelligent Data Engineering and Automated Learning – IDEAL 2018, Springer International Publishing, Cham. pp. 79–88.

[17] Fister Jr., I., Fister, I., 2021. A Brief Overview of Swarm Intelligence-Based Algorithms for Numerical Association Rule Mining. Springer Singapore, Singapore. chapter 3. pp. 47–59.

[18] Fister Jr, I., Yang, X.S., Fister, I., Brest, J., Fister, D., 2013. A brief review of nature-inspired algorithms for optimization. arXiv preprint arXiv:1307.4186 .

[19] Geng, L., Hamilton, H.J., 2006. Interestingness measures for data mining: a survey. ACM Comput. Surv. 38, 9–es.

[20] Ghosh, A., Nath, B., 2004. Multi-objective rule mining using genetic algorithms. Information Sciences 163, 123–133.

[21] Guvenir, H.A., Uysal, I., Repositor, F.A., 2000. Function approximation repository. Bilkent University. URL http://funapp. cs. bilkent. edu. tr/DataSets .

[22] Han, J., Pei, J., Yin, Y., Mao, R., 2004. Mining frequent patterns without candidate generation: A frequent-pattern tree approach. Data mining and knowledge discovery 8, 53–87.

[23] Heraguemi, K.E., Kamel, N., Drias, H., 2015. Association rule mining based on bat algorithm. Journal of Computational and Theoretical Nanoscience 12, 1195–1200.

[24] Heraguemi, K.E., Kamel, N., Drias, H., 2018. Multi-objective bat algorithm for mining numerical association rules. International Journal of Bio-Inspired Computation 11, 239–248.

[25] Kõiva, P., 2022. Implementation and performance assessment of swarm intelligence based numerical association rule mining algorithms. `https://digikogu.taltech.ee/en/Item/1bbeda96-9036-43be-8618-e83d811f3080`.

[26] Kahvazadeh, I., Abadeh, M.S., 2015. Mocanar: a multi-objective cuckoo search algorithm for numeric association rule discovery. Computer Science & Information Technology , 99–113.

[27] Kaushik, M., Sharma, R., Peious, S.A., Shahin, M., Yahia, S.B., Draheim, D., 2020. On the potential of numerical association rule mining,

in: International Conference on Future Data and Security Engineering, Springer. pp. 3–20.

[28] Kaushik, M., Sharma, R., Peious, S.A., Shahin, M., Yahia, S.B., Draheim, D., 2021. A systematic assessment of numerical association rule mining methods. SN Computer Science 2, 1–13.

[29] Kennedy, J., Eberhart, R., 1995. Particle swarm optimization, in: Proceedings of ICNN'95-international conference on neural networks, IEEE. pp. 1942–1948.

[30] Kuo, R., Gosumolo, M., Zulvia, F.E., 2019. Multi-objective particle swarm optimization algorithm using adaptive archive grid for numerical association rule mining. Neural Computing and Applications 31, 3559–3572.

[31] Ledmi, M., Moumen, H., Siam, A., Haouassi, H., Azizi, N., 2021. A discrete crow search algorithm for mining quantitative association rules. International Journal of Swarm Intelligence Research (IJSIR) 12, 101–124.

[32] Ma, Z., Wu, G., Suganthan, P.N., Song, A., Luo, Q., 2023. Performance assessment and exhaustive listing of 500+ nature-inspired metaheuristic algorithms. Swarm and Evolutionary Computation 77, 101248.

[33] Mavrovouniotis, M., Li, C., Yang, S., 2017. A survey of swarm intelligence for dynamic optimization: algorithms and applications. Swarm and Evolutionary Computation 33, 1–17.

[34] Moslehi, P., Bidgoli, B.M., Nasiri, M., Salajegheh, A., 2011. Multi-objective numeric association rules mining via ant colony optimization for continuous domains without specifying minimum support and minimum confidence. International Journal of Computer Science Issues (IJCSI) 8, 34.

[35] Mukhopadhyay, A., Maulik, U., Bandyopadhyay, S., Coello, C.A.C., 2014. A survey of multiobjective evolutionary algorithms for data mining: Part i. IEEE Transactions on Evolutionary Computation 18, 4–19.

[36] Poli, R., Kennedy, J., Blackwell, T., 2007. Particle swarm optimization. Swarm intelligence 1, 33–57.

[37] Ravber, M., Liu, S.H., Mernik, M., Črepinšek, M., 2022. Maximum number of generations as a stopping criterion considered harmful. Applied Soft Computing 128, 109478. URL: `https://www.sciencedirect.com/science/article/pii/S1568494622005804`, doi:https://doi.org/10.1016/j.asoc.2022.109478.

[38] Sharma, R., Kaushik, M., Peious, S.A., Bazin, A., Shah, S.A., Fister, I., Yahia, S.B., Draheim, D., 2022. A novel framework for unification of association rule mining, online analytical processing and statistical reasoning. IEEE Access 10, 12792–12813.

[39] Sharma, R., Kaushik, M., Peious, S.A., Yahia, S.B., Draheim, D., 2020. Expected vs. unexpected: selecting right measures of interestingness, in: Big Data Analytics and Knowledge Discovery, Springer International Publishing, Cham. pp. 38–47.

[40] Socha, K., 2008. Ant colony optimisation for continuous and mixed-variable domains. Ph.D. thesis. Univ. Libre de Bruxelles, IRIDIA, Brussels, Belgium,.

[41] Srikant, R., Agrawal, R., 1996. Mining quantitative association rules in large relational tables, in: Proceedings of the 1996 ACM SIGMOD international conference on Management of data, pp. 1–12.

[42] Stupan, Ž., Fister, I., 2022. Niaarm: A minimalistic framework for numerical association rule mining. Journal of Open Source Software 7, 4448.

[43] Tang, R., Fong, S., Yang, X.S., Deb, S., 2012. Wolf search algorithm with ephemeral memory, in: Seventh International Conference on Digital Information Management (ICDIM 2012), IEEE. pp. 165–172.

[44] Varol Altay, E., Alatas, B., 2020. Performance analysis of multi-objective artificial intelligence optimization algorithms in numerical association rule mining. Journal of Ambient Intelligence and Humanized Computing 11, 3449–3469.

[45] Yang, X.S., 2010. A new metaheuristic bat-inspired algorithm, in: Nature inspired cooperative strategies for optimization (NICSO 2010). Springer, Berlin, Heidelberg, pp. 65–74.

[46] Yang, X.S., Deb, S., 2009. Cuckoo search via lévy flights, in: 2009 World congress on nature & biologically inspired computing (NaBIC), IEEE. pp. 210–214.

Appendix 8

[VIII]

M. Kaushik, R. Sharma, and D. Draheim. Discretizing numerical attributes: An analysis of human perceptions (extended version). arxiv:2311.03278, 2023

Discretizing Numerical Attributes: An Analysis of Human Perceptions

Minakshi Kaushik[0000−0002−6658−1712], Rahul Sharma[0000−0002−9024−8768], and
Dirk Draheim[0000−0003−3376−7489]

Information Systems Group
Tallinn University of Technology
Akadeemia tee 15a, 12618 Tallinn, Estonia
{minakshi.kaushik,rahul.sharma,dirk.draheim}@taltech.ee

Abstract. Machine learning (ML) has employed various discretization methods to partition numerical attributes into intervals. However, an effective discretization technique remains elusive in many ML applications, such as association rule mining. Moreover, the existing discretization techniques do not reflect best the impact of the independent numerical factor on the dependent numerical target factor. This research aims to establish a benchmark approach for numerical attribute partitioning. We conduct an extensive analysis of human perceptions of partitioning a numerical attribute and compare these perceptions with the results obtained from our two proposed measures. We also examine the perceptions of experts in data science, statistics, and engineering by employing numerical data visualization techniques. The analysis of collected responses reveals that 68.7% of human responses approximately closely align with the values generated by our proposed measures. Based on these findings, our proposed measures may be used as one of the methods for discretizing the numerical attributes.

Keywords: Machine learning · data mining · discretization · numerical attributes· partitioning

1 Introduction

Various types of variables are available in real-world data. However, discrete values have explicit roles in statistics, machine learning (ML), and data mining. Presently, there is no benchmark approach to find the optimum partitions for discretizing complex real-world datasets. Generally, if a factor impacts another factor, in that case, humans can easily perceive the compartments or partitions because the human brain can easily perceive the differences between the factors and detect the partitions. However, it is not easy for a human or even an expert to find the appropriate compartments in complex real-world datasets. In state-of-the-art, to find the optimum partitions of the numerical values, various discretization techniques have also been presented in the literature [23,13,22]. However, the existing discretization techniques do not reflect best the impact of the independent numerical factor on the dependent numerical target factor. Moreover, no existing discretization approach uses numerical attributes as influencing and response factors.

To find the cut-points for the cases of two-partitioning and three-partitioning, we have proposed two measures *Least Squared Ordinate-Directed Impact Measure* (LSQM) and *Least Absolute-Difference Ordinate-Directed Impact Measure* (LADM) [18]. These measures provide a simple way to find partitions of numerical attributes that reflect best the impact of one independent numerical attribute on a dependent numerical attribute. In these measures, we use numerical attributes as influencing and response factors to distinguish them from the existing approaches.

In this paper, the outcome of *LSQM* and *LADM* measures are compared with the human-perceived cut-points to assess the accuracy of the measures. We use numerical attributes as influencing and response factors to distinguish them from the existing approaches. A series of graphs with different data points are used to collect the human responses. Here, data scientists, ML experts and other non-expert persons are referred to as humans.

The idea of this research emerged from the research on partial conditionalization [8,9], association rule mining (ARM) [31,29] and numerical association rule mining (NARM) [32,19,20]. These papers discuss the discretization process as an essential step for NARM. Moreover, research on discretizing the numerical attributes is an essential step in frequent itemset mining, especially for quantitative association rule mining (QARM) [32].

In the same sequence, we have also presented a tool named Grand report [27] and a framework [30] for unifying ARM, statistical reasoning, and online analytical processing. These paper strengthens the generalization of ARM by finding the partitions of numerical attributes that reflect best the impact of one independent numerical attribute on a dependent numerical attribute. Our vision is to develop an ecosystem to generalize the ML approaches by significantly improving the ARM from different dimensions.

The paper is organized as follows. In Sect. 2, we delve into the discussion of related work concerning discretization and its connection with human perception. This section aims to provide a comprehensive overview of prior research and studies that have explored the topic from different angles. In Sect. 3, we explain the motivation for conducting this study. Sect. 4 describes the *LSQM* and *LADM* measures. Then, we describe the design of the experiment in Sect. 5. In Sect. 6, we present the analysis and results. The conclusion and future work are given in Sect. 7.

2 Related Work

Based on human perception evaluation and different discretization techniques, we discuss the related work in the direction of discretization and clustering techniques and human perception.

2.1 Discretization

Many data mining and ML algorithms are not designed to work with numeric attributes and instead require nominal attributes as input data [13]. In order to convert numeric attributes into nominal attributes, different discretization methods have been employed as a pre-processing measure. Discretization techniques divide the range of a numeric attribute into n intervals, which are determined by $n - 1$ cut-points. A variety of discretization methods are available in the literature [13],[22] and [23]. Dougherty et al. [7] compared and analyzed discretization strategies along three dimensions: global versus local, supervised versus unsupervised, and static versus dynamic. Static approaches discretize each attribute separately, whereas dynamic methods conduct a search through space to find interdependencies between features. Liu et al. [23] performed a systematic study of existing discretization methods and proposed a hierarchical framework for discretization methods from the perspective of splitting and merging. The unsupervised static discretization method, such as equal-width, uses the minimum and maximum values of the continuous attribute and then divides the range into equal-width intervals called bins. In contrast, the equal-frequency algorithm determines an equal number of continuous values and places them in each bin [4].

The RUDE (Relative Unsupervised Discretization) algorithm [25] performs the discretization of numerical attributes in three steps: pre-discretizing, structure projection, and merging split points. In 2010, Joita [15] proposed an unsupervised method for selecting the initial cluster centers for the clustering of a one-dimensional vector of real-valued data. This method was based on a k-means clustering algorithm and can be used in single-attribute discretization. In 2013, Dietrich et al. [6] also proposed a method for obtaining cut-points that is more intuitive for human users. This smoothed discretization approach works as a post-processing step to obtain the intervals after using an arbitrary traditional discretization approach. The authors also proposed two measures, *distance-based deviation measure* and *instance-based deviation measure*, for comparing the original discretization method cut-points with modified cut-points. The discretization cut-points were computed for each training dataset for the three general discretization methods: equal-frequency [4] discretization, entropy-based discretization [12], [28] and Chi2 [24] discretization. After that, the introduced smoothing approaches were used with distance-based and instance-based modification measures to get the desired results. There is another similar work, called the best piecewise constant approximation [21], which deals with approximating a single variable function. Still, it is different because we are not using signals, and our primary focus is on data sets that use several data points for one value of the influencing factor. Eubank used the population quantile function as a tool to show the best piecewise constant approximation problem [11]. Later Bergerhoff [3] suggested a method for finding optimal piecewise constant approximations of one-dimensional signals using particle swarm optimization.

2.2 Human perceptual evaluation

Discretization approaches are usually evaluated based on their mathematical backgrounds. However, we are the first to assess the discretization measures by considering human perceptions.

In the state of the art, many studies have used human perception to evaluate various techniques. However, they are not completely related to discretization. Tatu et al. [33] proposed a preliminary investigation of human perception using visual quality criteria for multidimensional data. The authors conducted a user study to examine the relationship between human cluster interpretation and the measurements automatically retrieved from 2D scatter plots. Etemadpour et al. [10] conducted a perception-based evaluation of high-dimensional data where humans were asked to identify clusters and analyze distances inside and across clusters. Demiralp et al. [5] used human judgments to estimate perceptual kernels for visual encoding variables such as shape, size, color, and combinations. The experiment used Amazon's Mechanical Turk platform, with twenty Turkers completing thirty MTurk jobs. In [1], a new visual quality measure (VQM) based on perceptual data was proposed to rank monochrome scatter plots. This experiment collected perceptual data from human subjects, and the best clustering model was chosen to create a perceptual-based VQM of grouping patterns. Similarly, a study by Aupetit [2] analyzed and compared clustering algorithms through the lens of human perception in 2D scatter plots. The primary focus of the authors was to evaluate how accurately clustering algorithms aligned with the way humans perceive clusters. The authors evaluated Gaussian Mixture Models, CLIQUE, DBSCAN, Agglomerative Clustering methods, and 1437 variations of k-means on the benchmark data. Our work is also related to considering human perceptions for evaluating our proposed *LSQM* and *LADM* measures for discretizing numerical attributes.

3 Motivation

Real-world data sets contain real or numerical values frequently. However, many data mining and ML approaches need discrete values. For years, obtaining discrete values from numerical values has been a complex and ongoing task. The main issue with the discretization process is obtaining the perfect intervals with specific ranges and numbers of intervals. In state of the art, several discretization approaches such as equi-depth, equi-width [4], MDLP [12], Chi2 [24], D2 [4], etc. have been proposed. However, determining the most effective discretizer for each situation is still a challenging problem. The existing methods for discretizing numerical attributes are not automated and require expert knowledge; therefore, there is a need to develop an automated and formal measure for finding the optimal partition of numerical attributes.

In [18], we presented an order-preserving partitioning method to find the partitions of numerical attributes that reflect best the impact of one independent numerical attribute on a dependent numerical attribute. In extreme cases (such as step-functions), humans can easily visualize the perfect partitions and even the number of compartments. However, in distinct cases, the ideal partition range depends on the perception of data experts. In state of the art, no investigation is available to understand the human perception of partitioning. Moreover, the current literature provides a comparison of discretization methods and compares their results. In this paper, we take a different approach to compare the human perception of discretization with the outcome of the proposed discretization method. We aim to visualize the differences between the outcomes of the proposed methods and the human perception of discretization.

4 Impact Driven discretization Method

In the *Impact driven discretization method* [18], we perform discretization on the independent numerical attribute using order-preserving partitioning to understand its impact on the numerical target attribute. The method involves creating a total of $(k-1)$ cut-points, with k being the number of partitions recommended by the user. Below are two measures introduced in the paper [18].

4.1 The LSQM Measure

The *LSQM* measure operates by initially computing the squared difference between the y-value of each data point and the average of y-values within the current partition. This measure maintains the order of the independent variable by considering the values of data points, ensuring that the values within one partition are consistently lower than those in the subsequent partition. After summing up the squared differences of the several partitions, *LSQM* retrieves the minimum values as cut-points.

Definition 1 (Least Squared Ordinate-Directed Impact Measure). Given $n \geq 2$ real-valued data points $(< x_i, y_i >)_{1 \leq i \leq n}$, we define the *least squared ordinate-directed impact measure* for k-partitions (with $k-1$ *cut-points*) as follows:

$$\underset{i_0=0<i'_1<\ldots<i'_{k-1}<i'_k=n}{min} \sum_{j\,=\,1}^{k} \sum_{i'_{j-1}<i"\leq i'_j} (y_{i"} - \mu_{i'_{j-1}<\phi\leq i'_j})^2 \qquad (1)$$

where the *average of data values in a partition* $\mu_{a<\phi\leq b}$ between indexes a and b $(a < b \leq n)$ is defined as

$$\mu_{a<\phi\leq b} = \frac{\sum\limits_{a<\phi\leq b} y_\phi}{b-a} \qquad (2)$$

In (1), we have that i'_j is the highest element in the *j-th* partition, where *highest element* means the data point with the highest index.

Indeed, the *LSQM* (Least Squares Ordinate-Directed Impact Measure) measure may appear similar to the k-means clustering algorithm on the surface, as both involve partitioning data into clusters. However, they differ significantly in their underlying principles and applications.

k-means clustering is primarily an unsupervised ML technique employed for the task of clustering data points into groups or clusters, with each data point assigned to the cluster whose centroid is closest to it in terms of a chosen distance metric, often the Euclidean distance. Euclidean distance metric calculates dissimilarity between data points, which involves measuring the geometric distance between vectors X and Y. The primary goal of k-means is to minimize the sum of squared distances between data points and their assigned cluster centroids, and it finds applications in various domains, including customer segmentation, image compression, and data reduction. However, k-means' effectiveness is influenced by the initial random selection of cluster centers, which can lead to different clustering results depending on the initialization.

In contrast, *LSQM* is a specialized method designed specifically for discretizing numerical attributes. Its core objective is to partition a numerical attribute into intervals while preserving the order of data points within those intervals. *LSQM* achieves this by measuring the squared difference between the values of data points and the average of values within each partition, aiming to minimize the sum of squared differences. Unlike k-means, *LSQM* is not highly dependent on the initial point chosen to start the partitioning process, making it robust in this regard. *LSQM* is primarily employed in data preprocessing tasks related to data mining, enhancing the quality of numerical attribute discretization.

In summary, k-means clustering is a versatile and widely used clustering algorithm with applications across various domains, focusing on minimizing the squared distances between data points and cluster centroids. On the other hand, *LSQM* serves a specific purpose in discretizing numerical attributes while maintaining the order of data points, making it particularly valuable in data preprocessing for data mining tasks.

4.2 The LADM Measure

For the *LADM* measure, we take the sum of the absolute differences of the several partitions.

Definition 2 (Least Absolute-Difference Ordinate-Directed Impact Measure).
Given $n \geq 2$ real-valued data points $(< x_i, y_i >)_{1 \leq i \leq n}$, we define the *least absolute-difference ordinate-directed impact measure* for k-partitions (with $k-1$ *cut-points*) as follows:

$$\underset{i_0=0<i'_1<\ldots<i'_{k-1}<i'_k=n}{min} \sum_{j\,=\,1}^{k} \sum_{i'_{j-1}<i"\leq i'_j} |y_{i"} - \mu_{i'_{j-1}<\phi\leq i'_j}| \tag{3}$$

where the *average of data values in a partition* $\mu_{a<\phi\leq b}$ between indexes a and b $(a < b \leq n)$ is defined as

$$\mu_{a<\phi\leq b} = \frac{\sum\limits_{a<\phi\leq b} y_\phi}{b-a} \tag{4}$$

5 Experimental Design

To understand how humans partition numerical factors, we designed a series of graphs and asked several experts to partition the data given in the graphs. Initially, to produce a diverse collection of graphs with different data points, a set of graphs was shared and discussed with our own research team. The team consists of three early-stage researchers and one senior researcher. These graphs include step functions, linear functions, and mixed data graphs. Finally, twelve graphs were selected to be shared with humans (see Figs. 1, 2 and 3). These graphs are obtained from nine synthetic datasets (D1 to D9) and three real-world datasets (D10 to D12). These synthetic datasets (D1 to D9) have only two numerical attributes. The dataset D10 is a real-world dataset. The data set, DC public government employees [16], contains 33,424 records of DC public government employees and their salaries in 2011. This dataset is sourced from the Washington Times via Freedom of Information Act (FOIA) requests. The dataset D11 is Heart disease dataset [14], and it is sourced from the UCI machine learning repository. This dataset has 13 attributes and 303 records. We used attribute {Age} and {Cholesterol} for drawing the graph. The dataset D12 is a New Jersey (NJ) school teacher salaries (2016) [26] sourced from the (NJ) Department of Education. It contains 138715 records and 15 attributes. We have taken only an initial 23000 rows from the dataset. We are interested in the column {experience_total} and {salary}. A copy of all these datasets is available in the GitHub repository [17].

We designed a Google form by providing a series of graphs containing different types of numerical data points and relevant questions to collect human responses and their perceptions about discretization. We put some constraints in the Google form to know whether a response is submitted by DS/ML experts or not. By employing this procedure, we compare and comprehend the perceptions of both DS/ML experts and non-expert responders.

The Google form was sent to fifty DS/ML experts and non-experts to estimate the number of partitions and the ranges of these partitions to obtain the cut-points. The following data was gathered and compiled from the experiments: respondent identification (name), their email addresses, domain expertise (DS/ML expert or non-expert), number of partitions identified, and ranges of each partition.

6 Analysis and Result

Out of the fifty responses received via the Google form, two were incomplete; therefore, we did not consider them for the analysis. From the rest of the forty-

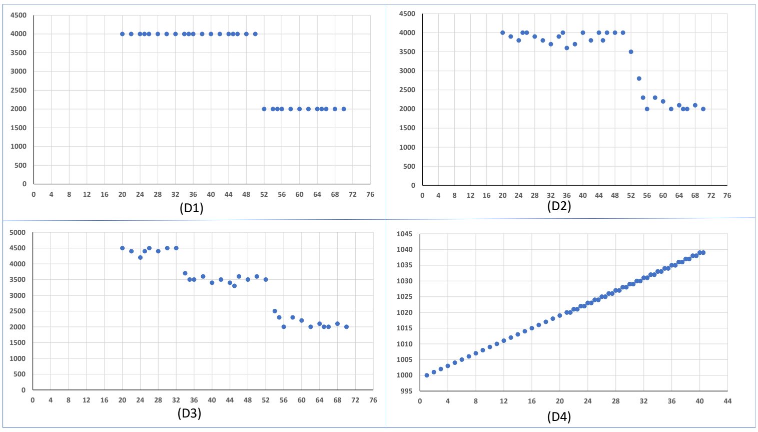


Fig. 1. Graphs for datasets D1 to D4.

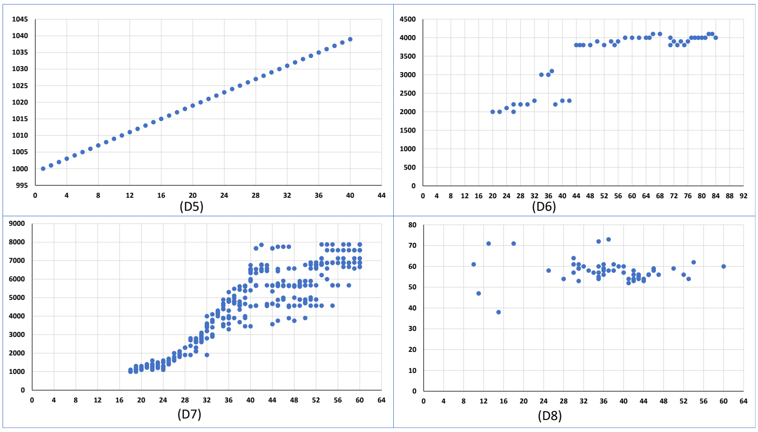


Fig. 2. Graphs for datasets D5 to D8.

Table 1. The comparison of human perception to identify number of partitions based on their profile.

Datasets	Number of Partitions	Total responses from DS/ML experts = 60%	Total Responses from Non-expert People = 40%
		% Responses	% Responses
D1	2	93.3%	90%
	3	6.67%	10%
D2	2	73.3%	60%
	3	26.6%	40%
D3	2	0%	0%
	3	93.3%	100%
	4	6.66%	0%
D4	0	33.3%	20%
	2	53%	70%
	3	13.3%	10%
D5	0	93.3%	90%
	2	0%	0%
	3	6.6%	10%
D6	2	13.3%	30%
	3	26.6%	0%
	4	26.6%	40%
	5	33.3%	30%
D7	2	60%	40%
	3	20%	40%
	4	20%	20%
D8	2	33.3%	60%
	3	66.6%	30%
	4	0%	10%
D9	2	73.3%	60%
	3	26.6%	30%
	4	0%	10%
D10	0	40%	40%
	2	40%	30%
	3	6.66%	20%
	4	6.66%	0%
	5	6.66%	10%
D11	0	53.3%	60%
	2	26.6%	30%
	3	20%	0%
	4	0%	10%
D12	0	40%	20%
	2	26.6%	30%
	3	6.66%	30%
	4	20%	10%
	5	6.66%	10%

Table 2. The comparison of human perceived cut-points with the LSQM and LADM measures.

D	P	Human Perception		LSQM	LADM
		R	Approx. near Cut-Points	Cut-points	Cut-Points
D1	2	92%	50(91.3%), 48(8.6%)	50	50
	3	8%	(48,60)(50%), (20,50)(50%)	(20, 50)	(20, 50)
D2	2	68%	50(88.2%), 52(11.7%)	52	52
	3	32%	(50,54)(37.5%), (20,53)(25%)	(52, 54)	(52, 54)
D3	3	96%	(32,52)(62%), (30,52)(16.6%)	(32,52)	(32,52)
	4	4%	(20,32,52)(100%)	(32,52,55)	(32,52,60)
D4	0	28%	NA	NA	NA
	2	60%	20(86.6%), 25(13.3%)	20	20
	3	12%	(20,45)(66.6%), (20,30)(33.3%)	(12, 24)	(12, 25)
D5	0	92%	NA	NA	NA
	2	0%	0%	20	19
	3	8%	(14,28)(100%)	(13, 26)	(13, 26)
D6	2	20%	32(40%), 42(40%) 50(20%)	42	42
	3	16%	(42,68)(50%), (32,42)(25%)	(32, 42)	(32, 42)
	4	32%	(32,37,42)(87.5%), (33,37,43)(12.5%)	(32, 37, 42)	(32, 37, 42)
	5	32%	(32,42,37,68)(87.5%), (17,32,38,42)(12.5%)	(32, 37, 42, 56)	(32, 37, 42, 56)
D7	2	52%	40(84.6%), 50(7.6%), 36(7.6%)	35	33
	3	28%	(32,39)(57.1%)	(32, 39)	(32, 39)
	4	20%	(32,39,50)(60%), (41,47,53)(40%)	(32,39,52)	(32,39,52)
D8	2	44%	18(36%), 30(27%)	40	40
	3	52%	(28,47)(53.8%), (18,47)(23%)	(13, 15)	(40,45)
	4	4%	(18,47,54)(100%)	(11, 13, 15)	(13,15,18)
D9	2	68%	40(41%), 50(23.5%),47(23.5%)	15	13
	3	28%	(24,36)(57%), (36,47)(28.5%)	(14, 50)	(8,15)
	4	4%	(24,39,47)(100%)	(14,50,52)	(13,15,18)
D10	0	40%	NA	NA	NA
	2	36%	44(33.3%), 24(33.3%), 52(22%)	56	11
	3	12%	(20,32)(66.6%), (18,45)(33.3%)	(11,56)	(11,56)
	4	4%	(12,29,42)(100%)	(49,50,56)	(11,52,56)
	5	8%	(12,24,30,40)(100%)	(11,49,50,56)	(11,41,50,56)
D11	0	56%	NA	NA	NA
	2	28%	52(42.8%), 60(42.8%), 67(14%)	67	67
	3	12%	(48,68)(66.6%), (40,68)(33.3%)	(67,70)	(67,70)
	4	4%	(40,48,68)(100%)	(51, 63, 67)	(62,67,70)
D12	0	32%	NA	NA	NA
	2	28%	50(42.8%), 40(28.5%), 24(28.5%)	17	15
	3	16%	(22,32)(50%), (14,34)(25%), (27,44)(25%)	(15,51)	(14,38)
	4	16%	(9,31,58)(50%), (20,36,48)(25%), (10,20,30)(25%)	(15,51,52)	(10,17,38)
	5	8%	(16,28,36,44)(50%), (7,20,28,36)(50%)	(15,50,51,52)	(14,37,51,52)

D: Datasets; P: number of partitions; R: percentage of responses

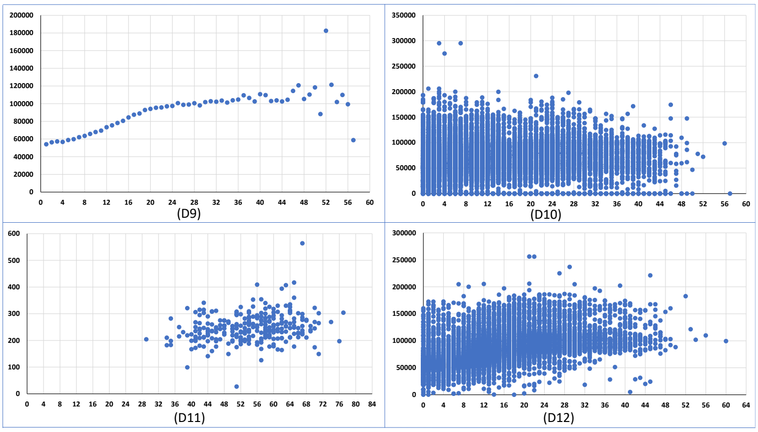


Fig. 3. Graphs for datasets D9 to D12.

eight responses, we divided the responses into two categories: expert responses and non-expert responses.

Table 1 illustrates the comparison of human perception to identify the number of partitions between the DS/ML experts' responses and non-expert people. We received 60% responses from DS/ML experts and 40% of answers from non-expert people. We analyzed that responses from both categories were opposite for graph D8. Out of the total responses for D8, 33.3% responses of DS/ML experts marked two partitions and 66.6% responses of experts marked three partitions; however, 60% of non-experts marked two partitions, and only 30% marked three partitions.

In graphs D3 and D5, we analyzed that no contributor (experts or non-experts) marked two partitions. No non-expert contributors marked three partitions for graph D6 and four partitions for graph D3, whereas 26.6% of DS/ML experts identified three partitions for D6, and 6.66% experts marked four partitions in graph D3. For graph D10, 40% DS/ML experts and 40% non-experts have marked no partition.

Table 2 illustrates the comparison between the results of human perception, the *LSQM* and the *LADM* measure. In the initial four columns, we detailed the dataset used, the count of partitions, the response percentage, and the approximate cut-points as observed by contributors. The last two columns present the cut-points assessed by the measures. Table 3 describes the similarity percentage between cut-points provided by human perceived experiment outcome and the *LSQM* and the *LADM* measures outputs. We have mentioned the cut-points from responses near the *LSQM* and the *LADM* provided cut-points. We determine the matching status by distributing the matching percentage into the

Table 3. Similarity between human perceived, LSQM and LADM cut-points.

D	P	LSQM Cut-Points	LADM Cut-Points	Human Perceived Cut-Points (Near to LSQM Cut-Points)	Matching%	Matching status
D1	2	50	50	50 (91.3%)	91.3%	Very High
	3	(20,50)	(20,50)	(20,50)(50%)	50%	Medium
D2	2	52	52	52(11.7%)	11.7%	Low
	3	(52,54)	(52,54)	(50,54)(37.5%)	19%	Low
D3	3	(32,52)	(32,52)	(32,52)(62%)	62%	High
	4	(32,52,55)	(32,52,60)	(20,32,52)(100%)	59%	Medium
D4	2	20	20	20(80.6%)	80.6%	Very High
	3	(12,24)	(12,25)	(20,30)(33.3%)	0%	No match
D5	3	(13,26)	(13,26)	(13,26)(100%)	100%	Very High
D6	2	42	42	42(40%)	40%	Medium
	3	(32,42)	(32,42)	(32,42)(25%)	25%	Low
	4	(32,37,42)	(32,37,42)	(32,37,42)(85.7%)	85.7%	Very High
	5	(32,37,42,56)	(32,37,42,56)	(32,37,42,68)(85.7%)	75%	High
D7	2	35	33	36(7.6%)	0%	No match
	3	(32,39)	(32,39)	(32,39)(57%)	57%	High
	4	(32,39,52)	(32,39,52)	(32,39,52)(60%)	60%	High
D8	2	40	40	30(27%)	0%	No match
	3	(13,15)	(40,45)	(18,47)(23%)	0%	No match
	4	(11,13,15)	(13,15,18)	(18,47,54)(100%)	0%	No match
D9	2	15	13	40(41%)	0%	No match
	3	(14,50)	(8,15)	(24,36)(57%)	0%	No match
	4	(14,50,52)	(10,17,33)	(24,39,47)(100%)	0%	No match
D10	2	56	11	52(22%)	0%	No match
	3	(11,56)	(11,56)	(18,45)(33.3%)	0%	No match
	4	(49,50,56)	(11,52,56)	(12,29,42)(100%)	0%	No match
	5	(11,49,50,56)	(11,41,50,56)	(12,24,30,40)(100%)	0%	No match
D11	2	67	67	50(42.8%)	0%	No match
	3	(67,70)	(67,70)	(48,68)(66.6%)	0%	No match
	4	(51,63,67)	(62,67,70)	(40,48,68)(100%)	0%	No match
D12	2	17	15	24(28.5%)	0%	No match
	3	(15,51)	(14,38)	(14,34)(25%)	0%	No match
	4	(15,51,52)	(10,17,38)	(20,36,48)(25%)	0%	No match
	5	(15,50,51,52)	(14,37,51,52)	(16,28,36,44)(50%)	0%	No match

D: Datasets; P: number of partitions; R: percentage of responses
Very High: 80-100%, High:60-80%, Medium:40-60%, Low:1-40%, No match: 0%

following categories: VH (Very High), H (High), M (Medium), L (Low) and NM (No match). The distribution of ranges is mentioned at the bottom of Table 3. It is clear from Table 3 that human perceived cut-points and the cut-points identi-

Table 4. Analysis of unmatched datasets in regard of number of partitions (#) for LSQM, LADM and human perceived cut-points.

D	P	LSQM Method		LADM Method		Human Perception		Remarks
		LSQM cut-points	LC	LADM cut-points	LC	Human Perceived cut-points	LC	
D8	2	40	Yes	40	Yes	30	Yes	Matter of perception.
	3	(13,15)	No	(40,45)	No	(18,47)	Yes	LSQM, LADM to be improved.
	4	(11,13,15)	No	(13,15,18)	No	(18,47,54)	Yes	LSQM, LADM to be improved.
D9	2	15	No	13	No	40	Yes	LSQM, LADM need to be improved.
	3	(14,50)	Yes	(8,15)	No	(24,36)	Yes	Matter of perception. However, LADM needs to be improved.
	4	(14,50,52)	No	(10,17,33)	No	(24,39,47)	Yes	LSQM and LADM to be improved.
D10	2	56	No	11	No	52	No	This dataset is an exceptional case; random cutpoints are obtained.
	3	(11,56)	No	(11,56)	No	(18,45)	Yes	
	4	(49,59,56)	No	(11,52,56)	No	(12,29,42)	Yes	
	5	(11,49,50,56)	No	(11,41,50,56)	No	(12,24,30,40)	Yes	
D11	2	67	Yes	67	Yes	50	Yes	Matter of perception.
	3	(67,70)	No	(67,70)	No	(48,68)	Yes	LSQM to be improved.
	4	(51,63,67)	Yes	(62,67,70)	No	(40,48,68)	Yes	Matter of perception. However, LADM needs to be improved.
D12	2	17	Yes	15	Yes	24	Yes	This dataset is an exceptional case; random cutpoints are obtained.
	3	(15,51)	Yes	(14,38)	Yes	(14,34)	Yes	
	4	(15,51,52)	No	(10,17,38)	No	(20,36,48)	Yes	
	5	(15,50,51,52)	No	(14,37,51,52)	No	(16,28,36,44)	Yes	
D4	3	(12,24)	Yes	(12,25)	Yes	(20,30)	Yes	Matter of perception.
D7	2	35	Yes	33	Yes	36	Yes	Matter of perception.

D: Datasets; P: number of partitions; LC: Logical Correctness

fied by the proposed measures $LSQM$ and $LADM$ do not match for the datasets D8 to D12. In Table 4, we present an analysis and reason for not getting similar cut-points for the datasets D8 to D12. If we look at Fig. 1(D8), then it seems logical to have cut-points at the data points of 40 (LSQM, LADM cut-point) and 30 (Human perceived cut-point) for two partitions on the X-axis. Humans divided the scattered points into the first partition and dense data points into the second partition. In contrast, both measures calculated the cut-point in the middle of the dense data points. This case can be observed as a matter of perception for human perceived cut-points, while the cut-points marked by both the measures seem analytically correct. For the cases of three partitions and four partitions, human perceived cut-points $(18, 47)$ and $(18, 47, 54)$ are good, but the cut-points provided by the $LSQM$measure and the $LADM$ measure are not satisfactory.

Similarly, in dataset D9, human perception identified a single cut-point at 40 for two partitions and two cut-points at $(24, 36)$ for three partitions, which intuitively makes sense. However,$LSQM$ and $LADM$ produced cut-points at 15 and 13 for two partitions, which are not analytically accurate. On the other hand,$LSQM$ cut-points $(14, 50)$ align analytically, albeit it remains a matter of perception. Even though the three partitions provided by$LADM$ $(8, 15)$ and the four partitions by $LSQM$ $(14, 50, 52)$ and $LADM$ $(10, 17, 33)$ might seem illogical, human-perceived cut-points $(24, 39, 47)$ appear appropriate.

In dataset D11, the situation is again contingent on perception, with discrepancies arising for both two partitions and four partitions in the case of $LSQM$. For four partitions, $LADM$ suggests cut-points $(62, 67, 70)$, which lack logical consistency. Meanwhile, for three partitions, both $LSQM$ and $LADM$ present unexpected cut-points $(67, 70)$.

Datasets D4 and D7 also lack matching results for three partitions and two partitions, respectively. In the case of D4, both $LSQM$ and $LADM$ suggest cut-points $(12, 24)$ and $(12, 25)$, while human perception identifies $(20, 30)$ as the appropriate cut-points. This instance can be attributed to varying perceptions.

Similarly, for D7, the proposed cut-points by $LSQM$ and $LADM$ for two partitions are 35 and 33, respectively, which do not exactly align with the human-perceived cut-point of 36. However, given the scattered distribution of data points on the graph, the difference between the proposed measures' cut-points and the human-perceived cut-point is negligible. In this case, both sets of cut-points can be considered suitable, further emphasizing the role of perception. While these cut-points do not match precisely, it does not affect the correctness of the measures due to the lack of similarity.

For datasets D10 and D12, the responses from contributors present a unique challenge. In the case of D10, 40% of contributors indicated no partition, while the remaining contributors marked random cut-points for two, three, four, and five partitions. Similarly, for D12, 32% of contributors opted for no partition, while 68% of contributors designated random cut-points for various partitions. These random cut-points identified by humans are not easily aligned with the cut-points derived from the proposed $LSQM$ and $LADM$ measures. Furthermore, these random cut-points lack analytical correctness.

As a result, for datasets with such characteristics, it becomes difficult for humans to identify the most appropriate partitions. The absence of clear patterns or logic in the random cut-points makes it challenging to establish meaningful partitions, emphasizing the complexity of the task in these scenarios.

The distribution of response percentages for each partition across the datasets is visually represented in Fig. 4. Notably, datasets D5, D10, D11, and D12 ex-

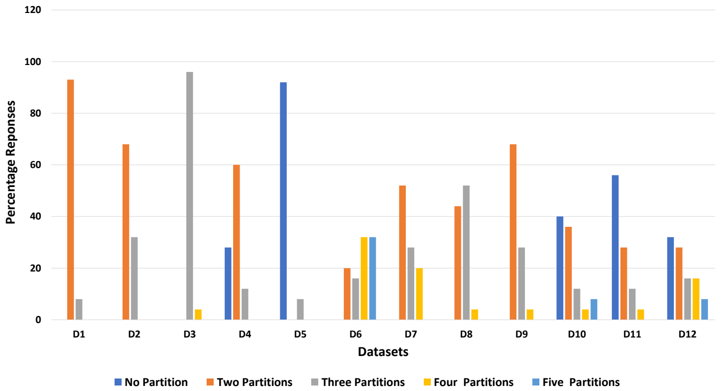


Fig. 4. Percentage Responses of partitions for each dataset.

hibit a significant prevalence of responses indicating no partition compared to other partition numbers. Hence, it proves that for these specific datasets, human participants struggled to form a definitive perception. It suggests that individuals had difficulty forming a clear and consistent perception of these datasets, making it unclear for them to identify appropriate cut-points.

It is worth highlighting that datasets D5 and D6 share a similar visual appearance. However, they received different responses in terms of cut-points. This discrepancy can be attributed to the distinct distribution of data points within each dataset. Notably, D6 did not receive any responses suggesting no partition, while D5 lacked responses suggesting two partitions.

Conversely, for datasets D1, D2, D4, D7, and D9, the majority of responses predominantly indicated the presence of two partitions. This indicates a higher degree of consensus among contributors regarding the presence of two partitions in these datasets.

Table 3 provides an overview of the alignment between human-perceived cut-points and those observed by the *LSQM* and *LADM* measures. The analysis reveals that 25% of responses exhibited a *Very High* level of similarity, 25% demonstrated a *High* level of similarity, 18.7% displayed a *Medium* level of similarity, and an additional 18.7% showed a *Low* level of similarity. When considering the collective matching statuses, it becomes evident that approximately 68.7% of the responses closely resembled the cut-points identified by the proposed *LSQM* and *LADM* measures. This analysis primarily pertains to the initial datasets (D1 to D7), as random cut-points were observed in the responses for datasets D8 to D12. These random cut-points in the latter datasets presented challenges in aligning them with the analytically calculated cut-points generated by the proposed measures. Further details and explanations for the dissimilarity in cut-points for datasets D8 to D12 can be found in Table 4.

7 Conclusion

This paper is the first step toward understanding the human perception regarding partitioning numerical attributes. We meticulously examine the partitions perceived by humans and compare them with the outputs generated by both proposed measures. Our approach involved evaluating human perception by presenting a series of graphs containing numerical data and subsequently comparing human-perceived cut-points for partitioning with the results generated by the *LSQM* and *LADM* measures.

The outcomes of this study indicate a close alignment between the cut-points produced by the proposed measures and those perceived by humans. Particularly for the initial datasets (D1 to D7), our proposed measures yielded results that closely approximated human perception. However, certain exceptional cases, such as datasets D10 and D12, highlighted situations where humans faced challenges identifying optimal partitions. The results also demonstrate that both measures yield approximately similar outcomes. These findings represent a promising step forward, signifying progress in the pursuit of advancing ARM by identifying numerical attribute partitions that reflect best the impact of an independent numerical attribute on a dependent numerical attribute. In future research endeavors, we intend to explore inter-measures for comparing partitions with varying numbers of k-partitions.

Acknowledgements

This work has been conducted in the project “ICT programme” which was supported by the European Union through the European Social Fund.

References

1. Abbas, M.M., Aupetit, M., Sedlmair, M., Bensmail, H.: Clustme: A visual quality measure for ranking monochrome scatterplots based on cluster patterns. In: Computer Graphics Forum. vol. 38, pp. 225–236. Wiley Online Library (2019)
2. Aupetit, M., Sedlmair, M., Abbas, M.M., Baggag, A., Bensmail, H.: Toward perception-based evaluation of clustering techniques for visual analytics. In: 2019 IEEE Visualization Conference (VIS). pp. 141–145 (2019). https://doi.org/10.1109/VISUAL.2019.8933620
3. Bergerhoff, L., Weickert, J., Dar, Y.: Algorithms for piecewise constant signal approximations. In: 27th European Signal Processing Conference (EUSIPCO). pp. 1–5. IEEE (2019)
4. Catlett, J.: On changing continuous attributes into ordered discrete attributes. In: European working session on learning. pp. 164–178. Springer (1991)

5. Demiralp, Ç., Bernstein, M.S., Heer, J.: Learning perceptual kernels for visualization design. IEEE transactions on visualization and computer graphics **20**(12), 1933–1942 (2014)
6. Dietrich, G., Lemmerich, F., Puppe, F.: Smoothed discretization for simplified cutpoints. In: LWA. pp. 103–106 (2013)
7. Dougherty, J., Kohavi, R., Sahami, M.: Supervised and unsupervised discretization of continuous features. In: Machine learning proceedings 1995, pp. 194–202. Elsevier (1995)
8. Draheim, D.: Generalized Jeffrey Conditionalization: A Frequentist Semantics of Partial Conditionalization. Springer (2017)
9. Draheim, D.: Future perspectives of association rule mining based on partial conditionalization. In: Hartmann, S., Küng, J., Chakravarthy, S., Anderst-Kotsis, G., A Min Tjoa, Khalil, I. (eds.) Proceedings of DEXA'2019 - the 30th International Conference on Database and Expert Systems Applications. LNCS, vol. 11706, p. xvi. Springer, Heidelberg New York Berlin (2019)
10. Etemadpour, R., da Motta, R.C., de Souza Paiva, J.G., Minghim, R., de Oliveira, M.C.F., Linsen, L.: Role of human perception in cluster-based visual analysis of multidimensional data projections. In: 2014 International Conference on Information Visualization Theory and Applications (IVAPP). pp. 276–283 (2014)
11. Eubank, R.: Optimal grouping, spacing, stratification, and piecewise constant approximation. Siam Review **30**(3), 404–420 (1988)
12. Fayyad, U., Irani, K.B.: Multi-interval discretization of continuousvalued attributes for classification learning, 1993. In: 13th Int'l Joint Conf. Artificial Intelligence (IJCAI) (1993)
13. Garcia, S., Luengo, J., Sáez, J.A., Lopez, V., Herrera, F.: A survey of discretization techniques: Taxonomy and empirical analysis in supervised learning. IEEE Transactions on Knowledge and Data Engineering **25**(4), 734–750 (2012)
14. Janosi, A., Steinbrunn, W., Pfisterer, M., Detrano, R.: Heart Disease. UCI Machine Learning Repository (1988)
15. Joiţa, D.: Unsupervised static discretization methods in data mining. Titu Maiorescu University, Bucharest, Romania (2010)
16. Kalish, M.: DC public employee salaries. `https://data.world/codefordc/dc-public-employee-salaries-2011` (2011)
17. Kaushik, M.: Datasets. `https://github.com/minakshikaushik/LSQM-measure.git` (2022)
18. Kaushik, M., Sharma, R., Peious, S.A., Draheim, D.: Impact-driven discretization of numerical factors: Case of two- and three-partitioning. In: Srirama, S.N., Lin, J.C.W., Bhatnagar, R., Agarwal, S., Reddy, P.K. (eds.) Big Data Analytics. pp. 244–260. Springer International Publishing, Cham (2021)
19. Kaushik, M., Sharma, R., Peious, S.A., Shahin, M., Yahia, S.B., Draheim, D.: On the potential of numerical association rule mining. In: International Conference on Future Data and Security Engineering. pp. 3–20. Springer (2020)
20. Kaushik, M., Sharma, R., Peious, S.A., Shahin, M., Yahia, S.B., Draheim, D.: A systematic assessment of numerical association rule mining methods. SN Computer Science **2**(5), 1–13 (2021)
21. Konno, H., Kuno, T.: Best piecewise constant approximation of a function of single variable. Operations research letters **7**(4), 205–210 (1988)
22. Kotsiantis, S., Kanellopoulos, D.: Discretization techniques: A recent survey. GESTS International Transactions on Computer Science and Engineering **32**(1), 47–58 (2006)

23. Liu, H., Hussain, F., Tan, C.L., Dash, M.: Discretization: An enabling technique. Data mining and knowledge discovery **6**(4), 393–423 (2002)
24. Liu, H., Setiono, R.: Feature selection via discretization. IEEE Transactions on Knowledge and Data Engineering **9**(4), 642–645 (1997). https://doi.org/10.1109/69.617056
25. Lud, M.C., Widmer, G.: Relative unsupervised discretization for association rule mining. In: European conference on principles of data mining and knowledge discovery. pp. 148–158. Springer (2000)
26. Naik, S.: NJ teacher salaries. `https://data.world/sheilnaik/nj-teacher-salaries-2016` (2016)
27. Peious, S.A., Sharma, R., Kaushik, M., Shah, S.A., Yahia, S.B.: Grand reports: a tool for generalizing association rule mining to numeric target values. In: International Conference on Big Data Analytics and Knowledge Discovery. pp. 28–37. Springer (2020)
28. Quinlan, J.R.: Induction of decision trees. Machine Learning **1**(1), 81–106 (1986)
29. Shahin, M., Arakkal Peious, S., Sharma, R., Kaushik, M., Ben Yahia, S., Shah, S.A., Draheim, D.: Big data analytics in association rule mining: A systematic literature review. In: International Conference on Big Data Engineering and Technology (BDET). p. 40–49. Association for Computing Machinery (2021)
30. Sharma, R., Kaushik, M., Peious, S.A., Bazin, A., Shah, S.A., Fister, I., Yahia, S.B., Draheim, D.: A novel framework for unification of association rule mining, online analytical processing and statistical reasoning. IEEE Access **10**, 12792–12813 (2022). https://doi.org/10.1109/ACCESS.2022.3142537
31. Sharma, R., Kaushik, M., Peious, S.A., Yahia, S.B., Draheim, D.: Expected vs. unexpected: selecting right measures of interestingness. In: International Conference on Big Data Analytics and Knowledge Discovery. pp. 38–47. Springer (2020)
32. Srikant, R., Agrawal, R.: Mining quantitative association rules in large relational tables. In: Proceedings of the 1996 ACM SIGMOD international conference on Management of data. pp. 1–12 (1996)
33. Tatu, A., Bak, P., Bertini, E., Keim, D., Schneidewind, J.: Visual quality metrics and human perception: An initial study on 2d projections of large multidimensional data. In: Proceedings AVI'10. p. 49–56. Association for Computing Machinery, New York, NY, USA (2010). https://doi.org/10.1145/1842993.1843002

Curriculum Vitae

Personal Data

Name	Minakshi Kaushik
Date and place of birth	23 September 1985, Bulandshahr, U.P, India
Nationality	Indian

Contact Information

Address	School of Information Technologies, Tallinn University of Technology Akadeemia tee 15a, 12618 Tallinn, Estonia
E-mail	minakshi.kaushik@taltech.ee

Education

2019–2024	PhD studies Information and Communication Technology, Tallinn University of Technology, School of Information Technologies
2010–2012	M.Tech, Computer Science Engineering, Dr. A.P.J. Abdul Kalam Technical University, Lucknow, Uttar Pradesh, India
2005–2008	B.Tech, Computer Science Engineering, Dr. A.P.J. Abdul Kalam Technical University,

Language Competence

Hindi	native
English	fluent

Professional Employment

2019–2024	Early Stage Researcher, Information Systems Group Department of Software Science, Tallinn University of Technology
2013–2019	Raj Kumar Goel Institute of Technology, Dr. A.P.J. Abdul Kalam Technical University, India
2010–2013	Sunderdeep Engineering College, Dr. A.P.J. Abdul Kalam Technical University, India
2009–2010	RCVGIT, Dr. A.P.J. Abdul Kalam Technical University, India
2008–2009	MIT, Dr. A.P.J. Abdul Kalam Technical University, India

Fields of Research[1]

- 4.6. Computer Science
- 4.7. Information and Communications Technologies

[1] Estonian Research Information System (ETIS) fields of research

Scientific Work

1. M. Kaushik, R. Sharma, S. A. Peious, M. Shahin, S. B. Yahia, and D. Draheim. On the potential of numerical association rule mining. In *Proceedings of FDSE: 7th International Conference on Future Data and Security Engineering*, pages 3–20, Vietnam, 2020. Springer

2. M. Kaushik, R. Sharma, S. A. Peious, M. Shahin, S. B. Yahia, and D. Draheim. A systematic assessment of numerical association rule mining methods. *SN Computer Science*, 2(5):1–13, 2021

3. M. Kaushik, R. Sharma, S. A. Peious, and D. Draheim. Impact-driven discretization of numerical factors: Case of two- and three-partitioning. In *Proceedings of BDA: 9th International Conference on Big Data Analytics*, pages 244–260, Cham, 2021. Springer International Publishing

4. M. Kaushik, R. Sharma, M. Shahin, S. A. Peious, and D. Draheim. An analysis of human perception of partitions of numerical factor domains. In *Proceedings of iiWAS 2022 – the 24th International Conference on Information Integration and Web Intelligence*, pages 137–144, Cham, 2022. Springer Nature Switzerland

5. M. Kaushik, R. Sharma, A. Vidyarthi, and D. Draheim. Discretizing numerical attributes: An analysis of human perceptions. In *Proceedings of ADBIS 2022 – the 26th International Conference on New Trends in Database and Information Systems*, pages 188–197, Cham, 2022. Springer International Publishing

6. M. Kaushik, R. Sharma, I. Fister Jr.2, and D. Draheim. Numerical association rule mining: A systematic literature review, arxiv, 2307.00662, 2023

7. M. Kaushik, R. Sharma, and D. Draheim. Discretizing numerical attributes: An analysis of human perceptions (extended version). arxiv:2311.03278, 2023

8. R. Sharma, M. Kaushik, S. A. Peious, A. Bazin, S. A. Shah, I. Fister, S. B. Yahia, and D. Draheim. A novel framework for unification of association rule mining, online analytical processing and statistical reasoning. *IEEE Access*, 10:12792–12813, 2022

9. R. Sharma, M. Kaushik, and D. Draheim. On statistical paradoxes and overcoming the impact of bias in artificial intelligence: towards fair and trustworthy decision making. *SSRN Electronic Journal*, pages 1–107, 2022

10. R. Sharma, M. Kaushik, S. A. Peious, S. B. Yahia, and D. Draheim. Expected vs. unexpected: Selecting right measures of interestingness. In M. Song, I.-Y. Song, G. Kotsis, A. M. Tjoa, and I. Khalil, editors, *Proceedings of DaWaK 2020 – the 22nd International Conference on Big Data Analytics and Knowledge Discovery*, pages 38–47, Cham, 2020. Springer International Publishing

11. R. Sharma, H. Garayev, M. Kaushik, S. A. Peious, P. Tiwari, and D. Draheim. Detecting simpson's paradox: A machine learning perspective. In C. Strauss, A. Cuzzocrea, G. Kotsis, A. M. Tjoa, and I. Khalil, editors, *Proceedings of DEXA 2022 – the 33rd International Conference on Database and Expert Systems Applications*, pages 323–335, Cham, 2022. Springer International Publishing

12. R. Sharma, M. Kaushik, S. A. Peious, M. Bertl, A. Vidyarthi, A. Kumar, and D. Draheim. Detecting simpson's paradox: A step towards fairness in machine learning. In S. Chiusano, T. Cerquitelli, R. Wrembel, K. Nørvåg, B. Catania, G. Vargas-Solar, and E. Zumpano, editors, *Proceedings of ADBIS 2022 – the 26th International Conference on New Trends in Database and Information Systems*, pages 67–76, Cham, 2022. Springer International Publishing

13. R. Sharma, M. Kaushik, S. A. Peious, M. Shahin, A. S. Yadav, and D. Draheim. Towards unification of statistical reasoning, olap and association rule mining: Semantics and pragmatics. In A. Bhattacharya, J. Lee Mong Li, D. Agrawal, P. K. Reddy, M. Mohania, A. Mondal, V. Goyal, and R. Uday Kiran, editors, *Proceedings of DASFAA 2022 – the 27th International Conference on Database Systems for Advanced Applications*, pages 596–603, Cham, 2022. Springer International Publishing

14. R. Sharma, M. Kaushik, S. A. Peious, M. Shahin, A. Vidyarthi, P. Tiwari, and D. Draheim. Why not to trust big data: Discussing statistical paradoxes. In U. K. Rage, V. Goyal, and P. K. Reddy, editors, *Proceedings of DASFAA 2022 International Workshops – the 27th International Conference on Database Systems for Advanced Applications*, pages 50–63, Cham, 2022. Springer International Publishing

15. R. Sharma, M. Kaushik, S. A. Peious, M. Shahin, A. Vidyarthi, and D. Draheim. Existence of the yule-simpson effect: An experiment with continuous data. In *Proceedings of Confluence 2022 – the 12th International Conference on Cloud Computing, Data Science & Engineering*, pages 351–355, 2022

16. S. A. Peious, R. Sharma, M. Kaushik, S. A. Shah, and S. B. Yahia. Grand reports: a tool for generalizing association rule mining to numeric target values. In *International Conference on Big Data Analytics and Knowledge Discovery*, pages 28–37. Springer, 2020

17. M. Shahin, S. Arakkal Peious, R. Sharma, M. Kaushik, S. Ben Yahia, S. A. Shah, and D. Draheim. Big data analytics in association rule mining: A systematic literature review. In *International Conference on Big Data Engineering and Technology (BDET)*, page 40–49. Association for Computing Machinery, 2021

18. M. Shahin, S. Saeidi, S. A. Shah, M. Kaushik, R. Sharma, S. A. Peious, and D. Draheim. Cluster-based association rule mining for an intersection accident dataset. In *2021 International Conference on Computing, Electronic and Electrical Engineering (ICE Cube)*, pages 1–6, 2021

19. M. Shahin, M. R. Heidari Iman, M. Kaushik, R. Sharma, T. Ghasempouri, and D. Draheim. Exploring factors in a crossroad dataset using cluster-based association rule mining. *Procedia Computer Science*, 201:231–238, 2022. The 13th International Conference on Ambient Systems, Networks and Technologies (ANT) / The 5th International Conference on Emerging Data and Industry 4.0 (EDI40)

Elulookirjeldus

1. Isikuandmed

Nimi	Minakshi Kaushik
Sünniaeg ja -koht	23 September 1985, U.P, India
Kodakondsus	India

2. Kontaktandmed

Aadress	Tallinna Tehnikaülikool, Infotehnoloogia teaduskond, Tarkvarateaduse instituut, Infosüsteemide rühm, Akadeemia tee 15a, 12618 Tallinn, Estonia
E-post	minakshi.kaushik@taltech.ee

3. Haridus

2019–2024	Tallinna Tehnikaülikool, Infotehnoloogia teaduskond, Info- ja kommunikatsioonitehnoloogia, doktoriõpe
2010–2012	M.Tech, Arvutiteadus ja tehnika, Dr A.P.J. Abdul Kalami Tehnikaülikool, Lucknow, U.P, India
2005–2008	B.Tech, Arvutiteadus ja tehnika, Dr A.P.J. Abdul Kalami Tehnikaülikool, Lucknow, U.P, India

4. Keelteoskus

hindi keel	emakeel
inglise keel	kõrgtase

5. Teenistuskäik

2019–2023	Infosüsteemide grupp, Tarkvarateaduse osakond, Tallinna Tehnikaülikool, analüütik
2013–2019	Raj Kumar Goeli Tehnoloogiainstituut (Dr. A.P.J. Abdul Kalami Ülikool) Professori abi
2010–2013	Sunderdeep Engineering College (Dr. A.P.J. Abdul Kalami Ülikool), Professori abi
2009–2010	RCVGIT, (Dr. A.P.J. Abdul Kalami Ülikool), Lektor
2008–2009	MIT, (Dr. A.P.J. Abdul Kalami Ülikool), Lektor

Teadustöö põhisuunad[2]

- 4.6. Arvutiteadused
- 4.7. Info- ja kommunikatsioonitehnoloogia

11. Teadustegevus

Teadusartiklite, konverentsiteeside ja konverentsiettekannete loetelu on toodud ingliskeelse elulookirjelduse juures.

[2] Eesti Teadusinfosüsteemi (ETIS) teadusvaldkondade ja -erialade klassifikaator

ISSN 2585-6901 (PDF)
ISBN 978-9916-80-107-9 (PDF)