

Effective Reduct Computation Strategies

A thesis submitted to the University of Hyderabad
in partial fulfillment of the requirements for the award of

Doctor of Philosophy

in

Computer Science

by

P.S.V.S. SAI PRASAD



School of Computer and Information Sciences

University of Hyderabad

Hyderabad – 500 046

India

June 2014

CERTIFICATE

This is to certify that the thesis entitled “**Effective Reduct Computation Strategies**” submitted by **P.S.V.S. Sai Prasad** bearing Reg. No. **09MCPC13** in partial fulfillment of the requirements for the award of **Doctor of Philosophy** in **Computer Science** is a bonafide work carried out by him under my supervision and guidance.

The thesis has not been submitted previously in part or in full to this or any other University or Institution for the award of any degree or diploma.

Prof. C. Raghavendra Rao
Professor
School of CIS
University of Hyderabad
Hyderabad – 500046

Prof. Arun K. Pujari
Dean
School of CIS
University of Hyderabad
Hyderabad – 500046

DECLARATION

I, **P.S.V.S. Sai Prasad**, hereby declare that this thesis entitled “**Effective Reduct Computation Strategies**” submitted by me under the guidance and supervision of **Prof. C. Raghavendra Rao** is a bonafide research work. I also declare that it has not been submitted previously in part or in full to this University or any other University or Institution for the award of any degree or diploma.

Date:

Signature of the student

Name: P.S.V.S. SAI PRASAD

Reg. No: 09MCPC13

ACKNOWLEDGMENTS

I express my sincere gratitude and indebtedness to my supervisor Prof. C. Raghavendra Rao. It was a rare privilege to obtain his constant guidance and encouragement. His immense knowledge in the subject, earnest desire of sharing the same have given the shape and substance to this research work.

I would like to acknowledge my DRC members Prof. Rajeev Wankar and Dr. Alok Singh for their valuable suggestions for improving the thesis.

I would like to express my sincere thanks to Prof. Arun K. Pujari, Dean, School of Computer and Information Sciences for his cooperation. I convey my deep regards and gratitude to honorable Vice Chancellor Prof. Ramakrishna Ramaswamy for the moral support extended at various stages.

I thank my colleagues of School of Computer and Information Sciences for all good wishes. I would like to thank office and technical staff of School of Computer and Information Sciences, Administration staff of University of Hyderabad for the support extended.

I owe special thanks to Prof. Arun Agarwal, Dr. K. Swarupa Rani for their valuable suggestions and help in my research period. My special thanks to my co research scholars Dr. K. Hima Bindu, Dr. P. Pavan Kumar, Mr. M. Raghava for invaluable support given during this period.

I am thankful to Dr. Hong Mei Chen of Southwest Jiatong University, China for assisting me in getting a translation of a Chinese paper used in the comparative study in this thesis. I am grateful to Prof. Roman Slowinski, Prof. Qiang Shen for sharing their research contributions in response to my request. I also sincerely acknowledge Prof. Domink Slezak, Prof. Sheela Ramanna for their interactions during visit to Hyderabad, which helped immensely in my research work. I would like to gratefully thank Prof. Mihir Chakraborty for the elderly support and advice extended during my visit to China for JRS 2012.

My sincere thanks to all my friends whose support is a great strength for my life's sojourn.

With deep sense of gratitude, I convey thanks and appreciation to my parents, my brother's and sister's family members, for their patience and constant encouragement and also for sparing me from several responsibilities so that I can focus on research work.

Nothing is possible and feasible without the grace of Almighty God. My prayers and thanks for His "Avyaja Prema" (Unconditional Love) bestowed on me.

ABSTRACT

Feature selection is an important pre processing technique in Data Mining or Soft Computing applications. Feature selection process removes the irrelevant and counter contributing attributes in the given decision or information system to enable to build models, like classifiers and clustering, through unsupervised and supervised learning strategies to overcome the problem of Curse of Dimensionality.

Prof. Z. Pawlak developed Rough Sets in 1982, which has subsequently become a popular soft computing paradigm. Rough sets provides the concept of “Reduct” for feature selection. Reduct has been successfully used for feature selection in many diverse applications for last three decades. Rough sets and its extensions have given raise to reduct computation approaches for Complete Symbolic, Incomplete Symbolic and Hybrid decision systems.

This thesis aims at developing effective reduct computation strategies (ERCS) for these three systems. The present study adopts a sequential forward selection paradigm by addressing inherent redundancies and its consequent ambiguities as first phase. The second part of the thesis addresses scalability issues with the help of novel sampling strategies proposed.

Classical quick reduct algorithm (QRA) for complete symbolic decision systems, has been computationally enhanced by incorporation of granular refinement, efficient positive region computation, positive region removal. Methodologies, using secondary heuristics, are developed for resolving ambiguity situations arising in QRA’s attribute selection process. The proposed Improved QRA (IQRA) algorithms are effective in reducing dispensable attributes in super reduct and overcome, to a large extent, the problem of Order Variance observed in QRA.

Sampling based reduct computation approaches, categorized into persistent and non persistent approaches, based on random and stratified sampling techniques are developed. Proposed approaches overcome the limitation of existing approaches, which can only result in subset of reduct, by wrapping with global validation. Non persistent approaches are recommended because of their ability to reduce the dispensable attributes and computational efficiency.

Existing reduct computation approaches for incomplete decision systems exhibit redundancy in computing positive region because of object wise similarity class computation. A novel granular framework is introduced in this study, enabling sub granular computation of similarity class, resulting in formulation of new definitions of lower approximation, positive region and information entropy. The-

oretical validation is given for positive region removal incorporated in proposed IQRAIG_Incomplete algorithm.

Reduct computation in hybrid decision systems is a much complex task owing to higher space and time complexities. Efficient approaches are proposed in the thesis using Radzikowska-Kerry's fuzzy rough set model by incorporation of proposed simplified computational model and absolute positive region removal. Theoretical validation is given for simplified model and absolute positive region removal. Algorithm IMQRA, as an improvement to existing MQRA algorithm, is introduced using the theoretical developments. Algorithm NPSS_IMQRA is developed using non persistent sampling approach.

Seed based scalable approaches are introduced in the thesis for incomplete and hybrid decision systems. In these approaches, incomplete or hybrid decision system is reduced to complete symbolic decision system and reduct obtained with reduced system is taken as the seed for evolving to reduct in respective decision systems. These approaches are found to be effective and provided an alternate approach for scalable reduct computation.

All the proposed methods were empirically found to be better or comparable to existing approaches based on experiments with various benchmark database repositories. Design of hybrid methodologies of ERCS in parallel and distributed environment is under consideration as a future work to meet some of the requirements of the emerging fields like data science.

Contents

ABSTRACT	i
1 Introduction	1
1.1 Dimensionality Reduction and Feature Selection	1
1.2 Rough sets for feature selection	3
1.3 Motivation	4
1.4 Problem Definition	4
1.5 Major Contributions	5
1.6 Organization of the Thesis	8
2 Rough Sets and approaches for Reduct computation	10
2.1 Overview of Classical Rough Sets	10
2.2 Extensions to Rough Sets	13
2.2.1 Variable Precision Rough Sets	13
2.2.2 Extension of Rough Sets for Incomplete Symbolic Decision Systems	14
2.2.3 Fuzzy Rough Sets for Hybrid Decision Systems	15
2.3 Reduct Computation in Complete Symbolic Decision Systems .	15
2.4 Motivation for ERCS	19
3 Extensions of Quick Reduct Algorithm	20
3.1 Quick Reduct Algorithm	20
3.2 Efficient implementation of QRA	22
3.2.1 Granular space	22
3.2.2 Granular Refinement approach	23
3.2.3 Positive Region Computation	24

3.2.4	Efficient QRA	24
3.3	Strategies for overcoming limitations of QRA	25
3.3.1	Positive Region Removal in QRA	25
3.3.2	Order Variance in QRA	26
3.3.3	Trivial Ambiguous situations in QRA	26
3.3.4	Non Trivial Ambiguity Situations in QRA	27
3.4	Improved Algorithms	27
3.4.1	Improved Quick Reduct Algorithm (IQRA)	27
3.4.2	Improved Quick Reduct Algorithm for Non Trivial Ambiguity (IQRA_NonTrivial)	29
3.5	Heuristics for Ambiguous Situations	30
3.5.1	Heuristics for Trivial Ambiguous Situations	31
3.5.2	Heuristic for Non Trivial Ambiguous Situations	34
3.6	Experiments and Results	34
3.6.1	Experiments with trivial ambiguous datasets	34
3.6.2	Experiments with non trivial ambiguous datasets	36
3.6.3	Comparative Experiments with Existing Methods	36
3.6.4	Analysis of Results	40
3.7	Summary	42
4	Sampling based Scalable Approach for Reduct Computation in Symbolic Decision Systems	43
4.1	Existing approaches and Limitations	44
4.2	Relevance of Sampling in the context of Rough Set based Attribute Reduction	44
4.3	Proposed Approach for Sample Based Reduct Computation	47
4.4	Persistent Sampling Based Approaches	49
4.4.1	Sample guided IQRA_IG (SGIQRA_IG) Algorithm	50
4.4.2	Stratified sample guided IQRA_IG (SSGIQRA_IG) Algorithm	51

4.5	Non Persistent Sampling Based Approach	52
4.5.1	Non Persistent Stratified Sampling based IQRA_IG (NPSS_IQRAIG) Algorithm	52
4.6	Experiments and Results	55
4.7	Analysis of Results	57
4.7.1	Analysis of Results with large decision systems	57
4.7.2	Analysis of Results of Comparative Experiments with SARA algorithm	70
4.8	Summary	70
5	Reduct Computation in Incomplete Decision Systems	72
5.1	Introduction and Literature Review	72
5.2	Extension of Rough Sets for Incomplete Decision Systems	74
5.2.1	Limitations of Existing Approaches	75
5.3	Novel Granular Framework for Incomplete Decision Systems	76
5.3.1	Properties	77
5.3.2	Granular Refinement	78
5.3.3	Positive Region using Granular Framework	81
5.3.4	Information Entropy of $Gr(U, A)$ in IDS	81
5.4	IQRAIG_Incomplete: Reduct algorithm using Granular Framework	83
5.4.1	Algorithm IQRAIG_Incomplete	83
5.4.2	Computation of $Gr(U, C)$	84
5.4.3	Validation of Positive Region Removal	85
5.4.4	Complexity Analysis of IQRAIG_Incomplete	87
5.5	Experiments and Results	87
5.5.1	Experiments on benchmark Incomplete Decision Systems	88
5.5.2	Classification Analysis using 3NN and Stability Analysis	88
5.5.3	Experiments with Trivial Ambiguous Incomplete Decision Systems	90

5.5.4	Comparative Experiments	91
5.6	Analysis of Results	94
5.7	Summary	95
6	Reduct Computation in Hybrid Decision Systems	96
6.1	Hybrid Decision Systems and Extensions of Rough Sets	96
6.2	Overview of Fuzzy Decision Reduct	100
6.2.1	Fuzzy Rough Set Theory	101
6.2.2	Modified Quick Reduct Algorithm	104
6.3	Improvements to Modified Quick Reduct Algorithm	105
6.3.1	Theoretical foundations	105
6.3.2	Improved Modified Quick Reduct Algorithm (IMQRA)	111
6.3.3	Modeling of IMQRA computations using Vector Operations	111
6.3.4	Analysis of IMQRA Algorithm	113
6.4	Illustration	114
6.4.1	Positive Region computation using conventional approaches	114
6.4.2	Positive region computation using vector operations	115
6.4.3	Illustration of IMQRA_MW algorithm	116
6.5	Non persistent Sampling based Scalable approach for Fuzzy Decision Reduct	117
6.5.1	Relevance of Sampling	118
6.5.2	NPSS_IMQRA algorithm	120
6.5.3	Analysis of NPSS_IMQRA Algorithm	122
6.6	Experiments and Results	123
6.6.1	Experiments using IMQRA_MW algorithm	124
6.6.2	Experiments using NPSS_IMQRA algorithm	125
6.7	Analysis of Results	128
6.7.1	Analysis of IMQRA_MW Results	128
6.7.2	Analysis of NPSS_IMQRA Results	130

6.8	Summary	131
7	Seed based Fuzzy Decision Reduct for Hybrid Decision Systems	132
7.1	Complexity Hierarchy of Reduct Computation	132
7.2	Seed based Reduct Computation in Incomplete Decision Systems	134
7.2.1	Transforming IDS into CDS	134
7.2.2	IQRA_IG for seed computation	135
7.2.3	Algorithm IQRAIG_Incomplete_Seed	135
7.2.4	Experiments and Results	137
7.2.5	Analysis of Results	139
7.3	Seed based Reduct Computation in Hybrid Decision Systems . .	142
7.3.1	Proposed Approach for Seed based Reduct Computation in Hybrid Decision Systems	144
7.3.2	Selection of Relevant Discretization Method	145
7.3.3	Empirical comparison of IQRA_IG and IMQRA_MW al- gorithms	146
7.3.4	Seed based Fuzzy Decision Reduct	147
7.3.5	Experiments and Results	149
7.3.6	Analysis of Results	151
7.4	Summary	153
8	Conclusion	156
8.1	Summary of Contributions	156
8.2	Future Scope of Work	156
	References	158

List of Tables

3.1	Details of Symbolic Decision Systems containing Trivial or Non Trivial Ambiguities	35
3.2	Reduct Lengths and Excess Count for Trivial Ambiguous Decision Systems	35
3.3	Computational Time in Seconds for Trivial Ambiguous Decision Systems	35
3.4	Order Variance Experiment Results for Trivial Ambiguous Datasets	36
3.5	Reduct Length and Excess Count Summary for Non Trivial Ambiguous Datasets	36
3.6	Computational Time for Non Trivial Ambiguous Datasets . . .	37
3.7	Order Variance Experiment Results for Non Trivial Ambiguous Datasets	37
3.8	Datasets used in FSPA based algorithms	38
3.9	Reduct Length comparison with FSPA based algorithms	38
3.10	Computational Time (in seconds) comparison with FSPA based algorithms	38
3.11	Datasets used in Comparative analysis with THQARA algorithm	39
3.12	Reduct lengths in comparative analysis with THQARA algorithm	39
3.13	Computational Time (in seconds) in comparative analysis with THQARA algorithm	39
4.1	Details of Datasets and IQRA_IG Results	55
4.2	Summary of SGIQRA_IG Results for 100 Trails	56
4.3	Summary of SSGIQRA_IG Results for 100 Trails	57
4.4	Summary of NPSS_IQRAIG Results for 100 Trails	58
4.5	Results of SARA algorithm reported from [44]	58
4.6	Comparative Results with SARA algorithm (SG: SGIQRA_IG, SSG: SSGIQRA_IG, NPSS: NPSS_IQRAIG)	58

4.7	Coverage Values for a single trail on Basehock Dataset	68
4.8	Iteration wise Attribute Selection (Local) CT vs Global Validation CT in a single trail of NPSS_IQRAIG on Basehock Dataset . .	68
4.9	P-values for t-test results on BASEHOCK dataset	69
4.10	P-values for t-test results on PCMAC dataset	69
4.11	P-values for t-test results on RELATHE dataset	69
5.1	Example of CDS	76
5.2	Example of IDS	76
5.3	Details of Incomplete Decision Systems	88
5.4	IQRAIG_Incomplete Results	88
5.5	Tenfold Experiment results on benchmark datasets	90
5.6	Trivial Ambiguity Experiment Results	91
5.7	Order Variance Experiment Results on Trivial Ambiguous Datasets	91
5.8	CT Comparison of existing and proposed algorithms in tenfold ex- periment	92
5.9	Stability Comparison of IFSPA-IPR with IQRAIG_Incomplete in tenfold experiment	93
6.1	Example Decision System	114
6.2	Fuzzy sets $R_{\{A\},x1}$, $R_{d,x1}$	114
6.3	Implication Value Matrix $\mathfrak{S}_{\{A\}}$ Computation	115
6.4	Details of Hybrid Datasets	124
6.5	Comparison Results of IMQRA_MW and MQRA_MW with FA- FSCE and FSCE algorithms	124
6.6	Reduct Size Comparison with FRFS and associated algorithms .	125
6.7	CT (in seconds) Comparison with FRFS and associated algorithms	125
6.8	Classification Results in Tenfold Experiment	126
6.9	Reduct Sizes in Tenfold Experiment	126
6.10	CT in tenfold experiments for IMQRA_MW and NPSS_IMQRA algorithms	127

6.11	Detailed comparative Results with Spambase dataset	127
7.1	Details of Incomplete Decision Systems	138
7.2	IQRAIG_Incomplete and IQRAIG_Incomplete_Seed Results .	138
7.3	IQRAIG_Incomplete_Seed Tenfold Experiment results on bench- mark datasets	139
7.4	CT (in Seconds) Comparison of existing [112] and proposed algo- rithms in tenfold experiment	139
7.5	Stability Comparison of IFSPA-IPR with IQRAIG_Incomplete in tenfold experiment	139
7.6	Details of Hybrid Decision Systems	149
7.7	Results of Reduct computation (M(Seed Reduct), Size of reduct and computational time) using entire data sets for Seed Reduct (SR), FDSR_Seed (FS), FDSR (F)	150
7.8	Average Result of Reduct computation (M(Seed Reduct), Size of reduct and computational time) in 10xrepeated tenfold cross vali- dation for Seed Reduct (SR), FDSR_Seed (FS), FDSR(F) . . .	151
7.9	The accuracy results (Mean accuracy, 95% Confidence interval) for 3nn, MLP, cart classifiers in 10x repeated 10 fold cross validation using attributes (all), Seed Reduct (SR), FDSR_Seed (FS), FDSR (F)	152

List of Figures

1.1	Summary of Contributions for Complete Symbolic Decision Systems (CDS), Incomplete Symbolic Decision Systems (IDS), Hybrid Decision Systems (HDS)	6
4.1	Excess Count Frequency Comparison for BASEHOCK Dataset 5% Sample Size Trails	59
4.2	Excess Count Frequency Comparison for BASEHOCK Dataset 10% Sample Size Trails	60
4.3	Excess Count Frequency Comparison for BASEHOCK Dataset 15% Sample Size Trails	60
4.4	Excess Count Frequency Comparison for BASEHOCK Dataset 20% Sample Size Trails	61
4.5	Excess Count Frequency Comparison for BASEHOCK dataset for NPSS_IQRAIG algorithm for all sample sizes	62
4.6	Excess Count Frequency Comparison for PCMAC Dataset 5% Sample Size Trails	62
4.7	Excess Count Frequency Comparison for PCMAC Dataset 10% Sample Size Trails	63
4.8	Excess Count Frequency Comparison for PCMAC Dataset 15% Sample Size Trails	63
4.9	Excess Count Frequency Comparison for PCMAC Dataset 20% Sample Size Trails	64
4.10	Excess Count Frequency Comparison for PCMAC dataset for NPSS_IQRAIG algorithm for all sample sizes	65
4.11	Excess Count Frequency Comparison for RELATHE Dataset 5% Sample Size Trails	65
4.12	Excess Count Frequency Comparison for RELATHE Dataset 10% Sample Size Trails	66
4.13	Excess Count Frequency Comparison for RELATHE Dataset 15% Sample Size Trails	66

4.14	Excess Count Frequency Comparison for RELATHE 20% Sample Size Trails	67
4.15	Excess Count Frequency Comparison for Relathe dataset for NPSS_IQRAIG algorithm for all sample sizes	68
4.16	Reduct Length Comparison of Comparative Results with SARA	71
5.1	CT Comparison of QRAIG_Incomplete vs IQRAIG_Incomplete for various sizes of dataset	92
5.2	CT (range $\mu - \sigma$ to $\mu + \sigma$) Comparison of IPR, QRAIG_Incomplete, IFSPA-IPR, IQRAIG_Incomplete	93
5.3	Stability (range $\mu - \sigma$ to $\mu + \sigma$) comparison in tenfold experiment	93
6.1	Computational Gain Percentage of IMQRA_MW over FSCE, FA-FSCE, MQRA_MW algorithms	128
6.2	Computational Gain Percentage of IMQRA_MW over FSCE, FA-FSCE, MQRA_MW algorithms	128
6.3	Reduction in Dataset Size in IMQRA_MW iterations in FA-FSCE comparative Experiment	129
6.4	Reduction in Dataset Size in IMQRA_MW iterations in FRFS Comparative Experiment	130
7.1	Complexity of Reduct Computation in Hybrid, Incomplete and Complete Symbolic Decision Systems	133
7.2	Computational Time Comparison of IQRA_IG and IQRAIG_Incomplete Algorithms for varying random dataset sizes	136
7.3	Computational Time Comparison of IQRA_IG and IQRAIG_Incomplete for varying percentages of missing values	136
7.4	3NN Performance Comparison (range $\mu - \sigma$ to $\mu + \sigma$) of Tenfold Experiment	140
7.5	Reduct Length Comparison (Mean $\pm$ Standard Deviation) of Tenfold Experiment	141
7.6	Stability Comparison (Mean $\pm$ Standard Deviation) of Tenfold Experiment	141
7.7	CT Comparison (range $\mu - \sigma$ to $\mu + \sigma$) of Tenfold Experiment	142

7.8	CT Comparison (range $\mu - \sigma$ to $\mu + \sigma$) of Tenfold Experiment with IFSPA, IPR algorithms	143
7.9	Stability Comparison of Tenfold Experiment (range $\mu - \sigma$ to $\mu + \sigma$) with IFSPA, IPR algorithms	143
7.10	CT Comparison of IQRA_IG and IMQRA_MW Algorithms for varying random hybrid dataset sizes	147
7.11	95% Confidence Interval for 3NN Classification Experiment Results	154
7.12	95% Confidence Interval for MLP Classification Experiment Results	154
7.13	95% Confidence Interval for CART Classification Experiment Results	155

NOTATIONS & ABBREVIATIONS

- β : Precision parameter of Variable Precision Rough Sets
- γ : Gamma Measure
- ϵ : Real value near to zero assuming values between 0 to 1
- μ_X^R : Rough membership function
- Γ : t-norm
- $\Im$: Implication
- $|B|$: Cardinality of set B
- $\underline{B}X$: Lower approximation of X defined using indiscernibility relation IND_B
- $\overline{B}X$: Upper approximation of X defined using indiscernibility relation IND_B
- $\underline{B}^\beta(X)$: Lower approximation in Variable Precision Rough Sets with β precision
- $\overline{B}^\beta(X)$: Upper approximation in Variable Precision Rough Sets with β precision
- $R \downarrow X$: Lower approximation notation used in fuzzy rough sets
- $R \uparrow X$: Upper approximation notation used in fuzzy rough sets
- $[x]_R$: Equivalence class of an object x using equivalence relation R
- $ABSOLUTE_POS$: Absolute positive region
- AL : Attributes (Reduct) Length
- C : Set of conditional attributes
- C^c : Set of categorical (symbolic) conditional attributes
- C^h : Set of hybrid conditional attributes
- C^n : Set of numeric (quantitative) conditional attributes
- CDS : Complete Symbolic Decision System
- CT : Computational Time
- D : Set of decision attributes
- d : Single decision attribute
- DM : Data Mining
- DT : Decision Table
- EC : Excess Count
- $ERCS$: Effective Reduct Computation Strategies

- f_a : Information mapping from set of objects U to set of domain values V_a of an attribute a in a decision system
- $FDSR$: Fuzzy M -decision super reduct
- $FDSR - \alpha$: Fuzzy M -decision super reduct to degree α
- $FDSR_Seed$: Fuzzy M -decision super reduct using seed attributes
- g : Granule (a set of objects) belonging to granular space
- $Gr(U, A)$: Granular space defined on a set of objects U . Granular space is formed using IND_A if U represents objects of CDS else if U represents objects of IDS then using $SIM(A)$
- H : Information Entropy
- HDS : Hybrid Decision System
- HDT : Hybrid Decision Table
- IC : Iteration Count
- IDS : Incomplete Symbolic Decision System
- IG : Information Gain
- $IMQRA$: Improved Modified Quick Reduct Algorithm
- $IMQRA_MW$: Improved Modified Quick Reduct Algorithm with vector based operations
- $IMQRA_Seed$: Seed based Improved Modified Quick Reduct Algorithm
- IND : Indisernibility relation
- $IQRA$: Improved Quick Reduct Algorithm
- $IQRA_ENSEMBLE$: Improved Quick Reduct Algorithm using Ensemble of Heuristics
- $IQRA_IG$: Improved Quick Reduct Algorithm using Information Gain Heuristic
- $IQRAIG_Incomplete$: Improved Quick Reduct Algorithm using Information Gain Heuristic for Incomplete Symbolic Decision Systems
- $IQRAIG_Incomplete_Seed$: Seed based Improved Quick Reduct Algorithm using Information Gain Heuristic for Incomplete Symbolic Decision Systems
- $IQRAIG_NonTrivial$: Improved Quick Reduct Algorithm for Non Trivial Ambiguity Situations Using Information Gain Heuristic
- $IQRA_MD$: Improved Quick Reduct Algorithm using Maximum Domain Cardinality Heuristic

- *IQRA_VPRS*: Improved Quick Reduct Algorithm using Variable Precision Rough Sets Heuristic
- *MD*: Maximum Domain Cardinality
- *MQRA*: Modified Quick Reduct Algorithm
- *MQRA_MW*: Modified Quick Reduct Algorithm with vector based operations
- *NOD*: Number of decision classes
- *NPSS_IQRAIG*: Non persistent Stratified Sampling based Improved Quick Reduct Algorithm using Information Gain Heuristic
- *NPSS_IMQRA*: Non persistent Stratified Sampling based Improved Modified Quick Reduct Algorithm
- *POS*: Positive Region
- $pos\beta_a$: Positive beta of an attribute yet to be included into reduct set
- *QRA*: Quick Reduct Algorithm
- *R*: Recommended reduct of reduct computation algorithms
- *RC*: Reduct Count
- $S_A(x)$: Similarity class of an object x using similarity relation $SIM(A)$
- *SBS*: Sequential Backward Selection Search Strategy
- *SFS*: Sequential Forward Selection Search Strategy
- *SGIQRA_IG*: Sample Guided Improved Quick Reduct Algorithm using Information Gain Heuristic
- $SIM(A)$: Similarity relation defined using set of attributes A
- *SS*: Sample Size
- *SSGIQRA_IG*: Stratified Sample Guided Improved Quick Reduct Algorithm using Information Gain Heuristic
- *U*: Set of Objects
- V_a : Set of domain values of an attribute a in a decision system

Chapter 1

Introduction

1.1 Dimensionality Reduction and Feature Selection

The advent of Internet, World Wide Web, Sensor Networks has resulted in massive data accumulation in recent years. This resulted in intensifying dynamic nature and versatility in behavior in current systems. The classical approaches based on the expert will not likely to work and a need for machine based automatic analysis of data is inevitable. This opened different discussion forums for providing strategies for meeting future requirements. The following are few deliberations emphasizing the needs of the future directions.

On August 8th 2000, as part of American Mathematical Society's meeting on "Mathematical Challenges of the 21st Century", a major event in celebration of World Mathematical Year 2000, Prof. David L. Donoho delivered a talk entitled "High Dimensional Data Analysis: The Curses and Blessings of Dimensionality". Prof. Donoho remarking about utter ubiquity of data collection said, "It is no exaggeration to say that the twenty first century may well be century of data" [24]. He further remarked "We are in the era of massive automatic data collection, systematically obtaining many measurements, not knowing which ones will be relevant to the phenomenon of interest. Our task is to find a needle in a haystack, teasing the relevant information out of a vast pile of glut."

A report [93] entitled "Leadership Under Challenge: Information Technology R & D in a Competitive World" was prepared by President's Council of Advisers on Science and Technology (PCAST) and submitted to President of USA on August 10, 2007. The report says, "The sheer volume of new data - the data deluge - is overwhelming the capacity of institutions to manage it and researchers to make use of it." The report emphasizes the vision of "a data and knowledge indexing system that could augment human activities by making the relevant information available whenever needed."

Both deliberations point out the essential need of efficient extraction of relevant information from data. The aspect of extraction of relevant information from data

is named as data reduction in Data Mining (DM). Data Mining [30] evolved as an important discipline with the aim of extracting knowledge and patterns from data. In preprocessing stage of data mining, denormalized data from multiple sources is represented as a single decision system or information system to be used for building knowledge, knowledge representation and for design and construction of inference engine.

Effective representation system will aid the data modelers as well as enhances the understanding of the system. Dimensionality of the decision system (cardinality of conditional attributes¹) influences the efficiency of the intended DM model. Principle of Curse of Dimensionality (COD) [5] states that number of objects needed to train classifier function grows exponentially with number of attributes. In other words, given a decision system size, there is a maximum number of features, above which the performance of the classifier will rather degrade than improve. Presence of irrelevant attributes in decision system deteriorate the performance of DM model [12, 67]. Hence, dimensionality reduction becomes an essential step in the pre-processing stage of any DM application. Approaches for dimensionality reduction are categorized into feature extraction methods and feature selection methods.

In feature extraction, a projection of all features into lower number of dimensions is obtained. Popular approaches in feature extraction include principal component analysis (PCA), linear discriminant analysis (LDA) and factor analysis (FA) [32, 157]. In feature selection [37, 119], an appropriate subset of the original features is obtained by removing the irrelevant and inconsistent features. The primary advantage of feature selection approach [60] is that the obtained classifier is interpretable in terms of the original features. The important advantages of feature selection are [119],

- Improving the generalization performance of DM model
- Interpretability of model leading to more insight into underlying process in predictor model
- Storage requirements being optimized
- Future evaluation of all features is drastically reduced and more appropriate in case of cost of evaluation of all futures is expensive.
- Simple models leading to reduction of construction and evaluation times of model

Feature selection approaches are developed for supervised, unsupervised, and semi-supervised scenarios. Further feature selection approaches are categorized into filter, wrapper and embedded approaches. In filter approaches, filtering of

¹Attribute is used in synonymous with feature as well as dimension.

irrelevant features occurs by using an evaluation function which depends only on the data. In wrapper approaches, evaluation of features for selection is done using the intended DM model. In embedded methods, feature selection implicitly occurs in the process of DM model construction such as decision trees. Both embedded and wrapper approaches are biased towards a particular DM model. The primary advantage of filter approaches is being unbiased towards any DM model and are potential to achieve good generalization performance with all DM models. Filter approaches are computationally faster compared to wrapper approaches. A good review of filter based approaches for feature selection is given in [37, 118, 119]. Present study is restricted to filter based supervised feature selection approaches.

1.2 Rough sets for feature selection

In 1982, Prof. Z. Pawlak introduced the concept of *rough sets* [97] for dealing with uncertainty and vagueness in decision and information systems. In rough sets, a vague concept is represented by two crisp sets called lower and upper approximations. Rough sets has evolved as an important soft computing paradigm with many applications in the areas of data mining and soft computing [22, 71, 94, 100–102, 131, 139, 153]. Rough sets provide a non invasive data mining approach for knowledge discovery in databases (KDD) [28]. Rough sets, as a soft computing paradigm, has complemented in resonance with other soft computing technologies like neural networks, fuzzy sets and several successful hybrid models were developed such as Rough Fuzzy-MLP [2], Rough Fuzzy-SOM [95], Rough Neural Networks [74], Rough genetic algorithms [40, 75], Fuzzy Rough SVM [15]. Fuzzy rough sets and rough fuzzy sets [25, 26] have evolved as an important hybrid paradigms combining the strengths of fuzzy sets and rough sets.

Applications of rough sets are found in wide range of domains [100]. Some of the application areas of rough sets include medical domain, control systems, biological systems, pharmacy and engineering. The extensive summary of rough set applications are reported in [64].

Rough set based data analysis [166, 173] preliminarily consists of obtaining a minimal representation of decision system through filter based feature selection using rough set principles. Subset of features selected using rough set principles is named as reduct in case of information systems, relative reduct in case of decision systems. In the rest of the thesis the word “reduct” refers to “relative reduct” [139] as the thesis is confined to decision systems. Reduct is a minimal subset of the conditional attributes preserving the classifying ability of the decision system. Reduct computation is distinguishable from other feature selection approaches as it is data driven method which does not need any external parameters like

probabilities, beliefs, membership values etc.

1.3 Motivation

Reduct computation attracted several researchers [140] and various algorithms have been proposed. Most of these algorithms don't consider the inherent redundancies cropping up during reduct computation. This led us to designing efficient reduct computational algorithms which will be meeting set constraints as well as objectives. Mostly the existing algorithms paid less attention and very little treatment for the instances which have large volume of decision table due to more attributes, more objects or both. Scalability [77] is another important aspect that should be embedded into reduct computation algorithms. These voluminous tables were mostly addressed by researchers [162] using parallel and distributed paradigms which demand heavy post processing and integrable infrastructure.

1.4 Problem Definition

The classical indiscernibility relation given by Prof Z. Pawlak has lead to reduct algorithms for complete symbolic decision systems. The basis of any rough set based approach is deduced from indiscernibility relation. To quote, Y.Y. Yao [173],

There are two fundamental issues: representations of indiscernibility and discernibility, and attribute reduction (information table simplification) based on indiscernibility and discernibility.

Using tolerance relation instead of indiscernibility relation lead to reduct algorithms for incomplete symbolic decision systems [66]. An extension of rough sets, known as fuzzy rough sets [25, 26], defining the relation to be a fuzzy tolerance relation, given raise to reduct algorithms for hybrid decision systems.

This thesis aims at developing effective and scalable reduct computation strategies for these three systems. The study adopts a sequential forward selection paradigm by addressing inherent redundancies and its consequent ambiguities as first phase. The second part of the thesis addresses scalability issues with the help of novel sampling strategies proposed.

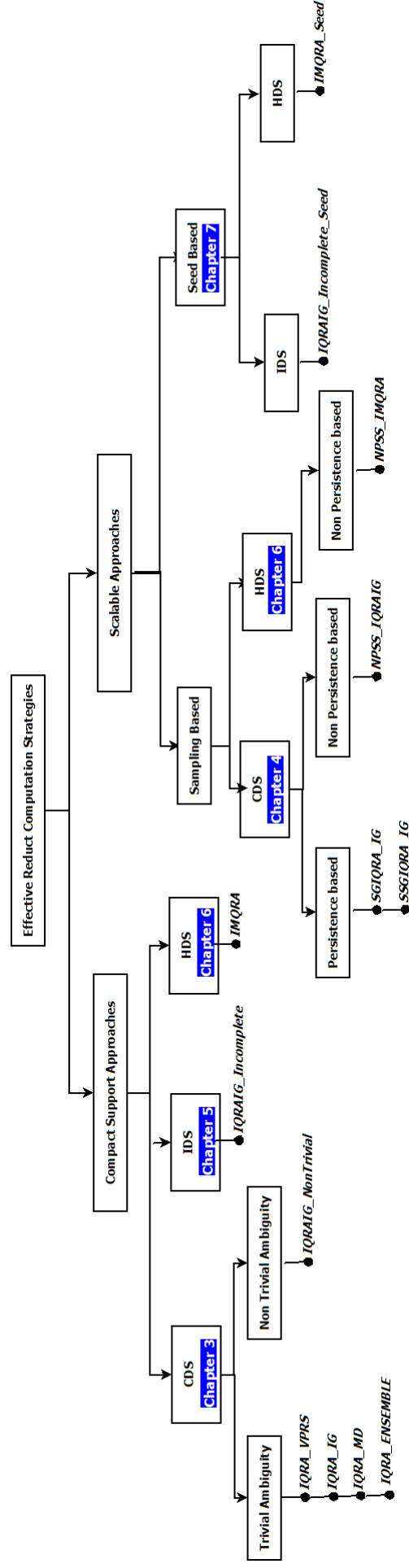
1.5 Major Contributions

The contributions of the thesis can be viewed as compact support approaches, sampling based scalable approaches for three types of decision systems. The objective of compact support is that indiscernibility relation or similarity relation is applied only to those objects which impact the decision making. Hence only necessary objects participate in positive region computations and further algorithms are formulated with intelligent ways of dealing with ambiguous situations in the attribute selection process. A scalable reduct computation algorithm needs to be adaptable to the given memory and CPU constraints of the system. The algorithm needs to be able to find reduct for the given decision system even if the decision system is very large in size and could not be loaded into memory at once. Popularly there are two approaches for arriving at scalable solutions. The first way is development of parallel and distributed algorithm and the second is to use sampling approach. Sampling approach has an advantage over distributed approach as minimal computational infrastructure is sufficient to implement the algorithm. Summary of contributions made for all three decision systems under compact support approaches and scalable approaches is depicted in Figure 1.1.

Publications based on the contributions of thesis

1. P. S. V. S. Sai Prasad, C. Raghavendra Rao: *IQuickReduct: An improvement to Quick Reduct algorithm*. In: In Proceedings of 12th International Conference on Rough Sets, Fuzzy Sets, Data Mining and Granular Computing (RSFDGrC'09), vol. 5908, LNCS, Springer-Verlag Berlin Heidelberg, pp. 152-159. (2009) [Indexed in DBLP, Scopus]
2. P. S. V. S. Sai Prasad, K. Hima Bindu, C. Raghavendra Rao: *Incremental learning in attributenets with dynamic reduct and IQuickReduct*. In: Proceedings of the 6th International Conference on Rough Sets and Knowledge Technology (RSKT'11), vol. 6954, LNCS, Springer-Verlag Berlin Heidelberg, pp. 195-200. (2011) [Indexed in DBLP, Scopus]
3. P. S. V. S. Sai Prasad, C. Raghavendra Rao: *Extensions to IQuickReduct*. In: Proceedings of the 5th Multi-disciplinary International Workshop on Artificial Intelligence (MIWAI'11), vol. 7080, LNCS, Springer-Verlag Berlin Heidelberg, pp. 351-362. (2011) [Indexed in DBLP, Scopus]
4. P. S. V. S. Sai Prasad, C. Raghavendra Rao: *Novel Granular Framework for Attribute Reduction in Incomplete Decision Systems*. In: Proceedings of the 6th Multi-disciplinary International Workshop on Artificial Intelligence

Figure 1.1: Summary of Contributions for Complete Symbolic Decision Systems (CDS), Incomplete Symbolic Decision Systems (IDS), Hybrid Decision Systems (HDS)



(MIWAI'12), vol. 7694, LNCS, Springer-Verlag Berlin Heidelberg, pp. 188-201. (2012) [Indexed in DBLP, Scopus]

5. P. S. V. S. Sai Prasad, C. Raghavendra Rao: *Scalable Improved Quick Reduct: Sample based*. In: Proceedings of 7th International Conference on Rough Sets and Knowledge Technology (RSKT'12), vol. 7414, LNCS, Springer-Verlag Berlin Heidelberg, pp. 34-39. (2012) [Indexed in DBLP, Scopus]
6. P. S. V. S. Sai Prasad, C. Raghavendra Rao: *Seed based Fuzzy Decision Reduct for Hybrid Decision Systems*. In: 2013 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE'13), IEEE, pp. 1-8, DOI: 10.1109/FUZZ-IEEE.2013.6622535. (2013) [Indexed in DBLP, Scopus]
7. P. S. V. S. Sai Prasad, C. Raghavendra Rao: *An Efficient Approach for Fuzzy Decision Reduct Computation*. Transactions on Rough Sets XVII, LNCS, vol. 8375, Springer-Verlag Berlin Heidelberg, pp. 82-108. (2014) [Indexed in DBLP]
8. P. S. V. S. Sai Prasad, C. Raghavendra Rao: *Non Persistent Stratified Sampling Based IQRA_IG for Scalable Reduct Generation*. International Journal of Granular Computing, Rough Sets and Intelligent Systems, vol. 3, no. 3, Inderscience, 234-255. (2014) [Indexed in DBLP]

Additional Relevant Publications

1. P. Venugopal, Y. Rama Devi, P. S. V. S. Sai Prasad: *Sensitivity Analysis of Fuzzy Parameters in the Computation of Reduct*. Proceedings of International Conference on Emerging Trends in Computer science, Communication and Information Technology (CSCIT), Nanded, Jan 9-11. (2010)
2. Y. Rama Devi, P. Venu Gopal, P. S. V. S. Sai Prasad: *Impact analysis of Jensen and Sk pal fuzzification in classification*. 1st Amrita ACM-W Celebration on Women in Computing in India, September 2010, Amrita University, Coimbatore. (2010) [Indexed in DBLP, Scopus]
3. K. Hima Bindu, P. S. V. S. Sai Prasad, C. Raghavendra Rao: *Hybrid Decision tree based on inferred attribute*. 1st Amrita ACM-W Celebration on Women in Computing in India, September 2010, Amrita University, Coimbatore. (2010) [Indexed in DBLP, Scopus]
4. Y. Rama Devi, P. Venu Gopal, P. S. V. S. Sai Prasad: *Fuzzy Rough Data Reduction Using SVD*. International Journal of Computer and Electrical Engineering vol. 3, no. 3, pp. 384-388. (2011)

5. B. Manasa, P. S. V. S. Sai Prasad, C. Raghavendra Rao: *Inference in Rough Set Flow Graphs: A Dynamic Programming Approach*. Proceedings of the National Conference on Emerging Trends in Soft Computing, Nowrosjee Wadia College, Pune, 2-3 Feb. (2011)

1.6 Organization of the Thesis

Contributions made in the thesis can be classified into methods using total decision table and methods using sampled decision table. Other categorization is based on the nature of the decision system i.e., complete symbolic, incomplete symbolic and hybrid decision systems.

The chapterization of the thesis has been made based on methods developed for complete symbolic decision systems, incomplete symbolic decision systems and hybrid decision systems and a set of new concepts introduced for addressing issues associated with incomplete and hybrid decision systems leading to seed reduct are presented.

Chapter 1 is the introduction of the thesis. The chapter reviews dimensionality reduction, feature selection and in particular feature selection based on reduct computation using rough sets. The motivation for designing effective and scalable computational techniques for reduct is presented. The contributions of the thesis and organization of the thesis are presented.

Chapter 2 contains the basic concepts of rough sets and reduct computation. A brief overview of extensions to rough sets such as variable precision rough sets, tolerance rough sets, fuzzy rough sets is presented. Literature survey of reduct computation in complete symbolic decision systems is presented.

Chapters 3 and 4 deal with total decision table of complete symbolic form. Chapter 3 highlights limitations of quick reduct algorithm. IQRA algorithm and its variants are developed taking account of redundancy as well as trivial and non trivial ambiguities. The algorithms developed in this chapter are IQRA_VPRS, IQRA_IG, IQRA_MD, IQRA_Ensemble, IQRAIG_NonTrivial. The contributions of this chapter are published in the proceedings of RSFDGrC-2009, RSKT-2011 and MIWAI-2011.

Chapter 4 contains literature review of scalable approaches for reduct computation in complete symbolic decision systems and discusses the sampling based approaches like SARA and FSR in detail. Sampling based reduct computation strategies are presented for dealing with scalability issues with large decision tables. Relevance of sampling in the context of reduct computation is given. Random sampling, stratified sampling based persistent approaches and strati-

fied sampling based non persistent approaches and associated algorithms are provided. Algorithms developed in this chapter are SGIQRA_IG, SSGIQRA_IG and NPSS_IQRAIG. The contributions of this chapter are published in the proceedings of RSKT-2012 and International Journal of Granular Computing, Rough Sets and Intelligent Systems (IJGCRSIS).

Incomplete symbolic decision systems and its implication on reduct computation is considered in Chapter 5. Literature review of existing approaches for reduct computation in incomplete decision systems is discussed and limitations of existing approaches are identified. This chapter contains proposed novel granular framework, new definitions of lower approximation, positive region, information entropy and algorithm IQRAIG_Incomplete. Theoretical validation of positive region removal using proposed granular framework is presented. Comparative analysis of IQRAIG_Incomplete with existing approaches is given. It is to be noted that a scalable approach which is seed based for incomplete decision systems is provided in chapter 7. This work is published in the proceedings of MIWAI-2012.

Chapter 6 is devoted for developing reduct computation strategies for hybrid decision tables. This chapter contains theory developed for simplified computation model for positive region in Radzikowska-Kerry's fuzzy rough set model and validation of absolute positive region removal. Improved MQRA algorithm for fuzzy decision super reduct computation by incorporating proposed modifications in MQRA algorithm is presented. The extension of IMQRA for the scalability aspects is presented in NPSS_IMQRA algorithm using non persistent treatment developed in Chapter 4. The contributions of this chapter are published in the proceedings of RSKT-2012 and Transactions on Rough Sets.

Chapter 7 describes the seed based scalable approaches for reduct computation in incomplete symbolic decision systems and hybrid decision systems. Algorithms IQRAIG_Incomplete_Seed and IMQRA_Seed are developed in Chapter 7. Experimental results containing comparative analysis with existing approaches is reported. This work is published in the proceedings of FUZZ-IEEE-2013.

Chapter 8 provides the summary of the contributions and future scope of the work.

Chapter 2

Rough Sets and approaches for Reduct computation

This chapter provides overview of classical rough sets in Section 2.1 and its extensions in Section 2.2. Reduct computation methods for complete symbolic decision systems are reviewed in Section 2.3 and motivating remarks for proposed effective reduct computation strategies (ERCS) are provided in Section 2.4.

2.1 Overview of Classical Rough Sets

This section gives overview of classical rough sets introduced by Prof. Z. Pawlak in 1982 based on [54, 97–99, 101] and in the context of decision system. The overview is restricted to the scope of the thesis.

A set in classical set theory is crisp because of no ambiguity in determining the belongingness of an object into set. An object either belongs to set or does not belong. Vagueness arises when ambiguity is present in defining belongingness such as defining belongingness to concepts like “intelligent”, “beautiful” etc. Fuzzy sets [169] deals with vagueness by defining fuzzy membership function taking values between 0 and 1. Frege [33, 101] referred to vagueness as the presence of boundary region, a region in which objects cannot be said to be either belonging or not belonging to set. Rough sets is a soft computing paradigm dealing with vagueness in the viewpoint of Frege.

Approximation Space and Rough Set

A pair (U, R) , where U is the set of objects of universe and R an equivalence relation defined over $U \times U$, is called as an approximation space. R is called as indiscernibility relation as it relates indistinguishable objects. Equivalence classes of R are called as elementary sets and are represented by U/R . An equivalence class of an object $x \in U$ is denoted by $[x]_R$. Any finite union of elementary sets is called as a definable set. Given a concept, $X \subseteq U$, lower approximation $\underline{R}X$ and upper approximation $\overline{R}X$ of X are defined as,

$$\underline{R}X = \{x \in U / [x]_R \subseteq X\} \quad (2.1)$$

$$\overline{R}X = \{x \in U / [x]_R \cap X \neq \emptyset\} \quad (2.2)$$

Hence lower approximation of X , using the knowledge of R , contains all objects completely belonging to X and upper approximation contains objects possibly belonging to X . The region $\overline{R}X - \underline{R}X$ is called boundary region and contains objects whose belongingness into X is vague. The region $U - \overline{R}X$ is called as negative region and contains objects which do not belong to X . X is said to be crisp in approximation space (U, R) if boundary region of X is empty otherwise X is called as rough set.

The definitions of lower and upper approximations in Eq. (2.1), Eq. (2.2) are equivalently expressed in terms elementary sets as,

$$\underline{R}X = \bigcup_{Y \in U/R \wedge Y \subseteq X} Y \quad (2.3)$$

$$\overline{R}X = \bigcup_{Y \in U/R \wedge Y \cap X \neq \emptyset} Y \quad (2.4)$$

In classical set theory, set is represented using characteristic function defined on universe mapping '1' to all elements belonging to set and '0' to all elements in the complement of set. A rough membership function is defined in rough sets for representing the possible membership between '0' and '1' for boundary region objects. The rough membership function $\mu_X^R : U \rightarrow [0, 1]$ is defined as,

$$\mu_X^R(x) = \frac{|[x]_R \cap X|}{|[x]_R|} \quad \forall x \in U \quad (2.5)$$

where for any set K , $|K|$ denotes cardinality of set K . The nature of rough membership function defined in Eq. (2.5) is to map lower approximation objects to '1', boundary region objects to a value between '0' and '1' and negative region objects to '0'.

The formalism of rough sets is applied to complete symbolic decision systems (tables) [98] consisting of complete information about objects of the universe for symbolic attributes¹. A complete symbolic decision system is represented by $DT = (U, C \cup D, \{V_a, f_a\}_{a \in C \cup D})$ where U is the set of objects. C is the set of symbolic conditional attributes. D is the set of decision attributes. V_a is the non empty set of domain values of attribute $a \in C \cup D$ and f_a is a information mapping from U to V_a associating a value for each object under attribute a .

The indiscernibility relation IND_B based on set of attributes $B \subseteq C \cup D$ is

¹Symbolic is used synonymous with nominal, qualitative.

defined as,

$$IND_B = \{(x, y) \in U \times U / \forall a \in B, f_a(x) = f_a(y)\} \quad (2.6)$$

Since IND_B is an equivalence relation, (U, IND_B) defines the approximation space for defining rough sets. The equivalence classes of IND_B are represented by U/IND_B . The conventional usage is to omit IND [54] and represent equivalence classes by U/B . The elementary set of an object $x \in U$ is represented by $B(x)$ and lower and upper approximations of concept $X \subseteq U$ in approximation space (U, IND_B) are represented as $\underline{B}X$ and $\overline{B}X$ respectively. Rough membership function is represented by notation μ_X^B .

Dependency of attributes and Reduct

Determining the dependency between attributes of a decision system is essential for performing feature selection. A set of attributes B depends totally on a set of attributes A , denoted by $A \implies B$ if values of attributes from B are completely determined by values of A . The complete dependency exists when the partition of elementary sets (equivalence classes) of IND_A are finer than partition of elementary sets of IND_B . To obtain the measure or extent of dependency that exist between two subsets of attributes, a measure called gamma measure $\gamma_A(B)$ is defined as,

$$\gamma_A(B) = \frac{|POS_A(B)|}{|U|} \quad (2.7)$$

where $POS_A(B)$ denotes set of positive region objects where,

$$POS_A(B) = \bigcup_{X \in U/B} \underline{A}(X) \quad (2.8)$$

Hence positive region denotes the union of lower approximations of elementary sets of IND_B in the approximation space defined by A , i.e., (U, IND_A) . Positive region can also be viewed as the collection of objects which are classifiable into elementary sets of B without ambiguity using the attributes information of A . $\gamma_A(B) = 1$ represents the complete dependence of B on A .

The dependency of decision attributes D on conditional attributes C is of particular importance. Gamma measure $\gamma_C(D)$ is a measure of classifiability of the decision system. If gamma measure $\gamma_C(D)$ is 1, indicating all objects are classifiable, then the decision system is said to be consistent otherwise is called as inconsistent decision system.

An attribute $a \in B \subseteq C$ is said to be indispensable, if $\gamma_{B-\{a\}}(D) < \gamma_B(D)$ otherwise is dispensable in B . Dispensable attributes are redundant and do not contribute to the classifiability of the system. Removal of dispensable attributes from C gives rise to a reduced set of attributes having the same classifiability as the total conditional attributes C . The reduced set of attributes is named as reduct and is formally defined as,

Set of attributes $B \subseteq C$ is a reduct if $\gamma_B(D) = \gamma_C(D)$ and B consists of indispensable attributes, i.e., $\forall a \in B, \gamma_{B-\{a\}}(D) < \gamma_B(D)$.

A decision system can have many reducts. Intersection of all reducts of a decision system is called as core set of attributes. It is to be noted that core of a decision system can be empty. Core is same as set of indispensable attributes in C .

In view of simpler DM model construction an experimenter can further remove the least significant attributes in reduct by compromising a manageable loss of classification performance. This results in ϵ -approximate reduct. Given an ϵ , a very small positive value near to zero, a set of attributes $B \subseteq C$ is an ϵ -approximate reduct [125] if,

$$\gamma_B(D) \geq ((1 - \epsilon) \times \gamma_C(D)) \quad (2.9)$$

and none of the proper subsets of B satisfies Eq. (2.9).

In the rest of thesis, sub reduct (subset of reduct) refers to ϵ -approximate reduct and super reduct (super set of reduct) refers to a set of attributes which contain a reduct as the subset.

2.2 Extensions to Rough Sets

In the evolution of rough sets in last three decades, several extensions were proposed to rough sets with the aim of extending the classical rough sets to realistic generalized scenarios. What follows is the brief overview of extensions of rough sets restricted to the scope of the thesis.

2.2.1 Variable Precision Rough Sets

Variable precision rough sets (VPRS) is an extension of rough sets introduced by Wojciech Ziarko [175] in 1991. VPRS is introduced to overcome the deterministic nature of classical rough sets in defining positive and negative regions. The classification into lower approximation and thereby into positive region needs to fully

correct or certain in case of classical rough sets. To quote Ziarko [175], “classification with a controlled degree of uncertainty is outside the realm of this approach (i.e., classical rough sets)”.

To accommodate controlled degree of uncertainty in classical rough set model, a precision parameter β is introduced in VPRS. Parameter β takes values between 0.5 to 1 [6]. The definitions of lower and upper approximations are given in VPRS as,

$$\underline{B}^\beta(X) = \{x \in U / \mu_X^B(x) \geq \beta\} \quad (2.10)$$

$$\overline{B}^\beta(X) = \{x \in U / \mu_X^B(x) \geq 1 - \beta\} \quad (2.11)$$

The definitions of positive region and gamma measure are extended using lower approximation given in Eq. (2.10). VPRS is equivalent to classical rough sets at $\beta = 1$. VPRS became the basis for other extensions of rough sets like probabilistic rough sets [163], decision theoretic rough sets (DTRS) [165].

2.2.2 Extension of Rough Sets for Incomplete Symbolic Decision Systems

Indiscernibility relation based classical rough sets have in fact given a new framework for feature selection, knowledge representation. The classical rough sets can have various perceptions. A relation based perception of classical rough sets gave birth to develop extensions for both incomplete symbolic decision systems, hybrid decision systems. The equivalence relation properties are relaxed in developing extensions to rough sets using relational perspective.

Incomplete symbolic decision systems contain missing values for some of the objects and occur frequently in real world decision systems owing to lost values or values immeasurable because of domain constraints etc. Classical rough sets needs to be extended for dealing with incomplete symbolic decision systems. Kryszkiewicz [66] developed an extension to rough sets for dealing with incomplete symbolic decision systems. The modification results in tolerance relation in contrast to equivalence relation of classical rough sets. Several modifications were proposed for Kryszkiewicz extension of rough sets [85, 132, 148]. Some more details of the extensions in this direction are given in Sections 5.1, 5.2 of Chapter-5.

2.2.3 Fuzzy Rough Sets for Hybrid Decision Systems

Hybrid decision systems comprise mixture of quantitative (numeric) and qualitative (symbolic or nominal) attributes. Application of classical rough sets to hybrid decision systems results in representations whose generalization ability is limited. In particular feature selection in such systems is considered to be a computationally intensive and hard problem and usually dealt by discretization of quantitative attributes [37, 76, 88]. Discretization offers a solution but inherently results in information loss.

The developments in rough sets are aptly summarized as a movement from indiscernibility to similarity [108]. Indiscernibility or equality is replaced by similarity for dealing with hybrid decision systems [128]. Tolerance rough sets [65, 123] is an extension of classical rough sets by using tolerance similarity relation for dealing with quantitative attributes in hybrid decision systems. Fuzzy rough sets [25, 26] introduced by Dubois and Prade in 1990 is much more information preserving extension of classical rough sets bringing the strengths of both fuzzy sets and rough sets together. Fuzzy rough sets incorporate fuzzy similarity relation instead of indiscernibility relation. Fuzzy rough sets bring graded indiscernibility into measuring the similarity between objects based on quantitative attributes. Some more details of the extensions in this direction are given in Section 6.2 of Chapter-6.

2.3 Reduct Computation in Complete Symbolic Decision Systems

A decision system can have many reducts. A DM model can be constructed using a single reduct [49, 140]. Minimum description length (MDL) principle states best hypothesis for a given set of data is the one that leads to the best compression of data [116]. In view of complexity involved in building any DM model using reduct attributes and based on MDL principle, size of the reduct places a significant role. A shorter reduct implies compact knowledge representation resulting in simpler DM model with less computational costs. But finding such minimal structure for knowledge representation is NP Hard [13, 50]. In particular in rough sets, Skowron et al. [122] have developed the concept of discernibility matrix and a methodology of boolean reasoning for computing all reducts of decision system. It is shown that computation of all reducts is equivalent to finding prime implicants of a logical expression in conjunctive normal form (CNF) [103, 122]. Hence the exhaustive algorithm of finding all reducts of a decision system through discernibility matrix is proved to be NP-Hard and hence arriving at a minimum length reduct is NP-hard [122]. This milestone theoretical result of rough sets lead to emergence of a

fertile area of research in rough sets with the objective of finding a reduct, using heuristic methods, which is near optimal to a minimum length reduct.

Review of reduct computational approaches are provided in [23, 54, 140]. In this section an overview of reduct computation approaches for complete symbolic decision systems are presented.

A reduct computation algorithm is characterized by a search strategy (control structures) and attribute selection heuristic. Search strategies used in reduct computation algorithms are primarily sequential forward selection based (*SFS*) and sequential backward selection based (*SBS*) [167]. In an *SFS* algorithm, search for reduct subset starts with an empty set and in each iteration an attribute is included into reduct set which optimizes the attribute selection heuristic. In an *SBS* algorithm, reduct is initialized to all attributes and in each iteration the least significant dispensable attribute is removed from reduct set based on attribute selection heuristic. There are approaches which are of hybrid nature containing both forward and backward search strategies as part of the algorithm.

Maciej Modrzejewski [83] developed PRESET algorithm for feature selection in binary decision systems which ranks the features and selects the significant features till relevance criteria is satisfied by construction of a decision tree. PRESET algorithm, like other rank based feature selection approaches [119], suffers from redundant attributes being present in the feature subset selected. Ron Kohavi et al. [63] developed wrapper approach for reduct computation in an algorithm named Holte-II algorithm. Holte-II algorithm computes sub reduct using best first search approach and a feature subset is evaluated for classification performance by using a table-majority classifier with bootstrapping approach. Wrapper approach and best first search control structure impacts the complexity of Holte-II algorithm and even though a good reduct can be obtained using best first search, such complexity can not scale well with large decision systems.

Discernibility matrix [122] computation requires $O(|C||U|^2)$ space and time complexities and making reduct computation algorithms based on discernibility matrix are not scalable to large decision systems [42]. Nguyen H.S. et al. [42] proposed efficient approaches for computing positive region, lower, upper approximations and heuristic reduct computation based on discernibility matrix that avoid construction and storing the discernibility matrix. A super reduct computation algorithm was developed in [42], which is a filter based *SFS* approach using attribute selection heuristic based on Johnson's strategy [61] for computing minimal prime implicant of CNF expression.

Several heuristic reduct computation approaches were developed based on gamma measure as the attribute selection heuristic. An *SBS* approach based heuristic reduct computation algorithm [52] was developed using gamma measure

and used as a pre processor for neural network based classifier construction. It is empirically established [52] that rough set based feature selection drastically reduces the computational and space complexity of neural network without significant degradation in classification performance. Alexios Chouchoulas et al. have developed Quick Reduct algorithm (QRA) [18]. QRA is a *SFS* based algorithm, for obtaining super reduct, using gamma measure as the attribute selection heuristic. Quick reduct was used in developing a classifier for categorization of E-mail messages. Authors of QRA have proposed implementation level developments for QRA in [17]. Accelerated QRA was proposed by Thangavel et al. [106] considering a set of attributes instead of a single attribute in an iteration. S.H. Liu et al. developed an algorithm [120] similar to QRA using gamma measure as the attribute selection heuristic. Z.Y. Xu et al. improved algorithm in [120] by using radix sort instead of quick sort for computing elementary sets of partition in rough approximation space [177]. Y. Liu et al. used hashing schemes to further improve algorithm of [177] and developed an algorithm named “Twice-Hash Attribute Reduction Algorithm” (THQARA) [168]. The time complexity of Algorithm THQARA in [168] is $O(|C|^2 |U/C|)$ and is considered as one of the most efficient algorithm for super reduct computation using gamma measure and *SFS* approach [23].

Another attribute selection measure widely employed in rough set based feature selection is information entropy. The approach of entropy reduct was introduced by Prof. Andrzej Skowron in 1993/94 as part of lectures given in Warsaw University [111]. Prof. Dominik Slezak formulated shannon entropy based reduct algorithms [124]. Prof. Slezak developed entropy sub reduct (approximate reduct) computation algorithms [125]. Wang et al. developed algorithms [38] namely conditional entropy based algorithm for reduction of knowledge with computing core (CEBARKCC), conditional entropy based algorithm for reduction of knowledge without computing core (CEBARKNC).

Several new attribute selection heuristics were developed by researchers with the objective of finding near to minimum length reduct. Zhong et al. developed reduct computation based on a combination heuristic of gamma measure and maximum size of indiscernibility class of positive region [174]. This algorithm is an algorithm of hybrid search strategy using *SBS* approach for core computation followed by computing super reduct using *SFS* approach. Han et al. introduced a new measure of relative dependency [39] between attributes based on number of equivalence classes and theoretically established the similarities between gamma measure and relative dependency measure for consistent decision systems. *SFS* and *SBS* based reduct computation algorithms were introduced [39] based on relative dependency heuristic. Measures similar to entropy were introduced and reduct computation algorithms were developed such as decision power heuristic [160], rough entropy measure [73], measure of approximate accuracy of rough

classification (Liang conditional entropy) [70], extension information entropy [16], combination heuristic of entropy and frequency of attributes [147], combination entropy [110] etc.

Yuhua Qian et al., using the notion of positive approximation, developed reduct computation algorithms [111] using representative of heuristics for attribute selection criteria based on gamma measure, shannon conditional entropy, Liang conditional entropy and combination conditional entropy. The corresponding reduct algorithms are FSPA-PR, FSPA-SCE, FSPA-LCE, FSPA-CCE respectively. FSPA algorithms follow hybrid search strategy of computing core initially using *SBS* and performing remaining attributes selection using *SFS* approach.

Several reduct computation methods were developed using evolutionary approaches. Jakub Wroblewski [158, 159] introduced genetic algorithm (GA) based approaches for reduct computation. It is identified that most time consuming operation is fitness evaluation of the candidate subset of features and GA based approaches have scalability issues while dealing with large decision systems. Bjorvand et al. [9] have introduced improvements to GA based reduct computation such as incorporating core in candidate subsets for arriving at good initial population, using dynamic mutation rate for avoiding wrong search paths and utilizing domain knowledge to arrive at an initial estimate of reduct length. GA based approaches for reduct computation were implemented in software systems for KDD using rough sets such as “Rough Enough” software system [8], Rough Set Exploration System [4], Rosetta [94]. Dominik Slezak et al. have extended the approach of GA reduct computation [158] for finding approximate entropy reduct [127]. The approach of quantum genetic algorithm (QGA) was used for obtaining reduct by Y.J. Lv and N.X. Liu [79]. Jensen et al. introduced reduct computation approaches with Ant Colony Optimization (ACO) [55], Particle Swarm Optimization (PSO) [151]. Refined approach for ACO based reduct computation was given in [62]. Bing Xu et al. [161] proposed new approaches of initializations and global, local best updation mechanisms for PSO based reduct computation. Bee Colony Optimization (BCO) was used by N. Suguna and K. Thanushkodi [135] for reduct computation.

The experiments and results reported for evolutionary reduct computation algorithms [135, 151], in which a comparative study of respective evolutionary reduct algorithm with filter based approaches is conducted, establish the following conclusion. Evolutionary approaches when compared to heuristic filter based approaches have advantages and shortcomings. Primary advantage of filter approaches is being scalable to large decision systems but have presence of redundancy in the super reduct generated. Advantages of evolutionary approaches are being multivariate but computationally not scalable compared to filter approaches [118].

2.4 Motivation for ERCS

The scalability of reduct computation algorithm is highly influenced by the complexity of attribute selection heuristic and search strategy. Two primary attribute selection heuristics in rough sets are gamma measure developed as an outcome of rough set theory and entropy based information theoretic measures. Most of the other measures used in literature are derived from these two measures.

The algorithms developed for reduct computation in this thesis use gamma measure as the primary attribute selection heuristic. The choice is due to the fact that gamma measure evolved out of core rough set theory principles. In comparative analysis done with other attribute selection heuristics [111] gamma measure resulted in similar length reducts compared to other attribute selection heuristics while preserving stability. Further the techniques developed in the thesis are adaptable to other attribute selection heuristics.

The *SFS* search strategy based filter selection approaches are computationally faster and suitable for developing scalable solutions [118] compared to *SBS* approaches and evolutionary methods. *SBS* approaches require entire data to be loaded into memory which can become a severe constraint for scalability.

The objective of effective reduct computation strategies (ERCS) is to develop scalable approaches for complete symbolic, incomplete symbolic and hybrid decision systems as mentioned in problem definition (section 1.4). The base framework for scalable approaches is formed by compact support approaches developed in the respective type of decision systems whose objective is to minimize the redundancy existing in the current approaches in reduct computation, overcoming the ambiguities present in the attribute selection process. The choices made at developing compact support approaches were guided by the objective of developing scalable solutions. Hence *SFS* based filter selection approach using gamma measure as the attribute selection heuristic is used in the thesis.

A popular algorithm which follows *SFS* search strategy and relies on gamma measure for attribute selection heuristic is quick reduct algorithm (QRA) [18]. The simple structure of QRA enables it to be a suitable algorithm for designing scalable algorithms and reduct computation approaches for other type of decision systems like incomplete symbolic and hybrid decision systems. The relevant intrinsic aspects of QRA are elaborated in Section 3.1 of Chapter 3. The extensions and improvements of QRA are provided in the following Chapter 3.

Chapter 3

Extensions of Quick Reduct Algorithm

This chapter deals with ERCS for complete symbolic decision systems (CDS). The sequential forward search (SFS) strategies, in particular based on quick reduct algorithm (QRA) using gamma measure, will suffer when gamma gain is trivial or little. Further the monotonic aspects of SFS methods in reduct computation using gamma measure aid to reduce the effective resultant tables reasonably monotonically decreasing in size. This chapter introduces the identification of positive region for achieving effective reduction in the decision table from iteration to iteration and introduces new concepts known as trivial and non trivial ambiguities and strategies to reduce Excess Count (EC) by inteplaying with apt secondary heuristics.

Quick reduct algorithm and relevant intrinsic issues are provided in section 3.1 and the effective implementation issues of QRA in section 3.2. Limitations of QRA and strategies to overcome the same are provided in Section 3.3. The proposed Improved Quick Reduct algorithms for handling trivial and non trivial ambiguities, which addresses the limitations of QRA, and their complexity analysis is given in section 3.4. The secondary heuristics for addressing trivial and non trivial ambiguities are provided in Section 3.5. As a proof of concept, experiments are designed and results are analyzed in Section 3.6. The chapter concludes with summary in Section 3.7.

3.1 Quick Reduct Algorithm

Let $DT = (U, C \cup D, \{V_a, f_a\}_{a \in C \cup D})$ represent symbolic decision system. U is the set of objects. C is the set of symbolic conditional attributes. D is set of decision attributes. In current notation and also in general in the standard decision systems available, a decision system usually comprises of a single decision attribute and denoted as $D = \{d\}$. In case of presence of multiple decision attributes, using rough set principle of indiscernibility relation, they can be treated as a single decision attribute. V_a is the domain set of attribute $a \in C \cup \{d\}$ and f_a is a mapping from U to V_a associating a value for each object under attribute a .

Quick Reduct algorithm (QRA) is a reduct algorithm for CDS proposed by A. Chouchoulas and Qiang Shen [18]. QRA uses hill climbing based greedy approach

for obtaining reduct. QRA follows SFS strategy. For completeness QRA given in [18] is reproduced here.

Algorithm 1: Quick Reduct (QRA)

Input : Conditional Attributes C
Decision Attributes D
Output: R Super Reduct

```

1  $R = \phi$ 
2 repeat
3    $T = R$ 
4   foreach  $x \in (C - R)$  do
5     if  $\gamma_{R \cup \{x\}}(D) > \gamma_T(D)$  then
6        $T = R \cup \{x\}$ 
7     end
8   end
9    $R = T$ 
10 until  $\gamma_R(D) = \gamma_C(D)$ 
11 Return  $R$ 

```

In algorithm QRA, R represents the super reduct being returned by QRA. Using sequential forward selection strategy, R is initially set to empty set. In each iteration a search is performed to select the attribute from set $C - R$, which increases the gamma measure maximally. The set $C - R$ is the set of available attributes from which an attribute can be chosen to be included into R in an iteration of QRA. For each $a \in C - R$, the value $\gamma_{R \cup \{a\}} - \gamma_R$ is named as “gamma gain” obtained by inclusion of a into R . Hence in an iteration of QRA, an attribute obtaining maximum gamma gain is included into R . This procedure is repeated till the end condition of $\gamma_R(D) = \gamma_C(D)$ is reached.

QRA performs local search in each iteration and selects the next best attribute using greedy strategy. The end condition of QRA assures that the recommended reduct satisfies the first property of reduct definition namely $\gamma_R(D) = \gamma_C(D)$. But the second condition of reduct definition i.e., without containing any dispensable attributes, cannot be assured by QRA and also by any SFS strategy based reduct algorithm [167]. Two irrelevant attributes when combined together can result in becoming relevant to decision attribute [37]. The impact of multivariate nature of a group of attributes cannot be dynamically computed in the SFS based algorithms. The higher order relevance, a group of attributes obtain, can result in some of the already selected attributes in QRA being dispensable. Hence QRA always results in a super set of reduct called as super reduct and the number attributes which are dispensable in super reduct tells about the amount of redundancy in the recommended reduct. Number of redundant attributes present in the recommended super reduct R here after is referred as Excess Count (EC).

The next section describes the efficient implementation of QRA operations and a new version of QRA is given, reflecting the efficient implementation.

3.2 Efficient implementation of QRA

Efficient reduct computation in CDS has been an important aspect of research in last two decades [17, 42, 111, 136, 168, 177]. The objective of the section is to embed efficient implementation of operations involved in QRA algorithm. The major operation involved in QRA is computation of gamma measure γ_B for some $B \subseteq C$. In the worst case this operation of computing gamma measure needs to be done for $O(|C|^2)$ times. Hence performance enhancement of QRA depends on efficient implementation of computing gamma measure. Gamma measure computation using Eq. (2.7) essentially requires computation of positive region. Positive region computation using Eq. (2.8) requires computation of lower approximation. Computation of lower approximation using Eq. (2.1) needs computation of approximation space (U, B) . Hence the primary operations, requiring efficient implementation for enhancement of performance, in QRA are computation of approximation space and positive region. The literature review of efficient reduct computation resulted in identifying granular refinement of approximation space from [42] and efficient implementation of positive region computation from [136] to be incorporated into QRA. The formal description of these implementation level improvements of QRA is given below.

3.2.1 Granular space

The approximation space induced by rough set approach is (U, IND) . It consists of equivalence classes induced by indiscernibility relation IND . For $B \subseteq C$ the approximation space induced on set of objects U is called as granular space, $Gr(U, IND(B))$ and for convenience abbreviated as $Gr(U, B)$. Hence $Gr(U, B)$ consists of partition of U formed by IND , i.e., equivalence classes $U/IND(B)$. From the definition of equivalence relation IND the following properties are satisfied,

1. For any granule $g \in Gr(U, B), g \subseteq U$
2. For any two distinct granules $g, g' \in Gr(U, B), g \cap g' = \emptyset$
3. $\bigcup_{g \in Gr(U, B)} g = U$

The computational complexity of QRA in an iteration depends on computation of $\gamma_{R \cup \{a\}}$ for every $a \in C - R$. The major computation in $\gamma_{R \cup \{a\}}$ is in computing $POS_{R \cup \{a\}}(D)$. Computation of $Gr(U, R \cup \{a\})$ is required for finding $POS_{R \cup \{a\}}(D)$. Hence efficient implementation of QRA requires efficient way of arriving at $Gr(U, R \cup \{a\})$. Next sub section describes granular refinement approach for finding $Gr(U, R \cup \{a\})$ from the granular space $Gr(U, R)$.

3.2.2 Granular Refinement approach

QRA starts with R as an empty set. In the first iteration, for each of conditional attributes, gamma measure is computed which depends on positive region computation. Hence for each attribute its corresponding granular space is formed. The formation of granular space with a single attribute is done using sorting. Objects are sorted based on the attribute values and the place of transition from one value of the attribute to next attribute are identified resulting in formation of equivalence classes, i.e., granular space. Two efficient sorting algorithms used are quick sort [18], radix sort [177]. Using Quick sort algorithm the complexity of forming $Gr(U, \{a\})$, for any attribute $a \in C$, becomes $O(|U| \log |U|)$ and by using radix sort the time complexity is $O(|U|)$.

Now consider the case of subsequent iterations of QRA. Here, R is not empty. The granular space needs to be formed with $R \cup \{a\}$, for all $a \in C - R$. One can obtain the required $Gr(U, R \cup \{a\})$ using sorting of objects based on object values of $R \cup \{a\}$. But that process becomes blind towards already obtained granular space $Gr(U, R)$. The following property called as granular refinement property enables computation of $Gr(U, R \cup \{a\})$ from $Gr(U, R)$.

Granular Refinement Property: Let $A \subseteq B \subseteq C$ where C is the set of conditional attributes. $Gr(U, B)$ is a refinement over $Gr(U, A)$ denoted by $Gr(U, B) \preceq Gr(U, A)$, satisfying the following property:

$$g \in Gr(U, B) \Rightarrow \exists g' \in Gr(U, A) \wedge g \subseteq g' \quad (3.1)$$

Granular Refinement property is a direct consequence of IND being equivalence relation. As $Gr(U, R \cup \{a\})$ is a refinement over $Gr(U, R)$, the formation of $Gr(U, R \cup \{a\})$ can be accomplished by splitting the granules of $Gr(U, R)$ using the attribute values of a . In other words, the sorting operation is not required to be done on U but objects in each granule $g \in Gr(U, R)$ are sorted to obtain the required granules of $Gr(U, R \cup \{a\})$. Let $Gr(U, R)$ consists of granules $\{g_1, \dots, g_n\}$. Then using quick sort for forming granules the time complexity becomes $\sum_{i=1}^n O(|g_i| \log |g_i|)$. This process of arriving at $Gr(U, R \cup \{a\})$ using granular refinement has computational gains [42] because of the following property:

$$\sum_{i=1}^n O(|g_i| \log |g_i|) \leq O(|U| \log |U|) \quad (3.2)$$

Similarly computational gains are obtained using Radix sort approach for forming granular space. This procedure can significantly simplify the space complexity of QRA as at any instance of algorithm only values of a single attribute need to be available. Hence the formation of using granular space has significant space

and time complexity gains. Hence the principle of granular refinement is used in the implementation for all subsequent iterations of QRA after first iteration.

3.2.3 Positive Region Computation

Using the definition of $POS_B(\{d\})$ for any $B \subseteq C$ in Eq. (2.8), the computations needed require $n \times m$ number of sub set operations between the granules of $Gr(U, B)$ and $Gr(U, \{d\})$ where n is the number of granules in $Gr(U, B)$ and m is the number of granules in $Gr(U, \{d\})$. Instead the following approach from [136] reduces the required computations. Here for each granule $g \in Gr(U, B)$, a linear search is performed on decision attribute values of g until the decision classes in the objects of g is more than 1. In case more than ‘1’ decision classes are found then g is not included in the positive region and the linear search is stopped. Otherwise g is included in the positive region. Hence the total computational complexity of finding positive region becomes $O(|U|)$. The definition of $POS_B(\{d\})$, using this procedure, becomes,

$$POS_B(\{d\}) = \bigcup_{g \in Gr(U, B) \wedge |g/\{d\}|=1} g \quad (3.3)$$

where $|g/\{d\}|$ denote number of decision equivalence classes (granules) in the granular space $(g, \{d\})$.

3.2.4 Efficient QRA

The above improvements make QRA computationally efficient. QRA algorithm rewritten using efficient computation of approximation space using granular refinement and positive region using Eq. (3.3) is given in Algorithm 2.

Algorithm 2: Quick Reduct

Input : $DT = (U, C \cup \{d\}, \{V_a, f_a\}_{a \in C \cup \{d\}})$: Symbolic Decision Table

Output: R : Super Reduct

1 **Compute** $Gr(U, C)$, $POS_C(\{d\})$, $\gamma_C(\{d\})$

2 $R = \emptyset$

3 $Gr(U, R) = \{U\}$

4 $\gamma_R(\{d\}) = 0$

5 **while** $\gamma_R(\{d\}) < \gamma_C(\{d\})$, **do**

6 $T = R$

7 $Gr(U, T) = Gr(U, R)$

8 **foreach** $a \in C - R$ **do**

9 **Compute** $Gr(U, R \cup \{a\})$ /* Using granular refinement. */

10 **Compute** $POS_{R \cup \{a\}}(\{d\})$, $\gamma_{R \cup \{a\}}(\{d\})$

11 **if** $\gamma_{R \cup \{a\}}(\{d\}) > \gamma_T(\{d\})$, **then**

12 $T = R \cup \{a\}$

13 $Gr(U, T) = Gr(U, R \cup \{a\})$

14 **end**

15 **end**

16 $R = T$, $Gr(U, R) = Gr(U, T)$

17 **end**

18 **Return** R

The computation of $Gr(U, C)$ can be done using global and local approach. In global approach, objects are sorted using values from all C attributes and the granular space is obtained. In local approach, granular space is obtained with first attribute of C . It is refined by adding in every iteration the next attribute in C . In experimental analysis it is found that global approach is computationally effective than local approach. But local approach is recommended in case loading entire decision system into memory is not feasible.

3.3 Strategies for overcoming limitations of QRA

Incorporating granular refinement of approximation space and efficient positive region computation has reduced redundant computations in QRA. Further analysis is done, partly based on experimental analysis, to fulfill the objective of embedding compact support approaches for QRA. Three important factors are perceived to cause still redundancy in QRA and are explained below.

3.3.1 Positive Region Removal in QRA

In each iteration of QRA, the gamma gain computation depends on the number of tuples of decision table and is constant. From classical rough set theory, the set of objects which falls into positive region at an iteration will be persistently in positive region thereafter. Hence decision making in sub sequent iterations is independent of these objects. Removal of this positive region will effectively reduce the computational time from iteration to iteration. Accounting this aspect, Quick

Reduct-IV has been proposed by the inventors of QRA [17]. This aspect of positive region removal is also reported in [141, 168, 174].

Granular refinement approach will address the removal of positive region effectively by tagging granules as positive region granule and non positive region granule. Hence the granulization which filters out the positive region granules has been incorporated in the proposed improvement.

3.3.2 Order Variance in QRA

In experiments that were conducted using QRA it was noticed that QRA is order variant. Though the set of conditional attributes is invariant with respect to order, however any implementation considers them as an array which imposes an order. An algorithm developed for such environment should conceptually uphold the set property. It is observed that the existing QRA exhibits variations in with respect to the order of conditional attributes as an array and in turn variations in Excess Count. This observation indicates that QRA is not invariant with respect to the order of conditional attributes.

Further this violation is noticed because of ambiguities in decision making at an iteration of QRA. Secondary heuristics need to be designed to overcome this limitation.

3.3.3 Trivial Ambiguous situations in QRA

In an iteration of QRA, an attribute which increases gamma in the maximum way is included into R . As QRA follows greedy hill climbing like search algorithm, plateaus like in hill climbing are possible and occur not so infrequently in QRA. A plateau [156] results when the search space is flat that the value returned by target function is same as the value returned by the neighborhood regions and the direction in which to move further is not decidable.

Plateau like situation occur in QRA when gamma gain with any available attribute in an iteration of QRA is “zero”. These situations are named as “trivial ambiguous situations”, i.e., $\forall a \in C - R$ it is the case that $(\gamma_{R \cup \{a\}} - \gamma_R) = 0$. QRA does not have any procedure to select an attribute in presence of trivial ambiguous situations. This ambiguity resulted in no reduct for some of the decision systems reported by [69], in contrary for the fact of existence of reduct for decision system. To overcome this consequence one can consider to include an attribute by default selection into R . Empirical study reveals that default selection conditions exhibit significant Excess Count. It is more probable that occurrence of order variance in QRA is because of trivial ambiguous situations.

In dealing with trivial ambiguous situations, secondary heuristics are proposed for selection of next attribute into R . Hence QRA uses primary heuristic of gamma measure for selection of attributes. In case of trivial ambiguous situations secondary heuristics are employed with the intention of reducing EC and achieving order invariance in QRA. Secondary heuristics are proposed in Section 3.5.

3.3.4 Non Trivial Ambiguity Situations in QRA

Non trivial ambiguous situation refers to the occurrence of primary heuristic (gamma measure) in QRA being positive for some available attribute but the obtained gamma gain is so less to be almost equal to considering as trivial ambiguity. In such situations even though trivial ambiguity does not arise, the information to decide on the next attribute is based on very small value not to be relied upon and may lead to higher length reducts. Such situations are referred as non trivial ambiguity connoting ambiguity present non trivially.

An α -non trivial ambiguous situation occurs in an iteration of QRA when,

$$\max_{a \in C-R} (\gamma_{R \cup \{a\}} - \gamma_R) \leq \alpha \quad (3.4)$$

Secondary heuristic proposed for dealing with α - non trivial ambiguous situation is described in Section 3.5.2.

3.4 Improved Algorithms

In this section algorithms, as improvements to QRA, embedding positive region removal and ambiguity resolution aspects are presented.

3.4.1 Improved Quick Reduct Algorithm (IQRA)

Proposed algorithm IQRA incorporates the aspects of positive region removal and handling trivial ambiguous situations using secondary heuristics. IQRA is given in Algorithm 3.

Algorithm 3: Improved Quick Reduct (IQRA)

Input : Symbolic Decision Table $DT = (U, C \cup \{d\}, \{V_a, f_a\}_{a \in C \cup \{d\}})$
 SH Secondary Heuristic for Trivial Ambiguous Situations
Output: R Super Reduct

```
1 Compute  $Gr(U, C)$ ,  $POSC_C(\{d\})$ ,  $\gamma_C(\{d\})$ 
2  $R = \emptyset$ 
3  $Gr(U, R) = \{U\}$ 
4  $\gamma_R(\{d\}) = 0$ 
5  $POSCOUNT = 0$  while  $\gamma_R(\{d\}) < \gamma_C(\{d\})$  do
6    $T = R$ 
7    $Gr(U, T) = Gr(U, R)$ 
8   foreach  $a \in C - R$  do
9     Compute  $Gr(U, R \cup \{a\})$  /* Using granular refinement. */
10    Compute  $POS_{R \cup \{a\}}(\{d\})$ ,  $\gamma_{R \cup \{a\}}(\{d\})$ 
11    if  $\gamma_{R \cup \{a\}}(\{d\}) > \gamma_T(\{d\})$  then
12       $T = R \cup \{a\}$ 
13       $Gr(U, T) = Gr(U, R \cup \{a\})$ 
14    end
15  end
16  if  $\gamma_T(\{d\}) - \gamma_R(\{d\}) = 0$ , then
17    /* Indicating the occurrence of trivial ambiguous situation */
18    Use secondary heuristic  $SH$  on attributes in  $C - R$  for selection of next best attribute  $s$ 
19     $R = R \cup \{s\}$ 
20    Compute  $Gr(U, R)$  using granular refinement.
21  end
22  else
23     $R = T$ ,  $Gr(U, R) = Gr(U, T)$ 
24    Compute  $POS_R(\{d\})$ 
25    and let POSGR represent the granules of  $Gr(U, R)$  belonging to positive region.
26     $POSCOUNT = POSCOUNT + |POS_R(\{d\})|$ 
27    Remove POSGR from  $Gr(U, R)$  /* Positive region removal */
28     $\gamma_R(\{d\}) = \frac{POSCOUNT}{|U|}$ 
29  end
30 end
31 Return  $R$ 
```

In Algorithm 3, handling trivial ambiguity situation using secondary heuristic is described in lines 16 to 20. If trivial ambiguity does not arise indicate that a new attribute is included into R giving maximum gamma gain in the current iteration. In which case, the current system needs to be reduced by removal of positive region granules. This phenomenon is described in lines 21 to 27.

3.4.1.1 Complexity Analysis of IQRA

The space complexity of IQRA is $O(|U| |C|)$ required for storing decision table and granular space. The space complexity of IQRA algorithm can be further optimized to $O(|U|)$, in case entire decision table is not loadable into memory, using granular refinement included in IQRA i.e., at any point of time of an execution algorithm need to have access to object values corresponding to a single attribute and current granular space of recommended reduct attributes.

In implementation of algorithms in this chapter and throughout the thesis,

stable quick sort algorithm is used instead of radix sort. The choice was arrived based on empirical analysis. The relevance of quick sort for sorting two dimensional tables is explored in [43, 149]. Hence time complexity analysis presume usage of quick sort in computing approximation space.

Let U_i denote the non positive region objects at the beginning of i^{th} iteration of IQRA and let R_i denote the attributes included into recommended reduct after i^{th} iteration, i.e., $U_1 = U$ and $U_i = U - POS_{R_{i-1}}(\{d\})$ for $i > 1$. As computations in IQRA are performed only on non positive region objects, the time complexity of i^{th} iteration of IQRA is, $O((|C| - i + 1) |U_i| \log |U_i|)$. Hence as iterations progress in IQRA, the iteration complexity decreases monotonically. If size of the reduct given by IQRA is n then the total time complexity of IQRA expressed in terms of summation of iteration wise time complexity is $\sum_{i=1}^n O((|C| - i + 1) |U_i| \log |U_i|) < O(|C|^2 |U| \log |U|)$.

3.4.2 Improved Quick Reduct Algorithm for Non Trivial Ambiguity (IQRA_NonTrivial)

Improved Quick Reduct Algorithm for Non Trivial Ambiguity (IQRA_NonTrivial) incorporates handling non trivial ambiguity situations using secondary heuristic. The parameter α defining the extent for non trivial ambiguity is an additional input parameter for IQRA_NonTrivial algorithm. Algorithm IQRA_NonTrivial is given in Algorithm 4.

Algorithm 4: Improved Quick Reduct for Non Trivial Ambiguity (IQRA_NonTrivial)

Input : Symbolic Decision Table $DT = (U, C \cup \{d\}, \{V_a, f_a\}_{a \in C \cup \{d\}})$
 SH Secondary Heuristic for Trivial Ambiguous Situations
Output: R Super Reduct

```
1 Compute  $Gr(U, C)$ ,  $POSC_C(\{d\})$ ,  $\gamma_C(\{d\})$ 
2  $R = \emptyset$ 
3  $Gr(U, R) = \{U\}$ 
4  $\gamma_R(\{d\}) = 0$ 
5  $POSCOUNT = 0$  while  $\gamma_R(\{d\}) < \gamma_C(\{d\})$  do
6    $T = R$ 
7    $Gr(U, T) = Gr(U, R)$ 
8   foreach  $a \in C - R$  do
9     Compute  $Gr(U, R \cup \{a\})$  /* Using granular refinement. */
10    Compute  $POS_{R \cup \{a\}}(\{d\})$ ,  $\gamma_{R \cup \{a\}}(\{d\})$ 
11    if  $\gamma_{R \cup \{a\}}(\{d\}) > \gamma_T(\{d\})$  then
12       $T = R \cup \{a\}$ 
13       $Gr(U, T) = Gr(U, R \cup \{a\})$ 
14    end
15  end
16  if  $\gamma_T(\{d\}) - \gamma_R(\{d\}) \leq \alpha$ , then
17    /* Indicating the occurrence of non trivial ambiguous situation */
18    Use secondary heuristic  $SH$  on attributes in  $C - R$  for selection of next best attribute  $s$ 
19     $T = R \cup \{s\}$ 
20    Compute  $Gr(U, T)$  using granular refinement.
21  end
22  if  $\gamma_T(\{d\}) - \gamma_R(\{d\}) > 0$ , then
23     $R = T$ ,  $Gr(U, R) = Gr(U, T)$ 
24    Compute  $POS_R(\{d\})$ 
25    and let POSGR represent the granules of  $Gr(U, R)$  belonging to positive region.
26     $POSCOUNT = POSCOUNT + |POS_R(\{d\})|$ 
27    Remove POSGR from  $Gr(U, R)$  /* Positive region removal */
28     $\gamma_R(\{d\}) = \frac{POSCOUNT}{|U|}$ 
29  end
30 end
31 Return  $R$ 
```

In Algorithm 4, lines 16 to 20 deal with handling non trivial ambiguity situation. In Algorithm IQRA, positive region removal occurs in case trivial ambiguity does not arise. In IQRA_NonTrivial lines 21 to 27 deal with positive region removal. The condition for positive region removal becomes non occurrence of non trivial ambiguity (positive gamma gain $> \alpha$) or in case of non trivial ambiguity, occurrence of positive gamma gain with the attribute selected using secondary heuristic. Both these cases are considered by the condition given in line 21.

Theoretical time and space complexities of IQRA_NonTrivial is same as that of IQRA but incurs more number of secondary heuristic computations compared to IQRA.

3.5 Heuristics for Ambiguous Situations

This section describes the secondary heuristics used for dealing with ambiguity situations.

3.5.1 Heuristics for Trivial Ambiguous Situations

The following heuristics are used for dealing with trivial ambiguity situation.

3.5.1.1 VPRS Heuristic

Variable precision rough sets [6, 175] is an extension of rough sets dealing with imprecision. Trivial ambiguity situation has arisen out of zero gamma gain with inclusion of any of the available attributes into recommended reduct R . Even though there is no gamma gain all the attributes need not be considered same with respect to obtaining gamma gain in future iterations. There may be some attributes which are near by to achieve gamma gain and VPRS heuristic enables a way for quantifying that nearness. The basics of VPRS are described in section 2.2.1 of Chapter 2.

Another way to look at trivial ambiguity situation is zero gamma gain at $\beta = 1$. Finding beta value at which an attribute's inclusion into R results in gamma gain in VPRS, that beta value indicates how near an attribute is in obtaining gamma gain. As beta value is nearer to '1' there is more chance that in the next iteration of IQRA, because of granular refinement, existence of gamma gain with $\beta = 1$. The beta value at which there is positive gamma gain is named as "*positive beta*". The attribute which has the maximum "*positive beta*" is chosen as the attribute to be included in the current iteration of IQRA. VPRS computations are used only for selection of attribute, for inclusion into R , in trivial ambiguity situations but not to calculate the positive region and gamma measure using variable precision of less than 1. This is already reflected in IQRA as there is no positive region computation and removal in case of trivial ambiguous situations in IQRA.

Based on theory of VPRS, a secondary heuristic, namely "*VPRS heuristic*" is proposed as a measure of nearness to obtain positive region in further iterations as follows. For an attribute $a \in C - R$, the "*positive beta*" $pos\beta_a$ is ,

$$pos\beta_a = \max_{A \in Gr(R \cup \{a\})} \left(\max_{X \in U/IND(\{d\})} \left(\frac{|A \cap X|}{|A|} \right) \right) \quad (3.5)$$

Similar to efficient positive region computation reported in section 3.2.3, there is no need for computing intersection of A with all decision equivalence classes. Instead projection of decision equivalence classes on A can be obtained by computing $A/IND(\{d\})$. Hence Eq. (3.5) can be equivalently rewritten as,

$$pos\beta_a = \max_{A \in Gr(R \cup \{a\})} \left(\max_{X \in A/IND(\{d\})} \left(\frac{|X|}{|A|} \right) \right) \quad (3.6)$$

The attribute $s \in C - R$ included into R in trivial ambiguous situations, where,

$$pos\beta_s = \max_{a \in C-R} pos\beta_a \quad (3.7)$$

3.5.1.2 Information Gain Heuristic

Information Entropy is proposed by Shannon [121] in the seminal paper “A Mathematical Theory of Communication”. Entropy is associated with the uncertainty associated with the random variable and tells the average number of bits needed for information transmission over a channel. Entropy is zero when there is no uncertainty associated with the outcome of the random variable.

Information entropy is applied in several areas like compression, machine learning [113, 115]. In Machine learning, it is widely applied in decision trees [113]. In the context of rough sets, conditional information entropy as a heuristic measure is used in several reduct algorithms [54, 111, 125, 147]. The formulas for Information gain (IG) in the context of approximation space specified as granular space is described here.

The information entropy of decision attribute $\{d\}$ on objects U is defined as,

$$H(U, \{d\}) = - \sum_{g \in Gr(U, \{d\})} p(g) \log(p(g)) \quad (3.8)$$

where for any $X \subseteq U$, $p(X) = \frac{|X|}{|U|}$.

The conditional information entropy of $\{d\}$ given an attribute set $B \subseteq C$ is defined as,

$$H(U, \{d\}/B) = - \sum_{g \in Gr(U, B)} p(g) * H(g, \{d\}) \quad (3.9)$$

The information gain of B with respect to the decision attribute $\{d\}$ is defined as the reduction in the entropy (uncertainty) of $\{d\}$ under the knowledge of B and is specified as,

$$IG(U, \{d\}, B) = H(U, \{d\}) - H(U, \{d\}/B) \quad (3.10)$$

There is no difference in attribute selection criterion between minimizing of conditional entropy or maximizing of *IG heuristic*. *IG heuristic* is used as secondary heuristic, instead of conditional information entropy, only because *IG heuristic* is monotonically increasing similar to gamma measure. In trivial am-

biguous situation the attribute included into R is determined by selecting that attribute s whose inclusion into R achieves maximum information gain, i.e.,

$$IG(U, \{d\}, R \cup \{s\}) = \max_{a \in C-R} IG(U, \{d\}, R \cup \{a\}) \quad (3.11)$$

3.5.1.3 Domain Cardinality Heuristic

Secondary heuristic computations add an overhead for computational cost of QRA. There can be situations where spending extra computational time for secondary heuristic may be unwarranted. In such situations instead of going for default attribute, a simple heuristic is required. One such heuristic is maximum domain cardinality. An attribute is selected into R if it has more domain cardinality. The simple idea behind this approach is presence of more domain values may lead to shorter granules being formed and hence more chance of getting into positive region in further iterations of QRA. Hence the attribute selected is s where,

$$|V_s| = \max_{a \in C-R} |V_a| \quad (3.12)$$

In the above, the available distinct values of an attribute in the current decision system in view of positive region removal aspect, is incorporated in IQRA. It is not applicable in binary decision systems or in decision systems where domain cardinality of all attributes is same. In such cases, domain cardinality heuristic is equal to blind or default selection in QRA.

3.5.1.4 Ensemble of Heuristics

Experimental analysis indicated that there are instances where one secondary heuristic performs better than others. Hence it may be possible that attribute which is good with respect to all the three secondary heuristics may perform better. In this method, rank order of attributes resulting from each heuristic is combined for determining the choice of attribute. But evaluating three secondary heuristics for every occurrence of trivial ambiguity situations is computationally costly. In case it is affordable the ensemble of secondary heuristics can be employed.

IQRA using *VPRS secondary heuristic* in trivial ambiguous situations is named as IQRA_VPRS, using *IG heuristic* is named as IQRA_IG, using *maximum domain cardinality heuristic* is named as IQRA_MD and the combination of heuristics when employed is called as IQRA_Ensemble.

3.5.2 Heuristic for Non Trivial Ambiguous Situations

Based on empirical analysis described in Section 3.6, both VPRS, IG heuristics have given better performance in dealing with trivial ambiguity situations. The maximum $pos\beta$ is 1 in case of strict non trivial ambiguity situation ($0 < (\gamma_T - \gamma_R) \leq \alpha$). Hence VPRS heuristic will be selecting the same attribute selected with positive gamma gain (lines 8 to 15 in Algorithm IQRA_NonTrivial) in case of strict non trivial ambiguity situation. Hence it is recommended to consider only IG heuristic for handling non trivial ambiguity situations.

Algorithm IQRAIG_NonTrivial is formulated, in which, under non trivial ambiguous situations secondary heuristic IG is used for choosing the attribute.

3.6 Experiments and Results

The experiments in all chapters are conducted in Intel(R) Core(TM) 2 Quad CPU @2.83 GHz with 3.8 GB RAM under Fedora 10 (Linux) operating system in Matlab environment. Experiments are divided into three subsections. Subsection 3.6.1 deals with experiments conducted on CDS having trivial ambiguity. Subsection 3.6.2 deals with experiments conducted on non trivial ambiguous CDS. Subsection 3.6.3 contains comparative study of IQRA_IG algorithm with reduct computation algorithms from literature.

3.6.1 Experiments with trivial ambiguous datasets

Table 3.1 contains the details of complete decision systems from UCI Machine Learning Repository [11] and Richard Jensen's repository of datasets [53]. Table 3.1 contains details about the size of the decision system in terms of number of objects, number of conditional attributes and number of decision classes. Some of the datasets containing numeric attributes are discretized for converting them to symbolic decision systems. Such dataset names are suffixed with the discretization method used. Applying QRA on these datasets the presence of trivial or non trivial ambiguous situations is identified and reported in Table 3.1. In this section, summary of results are obtained for datasets in Table 3.1 having trivial ambiguity.

The objective of experiments is for comparative analysis of performance of proposed algorithms with Quick Reduct algorithm in terms of length of the super reduct, excess count (EC) and computational time (CT). Algorithms QRA, IQRA_VPRS, IQRA_IG, IQRA_MD, IQRA_ENSEMBLE are applied on each trivial ambiguous dataset and results are summarized in Tables 3.2, 3.3¹. Each

¹Prominent results are highlighted in tables with bold face.

Table 3.1: Details of Symbolic Decision Systems containing Trivial or Non Trivial Ambiguities

Name	Objects	$ C $	$ V_d $	Trivial Ambiguity	Nontrivial Ambiguity
MofN	1000	13	2	Yes	No
Exactly2	1000	13	2	Yes	No
DNA	318	57	8	Yes	No
Exactly	1000	13	2	Yes	No
Chess	3196	36	2	Yes	Yes
Heart	294	13	2	No	Yes
Letters	26	33	26	Yes	Yes
Water (EqualWidth2)	521	38	3	No	Yes
Water(EqualWidth3)	521	38	3	No	Yes
Water (ChiMerge)	521	38	3	No	Yes
Isolet (ChiMerge)	34	617	17	No	Yes
Isolet (EqualWidth2)	34	617	17	No	Yes

Table 3.2: Reduct Lengths and Excess Count for Trivial Ambiguous Decision Systems

Dataset	Reduct Length (Reduct Type) Excess Count				
	QRA	IQRA_VPRS	IQRA_IG	IQRA_MD	IQRA_ENSEMBLE
MofN	7(SR)(1)	6(R)	6(R)	7(SR)(1)	6(R)
Exactly2	10(R)	11(SR)(1)	11(SR)(1)	10(R)	11(SR)(1)
DNA	7(R)	6(R)	5(R)	5(R)	6(R)
Exactly	8(SR)(2)	7(SR)(1)	8(SR)(2)	8(SR)(2)	7(SR)(1)
Chess	31(SR)(2)	31(SR)(2)	31(SR)(2)	31(SR)(2)	31(SR)(2)
Letters	11(SR)(3)	11(SR)(3)	10(SR)(2)	11(SR)(3)	11(SR)(3)

entry of Table 3.2 contains length of the super reduct obtained along with type of reduct (SR: Super Reduct, R: Reduct) and in case of the result being SR, the value of EC is reported. Table 3.3 reports the computational time incurred for computing the reduct on trivial ambiguous datasets in seconds.

In order to understand the effect of varying the order of the conditional attributes on the resulting super reduct, Order Variance experiment is conducted. In this experiment, on each dataset, application of QRA and proposed algorithms is repeated for N trails. In each trail, the order of conditional attributes is randomly permuted before application of algorithms. Taking number of trails $N = 100$, the

Table 3.3: Computational Time in Seconds for Trivial Ambiguous Decision Systems

Dataset	QRA	IQRA_VPRS	IQRA_IG	IQRA_MD	IQRA_ENSEMBLE
MofN	0.291	0.168	0.134	0.166	0.154
Exactly2	0.766	0.635	0.687	0.496	0.684
Exactly	0.351	0.275	0.448	0.323	0.352
Chess	32.7	1.135	1.124	1.086	1.136
Letters	0.316	0.160	0.156	0.147	0.173

Table 3.4: Order Variance Experiment Results for Trivial Ambiguous Datasets

Dataset	Reduct Lengths (Distinct Reduct Count)				
	QRA	IQRA_VPRS	IQRA_IG	IQRA_MD	IQRA_ENSEMBLE
MofN	6,7,8,9,10,11 (43)	6 (1)	6 (1)	6,7,8,9,10,11 (43)	7 (1)
Exactly2	10,11,12,13 (8)	11 (1)	11 (1)	10,11,12,13 (8)	10 (1)
Exactly	6,7,8,9,10,11,12 (55)	7 (1)	8 (1)	6,7,8,9,10,11,12 (55)	9 (1)
Chess	31,32 (12)	31 (4)	31 (4)	31 (4)	31 (4)
Letters	8,9,10,11,12,13,14,15 (100)	9,10,11,12,13 (98)	9,10,11,12 (86)	8,9,10,11,12,13,14,15 (100)	10,11,12,13,14 (100)

Table 3.5: Reduct Length and Excess Count Summary for Non Trivial Ambiguous Datasets

Dataset	QRA	IQRAIG_NonTrivial
Chess	31(SR)(2)	29(R)
Heart	7(R)	7(R)
Letters	11(SR)(3)	10(SR)(1)
Water (EqualWidth2)	29(SR)(6)	23(R)
Water(EqualWidth3)	17(SR)(3)	18(SR)(3)
Water (ChiMerge)	6(R)	5(R)
Isolet (ChiMerge)	3(R)	2(R)
Isolet (EqualWidth2)	12(SR)(1)	8(R)

summary of the results obtained are reported in Table 3.4. In each cell of Table 3.4 reports the lengths of the super reducts along with number of distinct reducts obtained. For example in Table 3.4, for MofN dataset, out of 100 repeated trails QRA resulted in super reducts consisting of 6,7,8,9,10,11 attributes and there are 43 different super reducts (with different attribute combination) present in the 100 super reducts obtained.

3.6.2 Experiments with non trivial ambiguous datasets

Similar experiments, as conducted for trivial ambiguity datasets, are conducted on datasets reported in Table 3.1 having non trivial ambiguity. Parameter α is taken at 0.1. Choosing a very small α value will result in occurrence of very few non trivial ambiguous situations and more or less resemble trivial ambiguous experiments. A higher α will result in increased evaluation of secondary heuristic computations. Hence α is taken as 0.1. The results for reduct length, reduct type and excess count for non trivial ambiguous datasets are given in Table 3.5 and computational time results are reported in Table 3.6. Order variance experiment results are summarized in Table 3.7.

3.6.3 Comparative Experiments with Existing Methods

IQRA_IG algorithm is taken as the representation algorithm of the proposed algorithms for performing comparative analysis with some of the existing reduct

Table 3.6: Computational Time for Non Trivial Ambiguous Datasets

Dataset	QRA	IQRAIG_NonTrivial
Chess	32.133	2.4826
Heart	0.55154	0.15761
Letters	0.31212	0.27251
Water (EqualWidth2)	6.3132	3.1775
Water(EqualWidth3)	7.482	2.3775
Water (ChiMerge)	3.3624	0.70755
Isolet (ChiMerge)	2.2582	2.0179
Isolet (EqualWidth2)	9.0051	2.3957

Table 3.7: Order Variance Experiment Results for Non Trivial Ambiguous Datasets

Dataset	Reduct Lengths (Distinct Reduct Count)	
	QRA	IQRAIG_NonTrivial
Chess	31,32 (13)	29 (2)
Heart	7 (2)	7 (2)
Letters	8,9,10,11,12,13,14,15 (99)	8,9,10 (85)
Water (EqualWidth2)	27,28,29,30,31,32 (44)	23 (2)
Water(EqualWidth3)	17,18,20 (15)	18 (3)
Water (ChiMerge)	6 (26)	5 (1)
Isolet (ChiMerge)	3 (70)	2,3 (49)
Isolet (EqualWidth2)	8,9,10,11,12,13 (100)	7,8,9 (98)

computation algorithms in literature. Comparative analysis is done with the results of positive approximation based reduct algorithms reported in [111] and hash based reduct computation reported in [168]. Approaches developed in [111] and in [168] incorporate positive region removal. Hence IQRA_IG algorithm becomes a comparable algorithm with these approaches.

In [111] positive region removal is named as positive approximation. The concept of positive approximation is incorporated in existing reduct computation algorithms leading to “Feature selection based on the positive approximation (FSPA)” algorithms. Positive region (PR) (gamma measure) heuristic based reduct algorithm in [48], Shannon conditional entropy (SCE) based reduct algorithm in [38], Liang conditional entropy (LCE) based reduct algorithm in [70], Combination conditional entropy (CCE) based reduct algorithm in [110] are improved with positive approximation as algorithms FSPA-PR, FSPA-SCE, FSPA-LCE, FSPA-CCE algorithms respectively. Details of datasets used in experiments reported in [111] are given in Table 3.8. In Table 3.8, dataset is suffixed with *Complete* refers to dataset is made complete after removal of rows containing missing values. Comparative results of reduct lengths obtained by FSPA based algorithms and IQRA_IG algorithm are reported in Table 3.9 and computational time (in seconds) are reported in Table 3.10.

Table 3.8: Datasets used in FSPA based algorithms

Name	Objects	$ C $	$ V_d $
Mushroom (Complete)	5644	22	2
Tic-Tac-Toe	958	9	2
Dermatology (Complete)	358	34	6
Kr-vs-Kp	3196	36	2
Breast-Cancer-Wisconsin (Complete)	683	9	2
Backup-large-test	376	35	19
Shuttle	58000	9	7
Letter-recognition	20000	16	26
TicData2000	5822	85	2

Table 3.9: Reduct Length comparison with FSPA based algorithms

Dataset	FSPA-PR	FSPA-SCE	FSPA-LCE	FSPA-CCE	IQRA_IG
Mushroom (Complete)	3	4	4	4	3(R)
Tic-Tac-Toe	8	8	8	8	8(R)
Dermatology (Complete)	10	11	10	10	7(SR)(1)
Kr-vs-Kp	29	29	29	29	31(SR)(2)
Breast-Cancer-Wisconsin (Complete)	4	4	5	4	4(R)
Backup-large-test	10	10	10	9	11(SR)(3)
Shuttle	4	4	4	4	4(R)
Letter-recognition	11	11	12	11	12(SR)(1)
TicData2000	24	24	24	24	23(R)

Table 3.10: Computational Time (in seconds) comparison with FSPA based algorithms

Dataset	FSPA-PR	FSPA-SCE	FSPA-LCE	FSPA-CCE	IQRA_IG
Mushroom (Complete)	20.4531	159.5938	294.0000	159.6406	0.38546
Tic-Tac-Toe	0.3125	3.1094	5.7813	3.1406	0.21524
Dermatology (Complete)	0.4375	1.9844	3.7500	2.2656	0.35441
Kr-vs-Kp	21.5781	105.9844	191.1250	105.7500	1.0963
Breast-Cancer-Wisconsin (Complete)	0.0938	0.8438	1.6719	0.8906	0.065656
Backup-large-test	0.4219	1.7656	3.2188	1.9844	0.28693
Shuttle	712.2500	10153.1719	20228.3906	10948.9219	15.2165
Letter-recognition	112.6250	2740.2500	5558.7813	2610.3594	11.9514
TicData2000	296.3750	1043.8906	1805.5625	1048.5781	23.4393

Table 3.11: Datasets used in Comparative analysis with THQARA algorithm

Name	Objects	$ C $	$ V_d $
Mushroom	8124	22	2
Vote	435	16	2
Tic-Tac-Toe	958	9	2
Lung-Cancer	32	56	3
German-Credit	1000	20	2
PenDigits	10992	16	10
Letter-recognition	20000	16	26

Table 3.12: Reduct lengths in comparative analysis with THQARA algorithm

Dataset	IQRA_IG
Mushroom	5(R)
Vote	10(SR)(1)
Tic-Tac-Toe	8(R)
Lung-Cancer	5(R)
German-Credit	8(R)
PenDigits	5(R)
Letter-recognition	12(SR)(1)

Twice Hash attribute reduction algorithm (THQARA) [168] uses gamma measure as the attribute selection heuristic and incorporates positive region removal. In the survey of efficient reduct computation algorithms conducted in year 2010 by Shifei Ding et al. [23], algorithm THQARA was considered as the most efficient sequential forward selection based heuristic reduct computation algorithm. This motivated to perform comparative analysis with algorithm THQARA.

Datasets used in THQARA experiments are given in Table 3.11. Reduct lengths obtained using IQRA_IG algorithm are given in Table 3.12. Table 3.12 contains reduct lengths for only IQRA_IG as details of reduct lengths are not reported in 3.12. Comparative results of computational time (in seconds) are given in Table 3.13 containing results reported for THQARA algorithm from [168]

Table 3.13: Computational Time (in seconds) in comparative analysis with THQARA algorithm

Dataset	THQARA	IQRA_IG
Mushroom	2.359	0.59605
Vote	0.182	0.11581
Tic-Tac-Toe	0.125	0.21242
Lung-Cancer	0.047	0.098181
German-Credit	0.078	0.97133
PenDigits	4.627	7.5366
Letter-recognition	21.355	11.9284

3.6.4 Analysis of Results

It is observed from the results reported for trivial ambiguous datasets in Tables 3.2,3.3 that proposed algorithms IQRA_VPRS, IQRA_IG, IQRA_ENSEMBLE, IQRA_MD have obtained either similar reduct lengths or shorter reduct lengths in comparison to QRA except for the dataset Exactly2. In MofN dataset, IQRA_IG, IQRA_VPRS and IQRA_ENSEMBLE have obtained reduct where as super reduct resulted from QRA, IQRA_MD. Unlike IQRA_ENSEMBLE, which require multiple secondary heuristic evaluations, IQRA_VPRS, IQRA_IG have obtained similar results to IQRA_ENSEMBLE by just relying on a single secondary heuristic for dealing with trivial ambiguous situations.

Computational gains by IQRA algorithms over QRA are highly significant especially for large dataset such as Chess obtaining computational gain percentage around 95%. Computational gain percentages are increasing with increasing size of dataset. Compact based approaches such as positive region removal, granular refinement, efficient positive region computation proved to be significant in reducing computational costs of QRA.

Exactly2 is a special dataset in which trivial ambiguity occurs in first four iterations of QRA. Only in fifth iteration, gamma gain is observable. Hence QRA including [1,2,3,4] attributes by default in the first four iterations results in a reduct [1,2,3,4,5,6,7,8,9,10]. It is to be noted that Exactly2 dataset has a single reduct [1,2,3,4,5,6,7,8,9,10]. As the default selected attributes are part of the final reduct, QRA apparently is resulting in the desired result. But from order variance experiment results given in Table 3.4, QRA results in 8 distinct super reducts out of 100 trails with lengths varying from 10 to 13. Using secondary heuristics in four trivial ambiguous situations that occur in Exactly2 dataset it is observed that IQRA_VPRS, IQRA_IG, IQRA_ENSEMBLE result in only a single distinct super reduct for 100 trails of size 11.

In order variance experiment results given in Table 3.4, IQRA_IG, IQRA_VPRS have resulted in the ability to arrive consistently same size super reduct in MofN, Exactly2, Exactly, Chess datasets and also same super reduct in MofN, Exactly2, Exactly datasets. Simple MD heuristic proved to be effective in Chess dataset as conditional attributes of Chess dataset have different domain cardinality. Since remaining datasets being binary datasets, results of IQRA_MD are same as that of QRA. Only in Letters dataset, varying length super reducts are obtained in IQRA algorithms still reducing the range (9 to 13) compared to QRA (8 to 15) owing to the reason that gamma gains are equally achieved by several attributes and in case of secondary heuristic evaluations also similar phenomenon is observed.

Prominently it can be concluded that trivial ambiguity resolution incorporated in IQRA based algorithms significantly reduce the variability of results due to order

of conditional attributes and can result in lesser length super reducts compared to QRA.

Similar results are observed in results for non trivial ambiguous datasets reported in Tables 3.5, 3.6, 3.7. IQRAIG_NonTrivial has obtained much shorter length super reducts compared to QRA. This is significantly observed in Water (EqualWidth2) dataset wherein IQRAIG_NonTrivial has obtained result which is shorter by 6 attributes and in Isolet(EqualWidth2) by obtaining shorter size by 4 attributes. Order variance experiment results also show a significant range reduction of super reduct lengths by IQRAIG_NonTrivial and almost overcoming order variance problem by resulting in the same super reduct or same length super reduct in all 100 trails for datasets such as Chess, Heart, Weather (EqualWidth2), Water (EqualWidth3), Water (ChiMerge).

From the results obtained for trivial and non trivial ambiguous datasets it emerges that IQRA_IG, IQRA_VPRS, IQRAIG_NonTrivial have performed better than other algorithms. In the rest of the thesis, IQRA_IG has been considered as the representative IQRA algorithm for extending in scalable approaches for reduct computation using sampling techniques and also as the basic structure to be followed in constructing reduct algorithms for incomplete and hybrid decision systems. Entropy based reduction, gamma measure based reduction are two important directions in which rough set based reduct computation algorithms have evolved. IQRA_IG gives a way of using both heuristics for optimal dividends. Even though IQRAIG_NonTrivial has given very good results, it is not taken as the representative as it is a parameter dependent approach and in extending for large decision systems too many secondary heuristic evaluations can become a bottleneck.

In comparative results with FSPA based algorithms with IQRA_IG, IQRA_IG obtained similar length super reducts or reducts but with highly significant computational gains. IQRA_IG has obtained significant gains over FSPA algorithms. On large decision systems like Mushroom, Kr-vs-Kp, Shuttle, Letter Recognition, TicData2000 gain percentages of IQRA_IG over FSPA-PR itself is in the order of 90 to 99% and over CE based algorithms it is in the order of 97 to 99.9%. Comparing FSPA-PR results with entropy based algorithms FSPA-SCE, FSPA-LCE, FSPA-CCE (refer to Table 3.10), exemplify the fact that entropy based computations are of higher complexity than gamma measure based computations. Hence IQRA_IG is an apt recommendation for large decision systems, as it judiciously uses high complex entropy computations in times of requirement and achieves performance with respect to reduct length comparable to pure entropy based approaches.

In comparative results with THQARA algorithm, reduct length comparison could not be performed as the reduct lengths are not mentioned in THQARA

results reported in [168]. THQARA obtained computational gain percentages of 40 to 90% over IQRA_IG in datasets Tic-Tac-Toe, Lung-Cancer, German-Credit, Pen-Digits. IQRA_IG obtained computational gain percentages of 35 to 74% over THQARA in datasets Mushroom, Vote, Letter-Recognition. IQRA_IG algorithm is promising over THQARA in obtaining gains in large datasets such as Letter_Recognition, Mushroom.

The comparative results empirically establish that IQRA_IG algorithm is better or comparable algorithm with some of the best reduct computational algorithms for CDS.

3.7 Summary

This chapter empirically establishes that ambiguity (trivial and non trivial) resolution is an essential requirement for SFS algorithms in avoiding redundancy (excess count) of the super reduct by considering QRA as a representative algorithm. Order variance experiments established that the accidental default choice in QRA can lead to a good or bad super reduct without any direction. Direction enabled through proper secondary heuristics has the power to overcome the order variance problem and results in shorter length super reducts irrespective of the organization of the order of the conditional attributes.

The literature survey regarding efficient implementation of reduct computation algorithms have given concepts like positive region removal, granular refinement, efficient positive region computation which enabled to make QRA into IQRA. Comparative results established that incorporated improvements make IQRA comparable or better than some of the best reduct computation approaches in complete symbolic decision systems from literature.

As IQRA_IG is a theoretical blend of two primary directions, dependency measure and information gain, for reduct computation yielded better results over several datasets. IQRA_IG is recommended over IQRAIG_NonTrivial in general, by considering implications of parameter dependency and intensity of secondary heuristic computations.

Chapter 4

Sampling based Scalable Approach for Reduct Computation in Symbolic Decision Systems

In this chapter, the issue of scalability for reduct computation for complete symbolic decision systems (CDS) is addressed. There are two problems in using the existing algorithms to deal with large decision systems. The memory constraint of the system may not allow the entire dataset to be loaded into memory at the same time. Hence the terms like small, large decision systems are relative to the system being used. A small decision system like “iris” may become a large decision system if it is to be loaded into an embedded device with meager memory specifications. The other aspect is computational time for obtaining reduct for large or very large decision systems can become very high, making feature selection only as an offline activity and not suitable to online scenarios. As feature selection is part of the preprocessing stage in any Data Mining application, it is desirable to make reduct algorithms that are adoptable to given memory and computational constraints. This chapter addresses this important issue incorporating sampling approach for reduct computation which can scale to large and very large CDS.

Section 4.1 contains the review of existing sampling based approaches for reduct computation. Section 4.2 elucidates the appropriateness of sampling based approach for reduct computation by integrating the global validation step and further discusses the importance of global validation step. Next section 4.3 discusses the structure of the proposed sampling based approach and details the importance of global validation step incorporated. Sections 4.4, 4.5 describes the proposed sampling based algorithms categorized into persistence approach and non persistent approach respectively. The details of experiments and results are given in section 4.6 followed by analysis of results in section 4.7.

4.1 Existing approaches and Limitations

The existing single reduct algorithms will fail in producing result in a reasonable time for large datasets. This issue of scalability has attracted researchers to mold the reduct finding problem as distributed or parallel [133, 138, 150]. Sample based approaches to reduct computation are proposed by [14, 44]. The first one is based on the whole dataset whereas the later approach is based on the representative part of the data. The first approach demands sophisticated infrastructure if not advanced whereas the second approach can be managed with a simple available infrastructure. This study focuses on sampling based approaches for reduct computation.

In a sampling technique, a recommended reduct is estimated using the sampled subset of objects from the given dataset. The algorithms proposed in [14, 44] use discernibility matrix based approaches for estimating reduct. The Sampling Approximate Reduct Algorithm (SARA) proposed in [44] involves two phases. In the first phase attributes are ranked depending on the frequency corresponding to their presence in entries of discernibility matrices of sample datasets. Second phase is an iterative phase. Starting with entire set of attributes as the reduct, a decision is taken whether to remove the least ranked attributes or include the last removed least ranked attribute by checking against the discernibility matrix of the sampled dataset. The algorithm Fast Scalable Reducts (FSR) given in [14] consists of efficiently estimating the Minimal Discernibility list (MDL) only based on few pair of sampled objects from the dataset. An approximation of the discernibility function for the dataset is estimated and feature subsets are generated. The important limitation of existing sampling based approaches is that they result only in approximate reducts (sub reduct) and fail to give the reduct or super reduct for a given dataset.

Before describing proposed approach for sampling based reduct computation which overcomes the limitation of existing approaches, the next section describes the relevance of sampling based approach for attribute selection.

4.2 Relevance of Sampling in the context of Rough Set based Attribute Reduction

Sample based analysis, mimicking the process based on whole population, is applicable to many scenarios. The relevance of sample based attribute reduction or feature selection using rough sets, particularly with respect to QRA like algorithm, is discussed in this section. In an iteration of QRA, an attribute is selected into reduct set based on the criteria of obtaining maximum gamma gain. As gamma

measure indicates the proportionality of positive region objects in the given Universe it is appropriate to see how positive region computations are affected in the constrained sample dataset. Let $DT = (U, C \cup D, \{V_a, f_a\}_{a \in C \cup D})$ represent the given symbolic decision system. Let $SDT = (U', C \cup D, \{V_a, f_a\}_{a \in C \cup D})$ represent the sampled dataset from DT where, $U' \subseteq U$.

Let B be a set of conditional attributes i.e., $B \subseteq C$. Using the notation given in Chapter 3, let $Gr(U, B)$, $Gr(U', B)$ represent the granular space induced by equivalence relation $IND(B)$ on DT , SDT respectively. The following properties are trivially established by using the principles of sampling and positive region computations.

Property 1: For each $g' \in Gr(U', B)$ there exists an associated unique granule $g \in Gr(U, B)$ such that $g' \subseteq g$.

The descriptions associated with granules g, g' are same in DT, SDT . From Property 1 it follows that the function association between granules of $Gr(U', B)$ and $Gr(U, B)$ is injective but usually it is not surjective. Surjective functionality is not assured because under sampling it is possible that none of the objects of a granule $g \in Gr(U, B)$ are selected in SDT i.e., $g \cap U' = \phi$. Usually chance occurrence of such lost granules under sampling is less and when occurs usually occurs with small size granules of $Gr(U, B)$. Hence the effect of such granule omission in positive region computation may not be significant. As sample size increases this chance of losing some granules decreases further. Losing granule means not having single object in SDT satisfying the granule description.

The following property follows from the definition of lower approximation and positive region.

Property 2: If $g \in Gr(U, B)$ is in positive region, i.e., $g \subseteq POS_B(\{d\})$ then the $g' \in Gr(U', B)$ satisfying $g' \subseteq g$ if exists is in positive region, i.e., $g' \subseteq POS_B(\{d\})$ in SDT .

But the converse of property 2 is not true which is given below.

Property 3: $g' \in Gr(U', B)$ in positive region, i.e., $g' \subseteq POS_B(\{d\})$ does not imply that the associated $g \in Gr(U, B)$ is in positive region.

False POS granules are defined as the granules in sample dataset whose belongingness in positive region does not correspond to the positive region belongingness of the associated granules in the original dataset. Occurrence of *false POS* granules leads to attribute significance to attribute set B in SDT even though there is no corresponding significance in DT . Hence the sampling design has to be with the

objective of overcoming *false POS* granules. Occurrence of *false POS* granule is influenced by distribution of decision classes in the associated granule. For example if an associated granule contains 99 objects belonging to a single decision class but 1 object belonging to other decision class then chance for occurrence of *false POS* granule in sampled dataset is high. But in such cases the false positive granule in the current iteration by granular refinement can become a true positive granule in future iterations. This limitation is well addressed in successive iterations.

On the other side, determination of non belongingness to positive region, if established in sampled granule, corresponds to the same in associated granule in *DT*. Hence,

Property 4: If $g' \in Gr(U', B)$ is not in positive region, i.e., $g' \cap POS_B(\{d\}) = \phi$ then the associated $g \in Gr(U, B)$ is also not in positive region of *DT*, i.e., $g \cap POS_B(\{d\}) = \phi$.

The determination of being in non positive region requires at least one pair of objects belonging to different decision classes of a granule are selected in the sampling process. The chance for not picking even a single pair is more when the granule objects belonging to a second decision class are few in number. But if the granule g objects are almost uniformly distributed into several decision classes then probability of not picking even a single pair is very less.

In summary, decisions arrived at positive region at *SDT* level are reflected in original *DT* excepting for occurrences of *false POS* granules and lost granules.

The attribute selection based on sample dataset is reliable owing to following reasons accounting the possibilities of *false POS* granules, lost granules.

1. The occurrence of *false POS* granules, lost granules are possible with respect to all available attributes in the selection process in an iteration of QRA. Hence the relative significance obtained can still mimic the selection process of *DT*.
2. The aim of the attribute selection using sample dataset is not to select the same attribute sequence selected by an algorithm working on *DT*. But aim is to select an attribute set B as a reduct or super reduct such that as a group B attributes achieves desired gamma and the size of B is similar to size of the reduct obtained working on the whole dataset. Hence there is a high chance that selection of significant attributes (even though may not be highly significant on original *DT*) on sample *DT* can result in as a significant collection of attributes as a group for *DT* as long as least significant (irrelevant) attributes are not selected in the process.

3. Taking the analogy from the variable precision rough sets [6, 175], when the granule in the original dataset is having very less β value there is a more chance of not having *false POS* granule. This is because in the original granule, objects belonging to different decision classes are available and for preserving the non positiveness of the granule at least two objects belonging to different classes are needed. The chance for occurrence of *false POS* granule from sample dataset is more when the β is near to 1 as very few objects belonging to different decision class than the majority decision class are available in the granule. But even this *false POS* granule obtained from sampled dataset will contribute to gamma measure in the future iterations as its β value is very near to 1.

Hence sampling based reduct computation approach is a viable alternative to reduct finding for large decision systems. The next section describes the proposed methodology for sampling based reduct finding algorithms and outlines its differences with existing approaches.

4.3 Proposed Approach for Sample Based Reduct Computation

Attribute reduction used in SARA, FSR algorithms rely only on sample datasets or sampled discernibility matrix (i.e., MDL) for attribute selection and in case of SARA for attribute validation. As no operations are performed with respect to the original dataset both the algorithms are assured of returning sub reduct or an ϵ -approximate reduct [125]. Both methods do not use any goal of achieving a gamma measure and in other words, in the resulting ϵ -approximate reduct, the value of ϵ is unknown.

The proposed sample based reduct computation approach is based on the structure and philosophy of IQRA_IG algorithm. Hence, contrary to the existing sample based approaches which are based on discernibility matrix, a sample based approach is being proposed based on heuristic algorithm. The aim of the approach is to result in super reduct or reduct than only arriving at sub reduct as in existing approaches. As a result of the algorithm, the gamma measure value of the recommended reduct needs to be known which establishes the validity of reduct computed.

The computations done in i^{th} iteration of IQRA_IG algorithm can be divided into two parts. The first part comprises of the new attribute selection to be included in the recommended reduct R and involves trivial ambiguity resolution using secondary heuristic. The second part consists of ascribing the se-

lected attributes on the dataset in computing positive region, gamma measure, i.e., $POS_R(\{d\})$, $\gamma_R(\{d\})$ and using the principle of positive region removal for reducing the dataset for next iterations. The time complexity of the first part is $O(|C - R| |U_i| \log |U_i|)$ where U_i denote the set of non positive region objects on which computations are done in i^{th} iteration of IQRA_IG. The time complexity of the second part, using the principle of granular refinement discussed in Section 3.2 of Chapter 3, is only $O(|U_i| \log |U_i|)$. Hence the second part is computationally feasible for large or very large datasets. This observation lead to arrive at an hybrid approach which consists of local selection of attributes based on sample dataset(s) as the first part and global validation of selected attributes in the second part.

The algorithms proposed using this approach are discussed in the next section. The first part of these algorithms are specific to each algorithm but all of them commonly have the same global validation step in second part. The result of first part in an iteration is, in contrast to IQRA_IG, a group of new attributes included into recommended reduct R . If N is the number of new attributes added then the complexity of global validation step is $O(N |U_i| \log |U_i|)$. N is usually a much smaller number than $|C|$ making the global validation step a feasible operation. As iteration progresses the time complexity reduces because of positive region removal. Empirically the comparison of computation of time of local and global steps are discussed in Section 4.6.

IQRA_IG algorithm requires the computation of gamma measure of all conditional attributes $\gamma_C(\{d\})$ which is used in the terminating criteria for the algorithm. Sampling based algorithms are recommended for large or very large decision systems having large number of objects and conditional attributes. In case both number of objects and attributes is large then computation of $\gamma_C(\{d\})$ can become a time consuming computation or infeasible owing to memory constraints for loading the entire dataset into memory. Hence algorithms proposed assume that $\gamma_C(\{d\})$ is given as an input and is computed either offline or taken from the reported literature of the dataset. In case the value of $\gamma_C(\{d\})$ is not known and computation of $\gamma_C(\{d\})$ is to be avoided due to memory or time complexity reasons then the following modified terminating criteria is used.

Terminating Criteria#2: If $\gamma_R(\{d\})$ is 1 return or return in case of the persistence of trivial gamma gain (zero gamma gain) for L (pre defined number) of attribute insertions into R .

In case number of iterations without gamma gain exceeded L , algorithm using this end condition returns the recommended reduct R after removal of L attributes inserted without gamma gain. In case of inconsistent dataset (where $\gamma_C(\{d\}) < 1$) the redundancy of computing excess L attributes usually happens on a very small

reduced dataset (due to positive region removal) and hence does not impact the complexity of the algorithm significantly. In experiments, L value is set to 5 based on empirical analysis.

There is another possibility in which the requirement for modified end condition does not arise. In inconsistent datasets usually the amount of inconsistency is very minimal. That is usually $\gamma_C(\{d\}) > 0.5$. That is at least more than 50% of objects are in positive region. If $\gamma_C(\{d\}) < 0.5$, such inconsistent datasets are not suitable for data mining applications. As dataset reduces in size in each iteration owing to positive region removal, there will come to a situation where computing positive region and gamma measure on the remaining dataset is feasible and using the current positive region count and computing $POS_C(\{d\})$ on the remaining dataset one can arrive at $POS_C(\{d\})$ for the original dataset and hence $\gamma_C(\{d\})$ can be computed as an intermediate step. Hence except in rare situations, as having trivial ambiguous situations occurring for many iterations during initial iterations of algorithm, proposed algorithms will result in reduct or super reduct.

The important advantage of global validation step is achieving reduct or super reduct. The other advantage being, in any iteration the sample dataset (s) is (are) generated confining to non positive region objects. This makes the search process focused and helps in reducing the redundancy in the recommended reduct. Hence global validation step not only impacts time and space complexities but also acts as a global controller for facilitating guided search for relevant attributes.

In the first step of the proposed approach, the attributes are selected from sampled dataset by applying a single reduct algorithm. From the results obtained in Chapter 3 it is established that IQRA_IG is a computationally efficient single reduct algorithm and has ability to deal with trivial ambiguous situations such that the algorithm results in reduct or super reduct with minimal redundancy. Hence in all proposed sampling based approaches, IQRA_IG has been used to apply on sampled datasets for selection of attributes. The next two sections describe the proposed algorithms using this approach and the algorithms are divided into persistent and non persistent categories.

4.4 Persistent Sampling Based Approaches

Let $DT = (U, C \cup D, \{V_a, f_a\}_{a \in C \cup D})$ represent symbolic decision system as described in chapter 3. In this section two algorithms following the methodology described in the earlier section are proposed. Both algorithms are called as persistent sampling based approaches as the sample generated is persistently used for extracting the maximum possible attributes that can be selected to the extent of potential of that sample into recommended reduct R .

4.4.1 Sample guided IQRA_IG (SGIQRA_IG) Algorithm

Sample guided IQRA_IG (SGIQRA_IG) algorithm follows the same structure as IQRA_IG algorithm. The algorithm starts with recommended reduct R as empty set of attributes. The variable POSCOUNT is used to represent the number of positive region objects obtained and is initialized to zero. In i^{th} iteration a sample dataset SDT is generated using random sampling from the set of non positive region objects U_i where $U_1 = U$ and $U_i = U - POS_R(\{d\})$ for $i > 1$. Algorithm IQRA_IG is applied on SDT with an variation from the standard approach. The variation is to start with existing R as already included attributes into recommended reduct instead of starting with empty set of attributes. The application of IQRA_IG on SDT result in new set of attributes IR such that $R \cup IR$ is a (super) reduct for SDT . IR is included into R and the global validation step is applied on DT by computing $POS_R(\{d\})$ and removing them from U_i such that remaining dataset comprising of non positive region objects U_{i+1} to be used in the next iteration. POSCOUNT is updated and $\gamma_R(\{d\})$ is computed for original dataset. The algorithm terminates when the required end condition $\gamma_R(\{d\}) = \gamma_C(\{d\})$ is obtained. In case $\gamma_C(\{d\})$ is not available the Terminating Criteria#2 discussed in earlier section is used.

Algorithm 5: Sample Guided IQRA_IG (SGIQRA_IG)

Input : Symbolic Decision Table $DT = (U, C \cup \{d\}, \{V_a, f_a\}_{a \in C \cup \{d\}})$
Sample Size s
 $\gamma_C(\{d\})$: Gamma measure of Consistency for all conditional attributes
Output: R Super Reduct

- 1 Compute $Gr(U, C)$, $POS_C(\{d\})$, $\gamma_C(\{d\})$
- 2 $R = \emptyset$
- 3 $POSCOUNT = 0$
- 4 $OBJECTCOUNT = |U|$
- 5 **while** $\gamma_R(\{d\}) < \gamma_C(\{d\})$ **do**
- 6 $SDT = \text{GenerateRandomSample}(DT, s)$
- 7 $IR = \text{IQRA_IG}(SDT, R)$
- 8 $R = R \cup IR$
- 9 Compute $POS_R(\{d\})$
- 10 Reduce DT by removing $POS_R(\{d\})$ objects
- 11 $POSCOUNT = POSCOUNT + |POS_R(\{d\})|$
- 12 $\gamma_R(\{d\}) = \frac{POSCOUNT}{OBJECTCOUNT}$
- 13 **end**
- 14 **Return** R

Let $\epsilon = s/100$. Then the size of SDT generated in i^{th} iteration is $\epsilon |U_i|$. The time complexity of attribute selection using random sampled dataset in i^{th} iteration of SGIQRA_IG is $O(|C - R|^2 \epsilon |U_i| \log \epsilon |U_i|)$ as attribute selection assumes R as already included attributes into reduct. The time complexity of the global validation step in i^{th} iteration is $O(|IR| |U_i| \log |U_i|)$.

As the selection of attributes is based on sampled datasets, SGIQRA_IG obtains significant computational gains over IQRA_IG algorithm. But from the

experiments on large decision systems it is observed that SGIQRA_IG results in having non negligible Excess Count (EC) in the recommended reduct. With the objective of reducing EC, algorithm Stratified Sampling guided IQRA_IG (SSGIQRA_IG) is proposed and described in the next subsection.

4.4.2 Stratified sample guided IQRA_IG (SSGIQRA_IG) Algorithm

The disadvantage of random sampling is that those decision classes having less number of objects in the dataset may not get representation of SampleDT and also chance for more *false POS* granules. Stratified sampling alleviates class imbalance problem in random sampling by proportionally obtaining sampled objects with respect to all decision classes. Incorporating Stratified Sampling in SGIQRA_IG algorithm instead of random sampling resulted in algorithm Stratified Sample guided IQRA_IG (SSGIQRA_IG). The algorithm is given below.

Algorithm 6: Stratified Sample Guided IQRA_IG (SSGIQRA_IG)

Input : Symbolic Decision Table $DT = (U, C \cup \{d\}, \{V_a, f_a\}_{a \in C \cup \{d\}})$
Sample Size s
 $\gamma_C(\{d\})$: Gamma measure of Consistency for all conditional attributes
Output: R Super Reduct

- 1 Compute $Gr(U, C), POS_C(\{d\}), \gamma_C(\{d\})$
- 2 $R = \emptyset$
- 3 $POSCOUNT = 0$
- 4 $OBJECTCOUNT = |U|$
- 5 **while** $\gamma_R(\{d\}) < \gamma_C(\{d\})$ **do**
- 6 $SDT = \text{GenerateStratifiedSample}(DT, s)$
- 7 $IR = \text{IQRA_IG}(SDT, R)$
- 8 $R = R \cup IR$
- 9 Compute $POS_R(\{d\})$
- 10 Reduce DT by removing $POS_R(\{d\})$ objects
- 11 $POSCOUNT = POSCOUNT + |POS_R(\{d\})|$
- 12 $\gamma_R(\{d\}) = \frac{POSCOUNT}{OBJECTCOUNT}$
- 13 **end**
- 14 Return R

The time complexity analysis of SSGIQRA_IG is same as SGIQRA_IG. Both random sampling and stratified sampling are of linear complexity and does not majorly attribute any difference in time complexity to SSGIQRA_IG.

It is observed in experimental analysis that SSGIQRA_IG has not significantly reduced Excess Count. A non persistent sampling based approach is developed to solve the problem of Excess Count and is given in the next section.

4.5 Non Persistent Sampling Based Approach

Analysis of results of SGIQRA_IG and SSGIQRA_IG identified that using a single *SDT* in iteration for estimation of attributes into *R* is leading to occurrence of more redundant attributes in the final recommended reduct. The bias given for a *SDT* will be ascribed to the *DT*. Especially the attributes included into *R* towards the final iterations of IQRA_IG algorithm for *SDT*, even with reasonable significance for *SDT*, may not have considerable significance for original dataset. The identification of candidate attributes in iteration from a *SDT* will be highly influenced by the information available in the *SDT*, i.e., the extent capability of inherent knowledge of the *SDT* will be a limitation. Whereas a potential attribute from a given *SDT* will utilize potential knowledge. By drawing another *SDT* for discovering the next predominant Sample will address over fitting to the limited knowledge of the given *SDT*. This situation is analogous to the designing in a Multi fidelity domain [144, 145]. This lead to an important shift in the approach by considering different samples for determining predominant attributes one by one. This approach will reduce the persistent influencing factor of the inherent knowledge of the *SDT*. Hence the proposed method is named as Non Persistent Stratified Sampling based IQRA_IG (NPSS_IQRAIG). Next sub section describes NPSS_IQRAIG algorithm.

The following is the summary of modifications included in NPSS_IQRAIG over SSGIQRA_IG, SGIQRA_IG.

- Introduction of a new parameter NOS (Number of sample datasets to be generated in each iteration)
- *SDT* is used only to perform only the first iteration of IQRA_IG, i.e., for selection of only one new attribute into *R*. Obviously the selection of new attribute is done by assuming that *R* is the available reduct for *SDT*.

Global validation step is as same as that used in SGIQRA_IG and SSGIQRA_IG algorithms. The algorithm NPSS_IQRAIG is given below.

4.5.1 Non Persistent Stratified Sampling based IQRA_IG (NPSS_IQRAIG) Algorithm

The initial conditions of NPSS_IQRAIG are same as previous algorithms. In i^{th} iteration of NPSS_IQRAIG, NOS number of stratified sampled datasets are generated using replacement. On every sampled dataset getBestAttr is used for selection of next attribute into *R*. The algorithm getBestAttr is used to select at most one attribute from the generated Stratified Sampled Dataset SampleDT by assuming *R* is already available selected attribute set. Hence it involves obtaining

non positive region objects in step 1 and 2 based on positive region obtained with R and performing only the first iteration of IQRA_IG for selection of the significant attribute SA in remaining steps. It may happen that DTRemain will be empty when R is a reduct for SampleDT. In that case the returned value is '-1' signifying no attribute is selected. Hence at most NOS number of attributes are included in an iteration of NPSS_IQRAIG. After selection of attributes, global validation follows as used in persistent approaches.

Algorithm 7: Non Persistent Stratified Sampling based IQRA_IG (NPSS_IQRAIG)

Input : Symbolic Decision Table $DT = (U, C \cup \{d\}, \{V_a, f_a\}_{a \in C \cup \{d\}})$
Sample Size SS
 $\gamma_C(\{d\})$: Gamma measure of Consistency for all conditional attributes
 NOS : Number of Samples to be used in an iteration
Output: R Super Reduct

```

1  $R = \emptyset$ 
2  $TObjectCount = |U|$  //Size of the dataset
3  $POSCOUNT = 0$  //Size of Positive Region
4  $\gamma_R(\{d\}) = 0$ 
5 while  $\gamma_R(\{d\}) < \gamma_C(\{d\})$  do
6   for  $i \leftarrow 1$  to  $NOS$  do
7      $S[i] = genStratifiedSample(DT, SS)$ 
8      $a = getBestAttr(S[i], R)$ 
9     if  $a \neq -1$  then
10       $R = R \cup \{a\}$ 
11    end
12  end
13   $NewPOS = POS_R(\{d\})$  //New positive region objects in the reduced dataset because of addition
    of attributes into  $R$ 
14   $POSCOUNT = POSCOUNT + |NewPOS|$ 
15   $\gamma_R(\{d\}) = \frac{POSCOUNT}{TObjectCount}$ 
16   $U = U - NewPOS$  //Reduce Decision Table by removing positive region objects
17 end
18 Return  $R$ 

```

The getBestAttr subroutine used in NPSS_IQRAIG algorithm is given below.

Algorithm 8: getBestAttr

Input : $SampleDT = (U, C \cup \{d\}, \{V_a, f_a\}_{a \in C \cup \{d\}})$
 $PReduct$: Partial Reduct (Subset of C denoting the already selected attributes)
Output: SA : Selected attribute to be included into $PReduct$

```
1  $PRegion = POS_{PReduct}(\{d\})$ 
2  $U = U - PRegion$ 
3 if  $|U| = 0$  then
4    $SA = -1$ 
5   Return
6 end
7  $AAttr = C - PReduct$  // Available attributes for next best attribute selection
8  $T = PReduct$ 
9 foreach  $a$  in  $AAttr$  do
10   if  $\gamma_{PReduct \cup \{a\}}(\{d\}) > \gamma_T(\{d\})$  then
11      $SA = a$ 
12      $T = PReduct \cup \{a\}$ 
13   end
14 end
15 if  $T = PReduct$  then
16   // Trivial ambiguity situation
17   Select  $SA$  using IG Heuristics
18 end
19 Return  $SA$ 
```

The time complexity of $O(|C - R| \epsilon |U_i| \log \epsilon |U_i|)$ is incurred by getBestAttr in i^{th} iteration of NPSS_IQRAIG. Hence the time complexity of attribute selection process in i^{th} iteration of NPSS_IQRAIG is $O(NOS |C - R| \epsilon |U_i| \log \epsilon |U_i|)$. The time complexity of global validation step in i^{th} iteration is $O(NOS |U_i| \log |U_i|)$ using granular refinement in obtaining positive region. Between proposed algorithms it is difficult to obtain theoretically better time complexity algorithm. It depends on the size of the recommended reduct R obtained which in turn depends on which algorithm obtained less EC. Hence this aspect will be discussed empirically in experiment analysis. But it is evident that all three algorithms have better time complexity than IQRA_IG as complex step of attribute selection is performed on the sample dataset.

As discussed in section 4.2, a sample may not represent the original dataset due to *false POS* granules and lost granules. There is more chance for *false POS* granules if the original dataset is class imbalanced. Hence stratified sampling helps in reducing the occurrences of *false POS* granules by sampling objects from less balanced decision class objects. The chance for *false POS* granules further decreases in non persistent methodology as a new stratified sample is used for each attribute selection.

In single iteration the sample datasets are generated using stratified sampling with replacement. Sampling with replacement assures of representing the relationship between objects present in the original dataset to a greater extent than the kind of sampling done in persistent approaches.

Table 4.1: Details of Datasets and IQRA_IG Results

Dataset Details				IQRA_IG Results		
Name	Objects	Attributes	Decision Classes	AL	EC	CT (sec)
BASEHOCK	1993	4862	2	47	7	369.97
PCMAC	1943	3290	2	39	2	457.38
RELATHE	1427	4323	2	28	1	433.98

Global validation used in persistent and non persistent approaches establish interaction between sample dataset and original dataset. Global validation enables all proposed approaches resulting in reduct or super reduct compared to resulting in only sub reduct in the existing sampling based approaches in literature.

In persistent approaches a single sample dataset is used to select the attributes till the IQRA_IG terminates on the sample dataset. As positive region removal is used in IQRA_IG towards the middle to end iterations of IQRA_IG the attribute selection will be based on a small portion of sampled dataset. Hence these attributes are selected at a much lower sample size than the initial iteration attribute selection. Hence their relevance with original dataset diminishes and chance for occurrence of more EC increases. But in non persistent approach a single attribute selection is based on a new stratified sample generated using sampling with replacement. Hence the attribute selection usually occurs at the similar sample size. Hence the chance for selection of relevant attributes and thus reduction of EC is high in non persistent approach.

4.6 Experiments and Results

The experiments were conducted to establish the performance of the proposed algorithms on large decision systems and also a comparative study with the experiments done in [44] using SARA algorithm is done. The definition of large decision system considered is a decision system containing both large number of objects and large number of attributes. Three large decision systems of this nature are considered for experiments which are from feature selection dataset repository of Arizona State University [146]. The details of these datasets along with the results obtained using IQRA_IG algorithm are given in Table 4.1.

For each dataset in Table 4.1 the experiments were conducted at SS values 0.5%, 1%, 5%, 10%, 15%, 20%. Experiments with more SS than 20% are resulting in meager computational gains over IQRA_IG and is counterproductive to aim of achieving scalability hence the bound for SS till 20%. At each SS the experiment is repeated for 100 trails. The repeated number of trails helps in understanding the behavior of the algorithm especially with respect to impact

Table 4.2: Summary of SGIQRA_IG Results for 100 Trails

Dataset	SS	AL	IC	CT	EC	RC
BASEHOCK	0.50%	109,169	90,156	420.86 (56.2)	43,109	0
	1%	59,95	28,61	270.41 (33.59)	10,41	0
	5%	42,60	4,6	195.23 (21.4)	2,13	0
	10%	37,60	2,3	213.46 (30.2)	1,13	0
	15%	35,56	2,3	261.6 (38.9)	0,16	1
	20%	40,59	2,2	275.01 (49.2)	1,17	0
PCMAC	0.50%	96,178	88,162	282.13 (45.06)	40,117	0
	1%	52,86	26,54	171.84 (22.7)	5,33	0
	5%	41,58	4,6	134.33 (13.6)	1,11	0
	10%	39,53	3,4	156.34 (21.19)	0,9	5
	15%	38,54	2,3	191.28 (23.37)	0,11	2
	20%	39,56	2,2	203.12 (34.67)	0,17	2
RELATHE	0.50%	88,170	82,168	305.81 (44.112)	38,112	0
	1%	44,78	30,66	208.33 (29.6)	11,36	0
	5%	27,39	4,7	122.05 (14.3)	1,12	0
	10%	26,39	3,4	144.2 (21.8)	0,9	4
	15%	24,40	2,3	162.6 (18.1)	0,10	12
	20%	26,41	2,3	188.8 (22.8)	0,7	12

of sampling strategy adopted in the algorithm. Summary of 100 repeated trails obtained using SGIQRA_IG algorithm are given in Table 4.2. Similarly that of SSGIQRA_IG are given in Table 4.3 and of NPSS_IQRAIG are given in Table 4.4.

The description of the terms used in reporting the results is applicable to all Tables but given here more particularly with reference to results given in Tables 4.2, 4.3 and 4.4. The results are specified using notations,

- NOD: Number of decision classes
- SS: Sample Size (in percentage)
- AL: Length of the recommended reduct R (Spread containing minimum AL, maximum AL for 100 trails is reported)
- IC: Iteration Count (Number of iterations involved is reported as minimum IC, maximum IC for 100 trails)
- CT: Computational Time in seconds (Reported as mean(standard deviation) for 100 trails)
- EC: Excess Count (Number of redundant attributes present in recommended reduct R and reported as minimum EC, maximum EC of 100 trails)
- RC: Reduct Count (Number of Reducts without any redundancy (EC=0) found in 100 trails)

Table 4.3: Summary of SSGIQRA_IG Results for 100 Trails

Dataset	SS	AL	IC	CT	EC	RC
BASEHOCK	0.50%	91,154	74,141	343.88 (42.35)	33,89	0
	1%	57,94	24,53	238.11 (32.02)	10,40	0
	5%	41,56	4,6	185.45 (18.89)	0,13	1
	10%	40,54	3,3	209.93 (32.24)	1,15	0
	15%	40,54	2,3	258.06 (29.13)	0,12	1
	20%	41,58	2,2	239.21 (46.6)	2,15	0
PCMAC	0.50%	83,144	62,131	230.13 (32.56)	24,86	0
	1%	51,80	23,52	159.44 (22.48)	8,32	0
	5%	39,52	4,6	132.2 (14.02)	1,10	0
	10%	39,53	3,4	157.5 (22.72)	0,11	1
	15%	38,56	2,3	185.62 (21.55)	0,11	1
	20%	38,59	2,2	197.93 (35.21)	0,15	4
RELATHE	0.50%	70,136	58,129	254.13 (42.67)	25,90	0
	1%	43,74	29,56	182.47 (28.35)	6,31	0
	5%	28,39	4,7	120.04 (14.2)	0,8	1
	10%	26,37	3,4	139.14 (22.48)	0,8	7
	15%	26,39	2,3	161.77 (15.4)	0,7	13
	20%	24,36	2,3	181.39 (20.5)	0,6	14

A comparative study is conducted with SARA algorithm using the results reported in [44]. The datasets are from UCI Machine Learning repository [11]. For completeness the results reported in [44] are given in Table 4.5. The results obtained by applying proposed algorithms on the datasets used in [44] are given in Table 4.6.

4.7 Analysis of Results

In this section the analysis of results reported in earlier section is given. Section 4.7.1 deals with the experiments and results obtained using large decision systems given in Table 4.1. Section 4.7.2 contains analysis of experiments done for comparative study of proposed algorithms with SARA algorithm.

4.7.1 Analysis of Results with large decision systems

In all three algorithms NPSS_IQRAIG, SSGIQRA_IG, SGIQRA_IG, very small sample sizes of 0.5%, 1% resulted in more IC and more EC. Even though in NPSS_IQRAIG CT and EC reduced in comparison to SSGIQRA_IG, SGIQRA_IG, the EC obtained is still large to be acceptable. Hence it is recommended to avoid such small samples sizes as they cannot represent the dataset (by resulting in more lost granules and more *false POS* granules) in choosing appropriate attributes into

Table 4.4: Summary of NPSS_IQRAIG Results for 100 Trails

Dataset	Sample Size	AL	IC	CT	EC	RC
BASEHOCK	0.50%	74,117	22,36	96.27 (7.91)	17,51	0
	1%	64,119	16,46	92.82 (6.77)	15,54	0
	5%	42,56	9,12	124.53 (11.3)	0,12	1
	10%	40,52	9,12	152.09 (15.68)	0,15	3
	15%	39,52	9,11	188.26 (19.79)	0,10	2
	20%	39,51	8,11	222.21 (21.72)	0,10	6
PCMAC	0.50%	70,120	18,35	61.8 (5.81)	18,51	0
	1%	67,107	18,36	60.29 (5.44)	13,45	0
	5%	41,55	9,12	85.27 (6.69)	0,9	2
	10%	39,49	8,11	113.09 (8.24)	0,7	3
	15%	39,49	8,11	149.45 (10.8)	0,6	11
	20%	38,48	8,11	177.86 (15.42)	0,6	15
RELATHE	0.50%	51,99	15,35	61.47 (5.63)	12,43	0
	1%	48,84	14,31	59.62 (6.01)	6,41	0
	5%	28,39	6,9	74.43 (7.119)	0,8	4
	10%	28,37	6,9	90.2 (7.44)	0,8	7
	15%	26,34	6,9	113.03 (10.23)	0,6	15
	20%	24,33	5,8	137.65 (12.88)	0,5	33

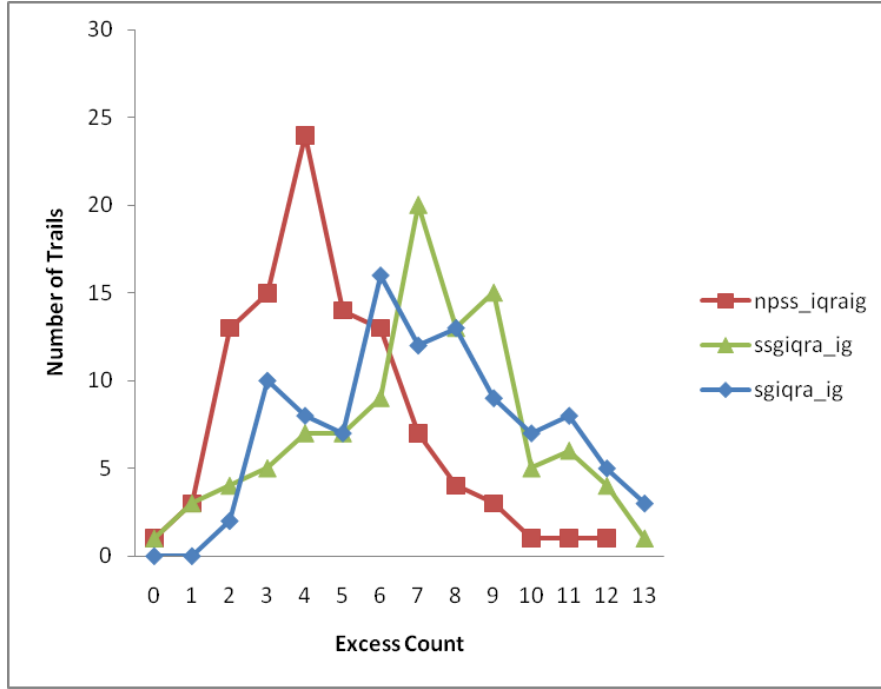
Table 4.5: Results of SARA algorithm reported from [44]

Dataset	Inst.	Attr.	Opt.Len	Sampling Times	Sampling Rate	Appr. Len.	Time
Letter	14999	17	12	150	100/14999	8 (0.009)	8.76
Shuttle-small	3866	10	5	40	100/3866	4 (0.017)	0.56
Mushroom	5415	23	4	60	100/5415	5 (0.002)	2.44
Chess	2130	37	24	30	100/2130	15 (0.076)	1.88
Satimage	4435	37	12	50	100 /4435	11 (0.007)	4.56
DNA	2000	181	22	20	100/2000	19 (0.010)	5.16
Adult	32561	15	12	350	100/32561	13 (0.000)	8.70
Hypothyroid	2108	26	12	20	100/2108	4 (0.027)	0.21
Shuttle	43500	10	6	450	100/43500	4 (0.064)	6.49

Table 4.6: Comparative Results with SARA algorithm (SG: SGIQRA_IG, SSG: SSGIQRA_IG, NPSS: NPSS_IQRAIG)

Dataset	Inst.	Samples			AL			EC			CT		
	(* differ)	SG	SSG	NPSS	SG	SSG	NPSS	SG	SSG	NPSS	SG	SSG	NPSS
Letter	20000*	2	2	15	12	12	12	0	1	1	3.42	3.42	3.55
Mushroom	5644*	1	1	10	3	3	4	0	0	1	0.03	0.03	0.06
Chess	3196*	2	2	30	30	30	30	1	1	1	0.63	0.94	0.52
Satimage	4435	2	2	15	7	7	8	0	0	1	1.19	1.35	0.95
DNA	2000	3	3	25	20	19	23	0	0	1	5.54	6.33	3.83
Adult	32561	2	2	15	8	8	9	0	0	1	5.38	5.18	5.63
Hypothyroid	2108	3	2	2	3	2	2	1	0	0	0.12	0.10	0.10
Shuttle	43500	2	1	5	4	4	4	0	0	0	0.73	2.51	2.85

Figure 4.1: Excess Count Frequency Comparison for BASEHOCK Dataset 5% Sample Size Trails



recommended reduct. The following analysis is pertained only to 5% to 20% sample sizes.

All three algorithms have got significant computational gains in comparison to IQRA_IG. NPSS_IQRAIG has achieved the least computational times for almost all experiments. For example for PCMAC dataset for 5% sample size trails, NPSS_IQRAIG has achieved 81% computational gain over IQRA_IG whereas SGIQRA_IG and SSGIQRA_IG have achieved around 70% computational gain. Hence these algorithms are suitable for large decision systems and are scalable. The algorithm which further results in minimum size recommended reduct and less Excess count will be the preferred algorithm.

From the results reported in Tables 4.2, 4.3 and 4.4 Attribute Length AL parameter ranges are of smaller values in all sample size trails of 5% to 20% in NPSS_IQRAIG compared to other algorithms. Hence from the objective of obtaining near to minimal length reduct NPSS_IQRAIG is preferred. Along with the size of the reduct obtaining less Excess Count means less redundancy leads to better as well as compact classifier models built with reduct attributes. The remaining analysis will focus on Excess count parameter for finding the best algorithm.

The figures 4.1 to 4.15 depict the frequency of trails against the Excess Count out of 100 trails for different datasets for the proposed algorithms. The performance characteristic of these algorithms has been further established by adopting left tailed two sample t-test for various sample size combinations on Excess Count

Figure 4.2: Excess Count Frequency Comparison for BASEHOCK Dataset 10% Sample Size Trails

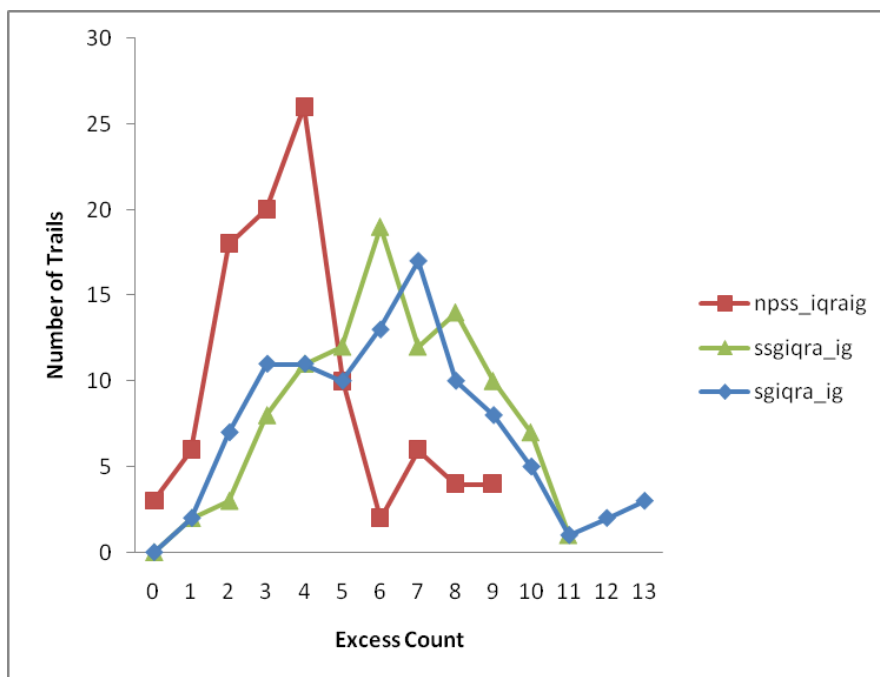


Figure 4.3: Excess Count Frequency Comparison for BASEHOCK Dataset 15% Sample Size Trails

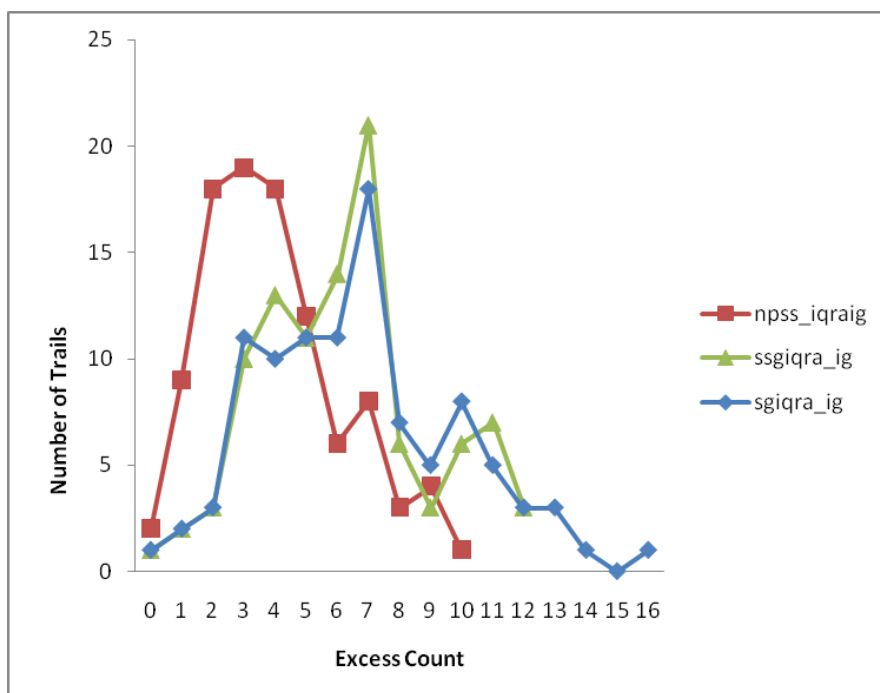
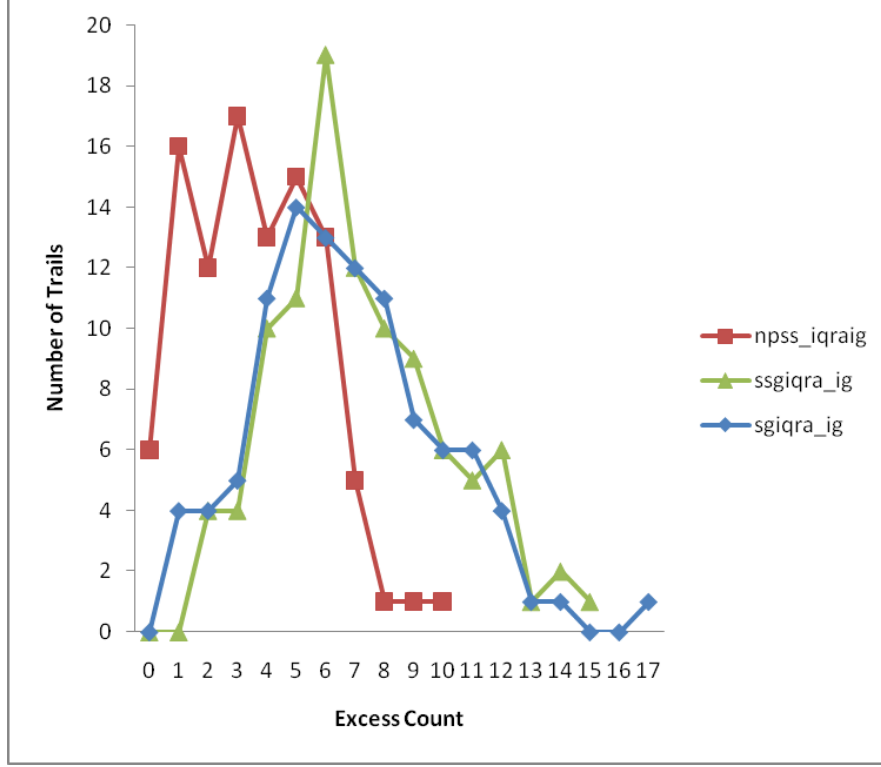


Figure 4.4: Excess Count Frequency Comparison for BASEHOCK Dataset 20% Sample Size Trails



values of repeated trails. The P-values obtained from these tests are reported in Tables 4.9, 4.10, 4.11 for Basehock, PCMAC, Relathe datasets respectively. The entry P_{ij}^{rs} is the P-value corresponding to the left tailed two sample t-test performance betterment analysis of the algorithm associated with i^{th} row at r^{th} percentage of sample size versus the algorithm associated to j^{th} column at s^{th} percentage of sample size. P-values less than or equal to significance level 0.05 indicate significant betterment (shown with light gray color back ground) and less than or equal to significance level 0.01 indicate highly significant betterment (shown with dark gray background).

The important observation from Tables 4.9 to 4.11 and Figures 4.1 to 4.15 is that for all three datasets NPSS_IQRAIG has achieved significant betterment over SSGIQRA_IG and SGIQRA_IG for almost all sample sizes. In Basehock dataset results primarily and also in PCMAC dataset, with 5% sample size trails NPSS_IQRAIG is highly significant over higher sample size trails of other two algorithms. Especially in Relathe dataset 33 trails have resulted in reducts (with EC=0) for 20% sample size experiment of NPSS_IQRAIG (See Figures 4.14, 4.15). Another observation is that in all three datasets NPSS_IQRAIG has achieved betterment with increase of sample size in reducing Excess Count (See Figures 4.5, 4.10, 4.15). This is highly significant in Relate and PCMAC datasets. From Figures 4.11 to 4.14 and Table 4.11 it is observed that NPSS_IQRAIG has similar performance with respect to 5%, 10% sample size trails with other two algorithms

Figure 4.5: Excess Count Frequency Comparison for BASEHOCK dataset for NPSS_IQRAIG algorithm for all sample sizes

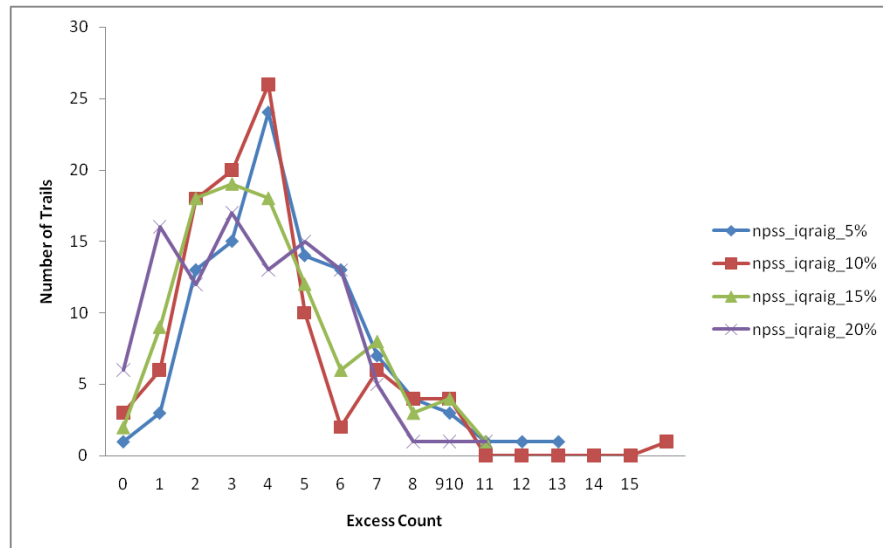


Figure 4.6: Excess Count Frequency Comparison for PCMAC Dataset 5% Sample Size Trails

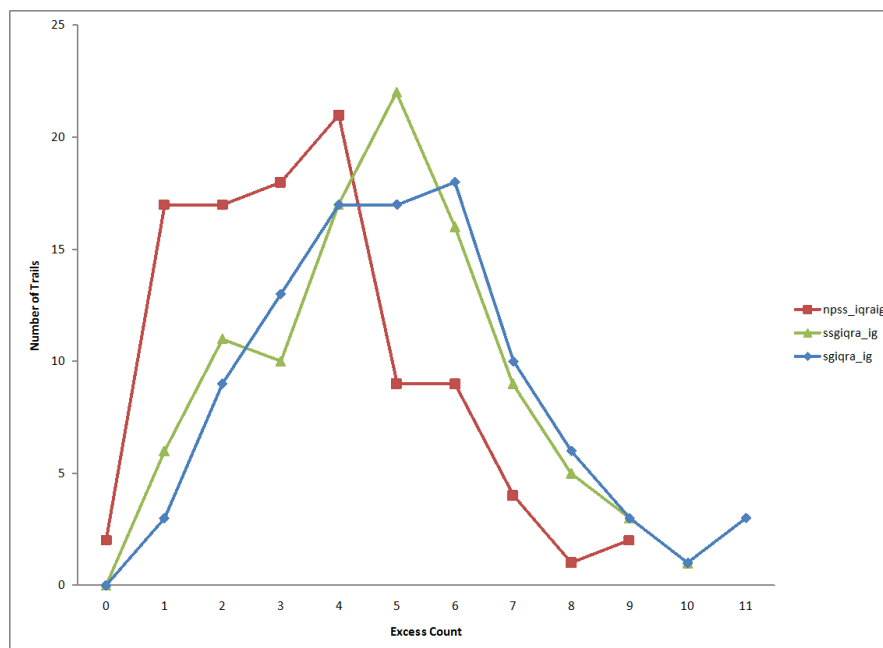


Figure 4.7: Excess Count Frequency Comparison for PCMAC Dataset 10% Sample Size Trails

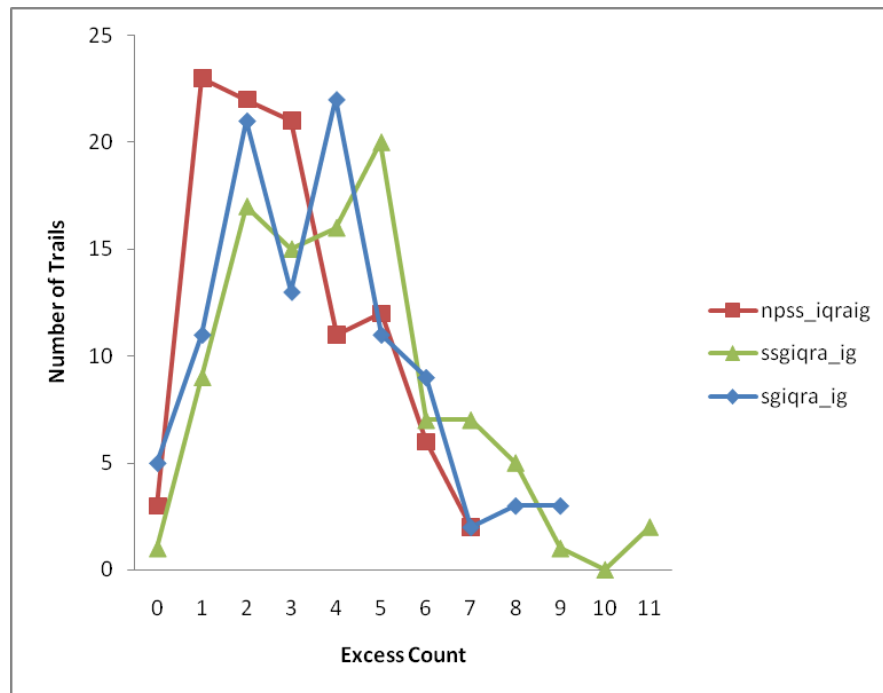


Figure 4.8: Excess Count Frequency Comparison for PCMAC Dataset 15% Sample Size Trails

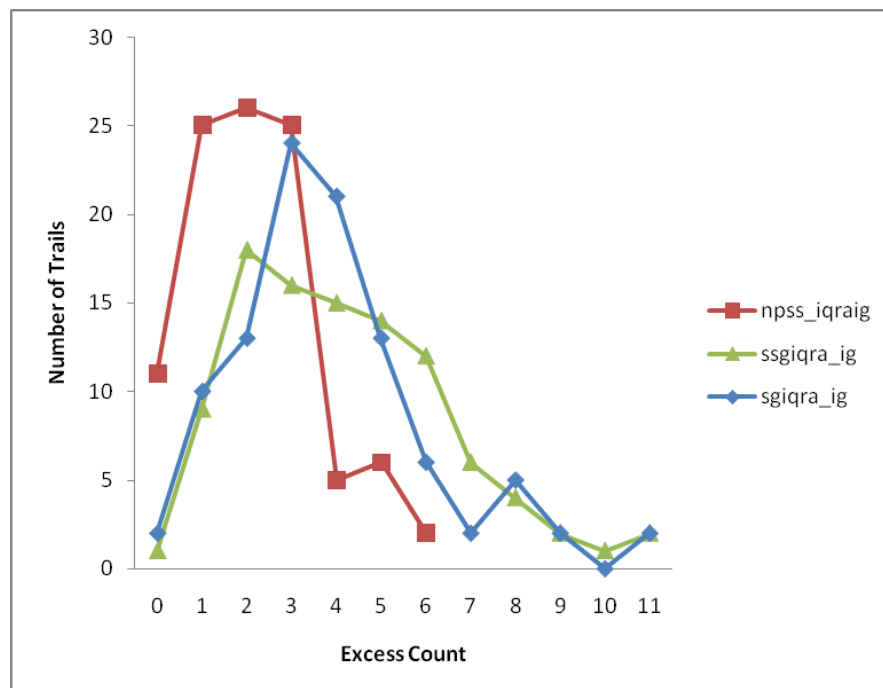
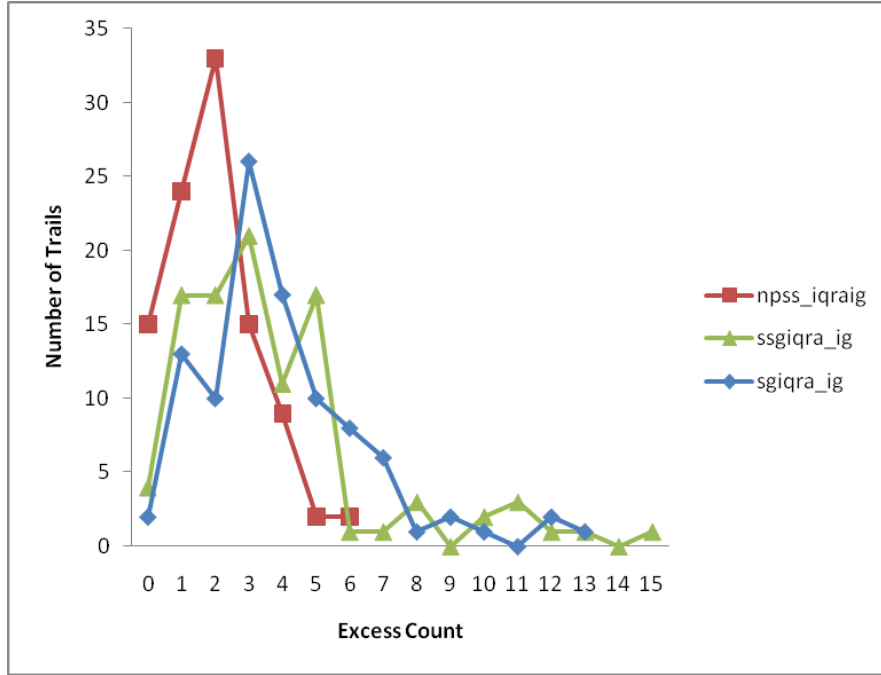


Figure 4.9: Excess Count Frequency Comparison for PCMAC Dataset 20% Sample Size Trails



in Relatthe dataset. But a significant betterment with 15% and highly significant betterment with 20% sample size trails is achieved by NPSS_IQRAIG over persistent algorithms.

Hence NPSS_IQRAIG is the better algorithm in giving less Excess Count with less Attribute Length in less computational time. SSGIQRA_IG, SGIQRA_IG have given mixed results in comparison with each other. Wherever there is a significant betterment it is always with respect to higher sample size trails of one algorithm over lower sample size trails of other algorithm. Only by NPSS_IQRAIG a lower sample size experiment has achieved better result than higher sample sizes of other algorithms consistently.

In some instances, the 10%, 15% trails of SGIQRA_IG and SSGIQRA_IG are better than their 20% sample size trails of respective algorithm. This is highly significant in SSGIQRA_IG in Basehock dataset (see Table 4.9). This kind of inconsistencies is not noticed in NPSS_IQRAIG. This monotonic behavior of NPSS_IQRAIG gained edge over other algorithms.

A measure coverage value is defined as the percentage of objects participated in attribute selection by being member of at least one sample dataset in a sampling based algorithm. For simplicity the number of times an object appeared in the sample datasets is not considered. As an illustration, the coverage values obtained for a single trail of NPSS_IQRAIG on Basehock dataset is depicted in Table 4.7 for different sample sizes. The corresponding values given for SGIQRA_IG and SSGIQRA_IG algorithms in Table 4.7 are estimated upper bound values based

Figure 4.10: Excess Count Frequency Comparison for PCMAC dataset for NPSS_IQRAIG algorithm for all sample sizes

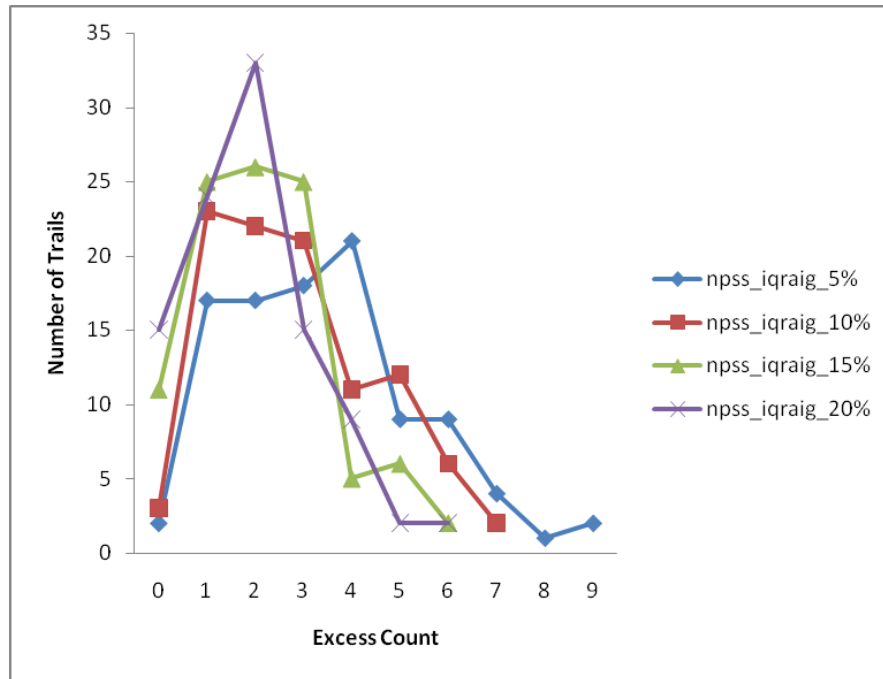


Figure 4.11: Excess Count Frequency Comparison for RELATHE Dataset 5% Sample Size Trails

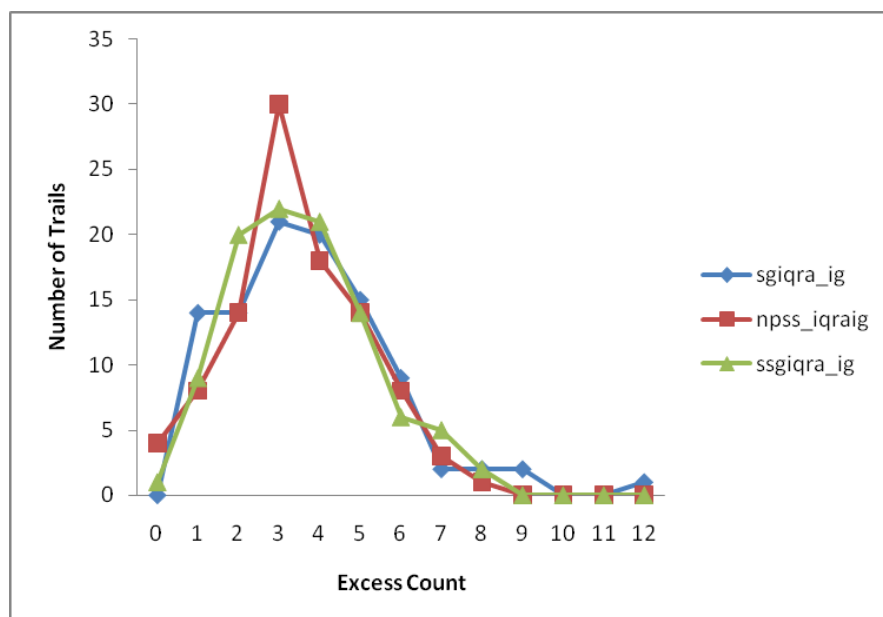


Figure 4.12: Excess Count Frequency Comparison for RELATHE Dataset 10% Sample Size Trails

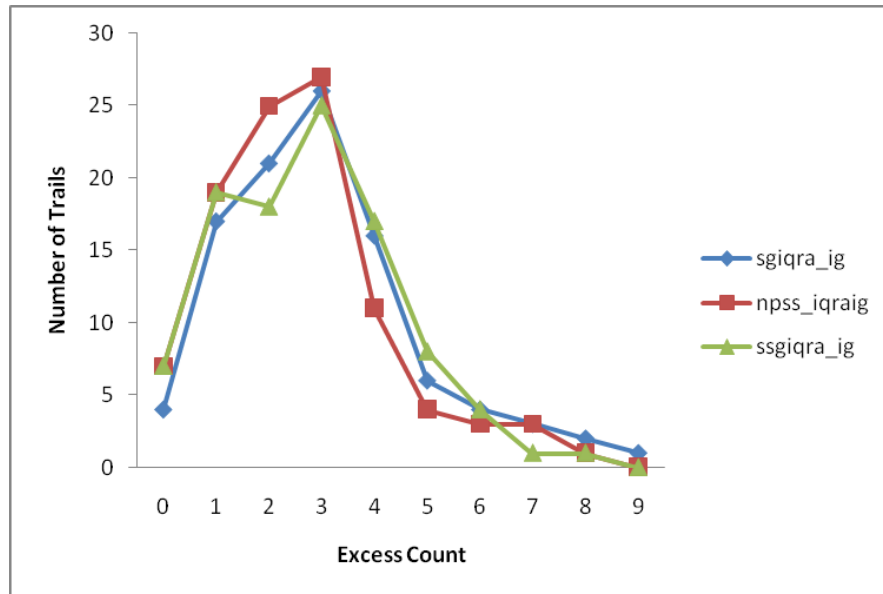


Figure 4.13: Excess Count Frequency Comparison for RELATHE Dataset 15% Sample Size Trails

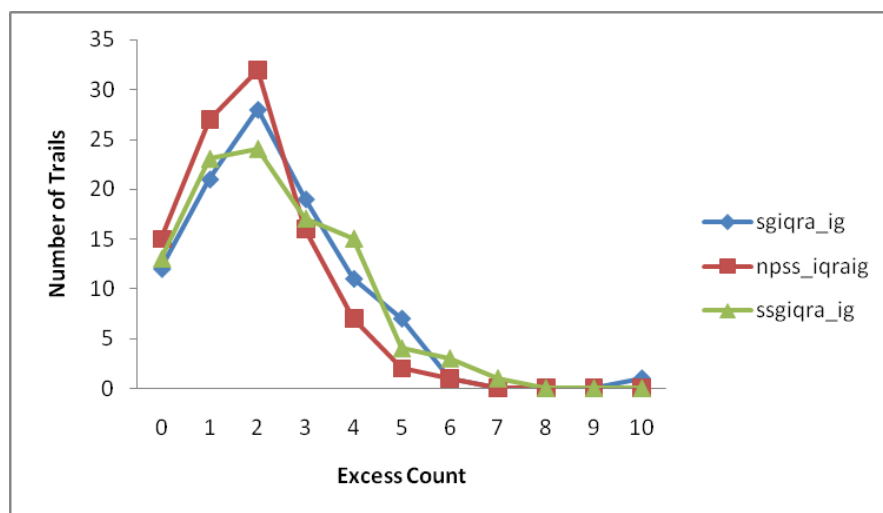
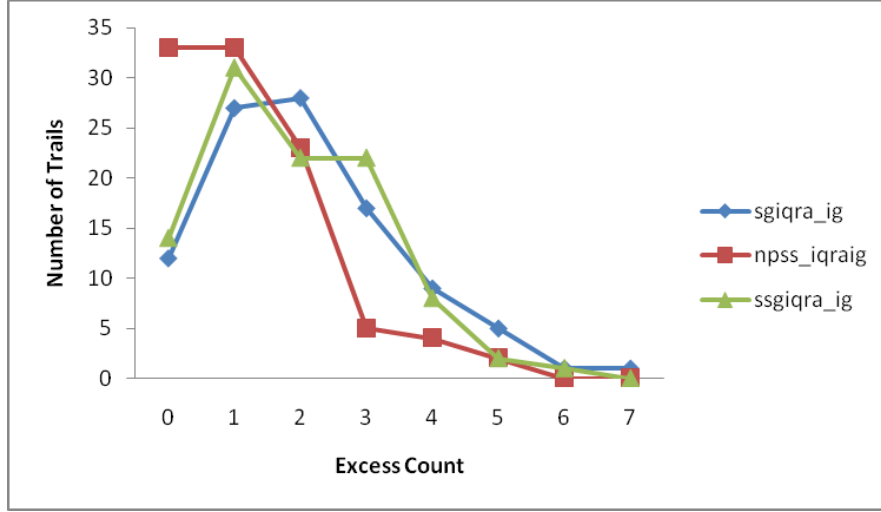


Figure 4.14: Excess Count Frequency Comparison for RELATHE 20% Sample Size Trails



on maximum IC obtained for the corresponding sample size (i.e., $\max \text{IC} \times \text{Sample Size}$). Table 4.7 illustrates the advantage of NPSS_IQRAIG approach as the non persistent sampling approach achieves higher coverage value compared to other sampling approaches used in SGIQRA_IG and SSGIQRA_IG. This is one of the important reasons for NPSS_IQRAIG to achieve better performance as attribute selection is based on using more objects from the original dataset reflecting better approximation of original dataset by sampled datasets.

Global validation step incorporated in proposed approaches enables achieving super reduct or reduct in comparison to obtaining only sub reduct in existing sampling based approaches. The computational cost of global validation step using original dataset is less than the computational cost of local attribute selection based on sampled datasets. To illustrate this, iteration wise computational costs involved in local and global steps are given in Table 4.8 for a single trail of Basehock dataset at 20% Sample Size using NPSS_IQRAIG algorithm. The results establish significantly the relevance of incorporating global validation step in a sampling based reduct computation approach.

IQRA_IG algorithm uses the entire dataset for selecting the Reduct attributes and hence it expected that Sampling based algorithms will result in more Excess Count than IQRA_IG. But significantly in the case of Basehock dataset NPSS_IQRAIG using 20% sample size has got 97 trails having Excess Count less than or equal to that of IQRA_IG (with $\text{EC}=7$). Out of 97 trails 92 trails have got Excess Count less than 7. This observation suggests that because of non deterministic sampling aspect is involved, there is a chance to get lower Excess Count than algorithms without using sampling.

Figure 4.15: Excess Count Frequency Comparison for Relathe dataset for NPSS_IQRAIG algorithm for all sample sizes

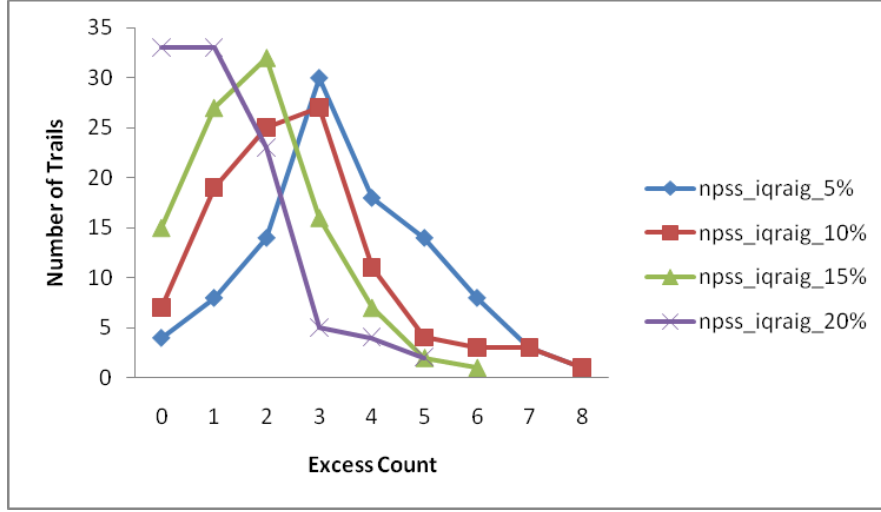


Table 4.7: Coverage Values for a single trail on Basehock Dataset

Sample Size	Coverage Value (%) for NPSS_IQRAIG	Estimated upper bound for coverage in SGIQRA_IG (%)	Estimated upper bound for coverage in SSGIQRA_IG (%)
5%	45.1	30	30
10%	62.9	30	30
15%	76.3	45	45
20%	82.9	40	40

Table 4.8: Iteration wise Attribute Selection (Local) CT vs Global Validation CT in a single trail of NPSS_IQRAIG on Basehock Dataset

Iteration	Local CT (sec)	Global CT (sec)
1	10.274	0.0079
2	22.313	0.02
3	46.483	0.012
4	73.997	0.006
5	40.856	0.003
6	18.29	0.002
7	11.47	0.0009
8	5.293	0.0006
9	1.62	0.0003
TOTAL	230.516	0.0527

Table 4.9: P-values for t-test results on BASEHOCK dataset

Algorithm		NPSS_IQRAIG				SSGIQRA_IG				SGIQRA_IG			
	SS	5%	10%	15%	20%	5%	10%	15%	20%	5%	10%	15%	20%
NPSS_IQRAIG	5%	0.5	0.98	0.98	1	0	0	0	0	0	0	0	0
	10%	0.02	0.5	0.46	0.86	0	0	0	0	0	0	0	0
	15%	0.02	0.54	0.5	0.89	0	0	0	0	0	0	0	0
	20%	0	0.14	0.11	0.5	0	0	0	0	0	0	0	0
SSGIQRA_IG	5%	1	1	1	1	0.5	0.96	0.97	0.33	0.27	0.98	0.75	0.73
	10%	1	1	1	1	0.04	0.5	0.59	0.01	0.01	0.66	0.17	0.15
	15%	1	1	1	1	0.03	0.41	0.5	0.01	0.01	0.58	0.14	0.47
	20%	1	1	1	1	0.67	0.99	0.99	0.5	0.44	0.99	0.86	0.85
SGIQRA_IG	5%	1	1	1	1	0.73	0.99	0.99	0.56	0.5	1	0.9	0.89
	10%	1	1	1	1	0.02	0.33	0.42	0.01	0	0.5	0.11	0.09
	15%	1	1	1	1	0.24	0.82	0.86	0.13	0.1	0.89	0.5	0.47
	20%	1	1	1	1	0.26	0.84	0.88	0.15	0.11	0.91	0.53	0.5

Table 4.10: P-values for t-test results on PCMAC dataset

Algorithm		NPSS_IQRAIG				SSGIQRA_IG				SGIQRA_IG			
	SS	5%	10%	15%	20%	5%	10%	15%	20%	5%	10%	15%	20%
NPSS_IQRAIG	5%	0.5	0.99	1	1	0	0.01	0.01	0.19	0	0.34	0.06	0.03
	10%	0.01	0.5	1	1	0	0	0	0	0	0.01	0	0
	15%	0	0	0.5	0.86	0	0	0	0	0	0	0	0
	20%	0	0	0.14	0.5	0	0	0	0	0	0	0	0
SSGIQRA_IG	5%	1	1	1	1	0.5	0.97	0.95	1	0.11	1	1	0.96
	10%	0.99	1	1	1	0.03	0.5	0.44	0.87	0	0.97	0.81	0.56
	15%	0.99	1	1	1	0.05	0.56	0.5	0.89	0	0.98	0.84	0.61
	20%	0.81	1	1	1	0	0.13	0.11	0.5	0	0.7	0.35	0.19
SGIQRA_IG	5%	1	1	1	1	0.89	1	1	1	0.5	1	1	1
	10%	0.66	0.99	1	1	0	0.03	0.02	0.3	0	0.5	0.14	0.06
	15%	0.94	1	1	1	0	0.19	0.16	0.65	0	0.86	0.5	0.27
	20%	0.97	1	1	1	0.04	0.44	0.39	0.81	0	0.94	0.73	0.5

Table 4.11: P-values for t-test results on RELATHE dataset

Algorithm		NPSS_IQRAIG				SSGIQRA_IG				SGIQRA_IG			
	SS	5%	10%	15%	20%	5%	10%	15%	20%	5%	10%	15%	20%
NPSS_IQRAIG	5%	0.5	1	1	1	0.32	1	1	1	0.14	0.96	1	1
	10%	0	0.5	1	1	0	0.25	0.93	1	0	0.05	0.91	0.99
	15%	0	0	0.5	1	0	0	0.02	0.37	0	0	0.01	0.1
	20%	0	0	0	0.5	0	0	0	0	0	0	0	0
SSGIQRA_IG	5%	0.68	1	1	1	0.5	1	1	1	0.25	0.98	1	1
	10%	0	0.75	1	1	0	0.5	0.98	1	0	0.17	0.98	1
	15%	0	0.07	0.98	1	0	0.02	0.5	0.97	0	0	0.47	0.81
	20%	0	0	0.63	1	0	0	0.03	0.5	0	0	0.03	0.17
SGIQRA_IG	5%	0.86	1	1	1	0.75	1	1	1	0.5	1	1	1
	10%	0.04	0.95	1	1	0.02	0.83	1	1	0	0.5	1	1
	15%	0	0.09	0.99	1	0	0.02	0.53	0.97	0	0	0.5	0.83
	20%	0	0.01	0.9	1	0	0	0.19	0.83	0	0	0.17	0.5

4.7.2 Analysis of Results of Comparative Experiments with SARA algorithm

The following analysis is based on the results reported in Tables 4.5, 4.6. The shuttle-small dataset reported in [44] could not be derived from the shuttle dataset in UCI Machine Learning repository [11]. Letter dataset size is 20000 objects, Chess with 3196 objects [11] which are different from size of the datasets reported in [44]. Mushroom dataset is having objects with missing attribute values. After removal of such objects, the resulting complete decision system contains 5644 objects and the same is used in current experiments. Apart from these datasets, the remaining datasets are having same size as used in SARA experiments.

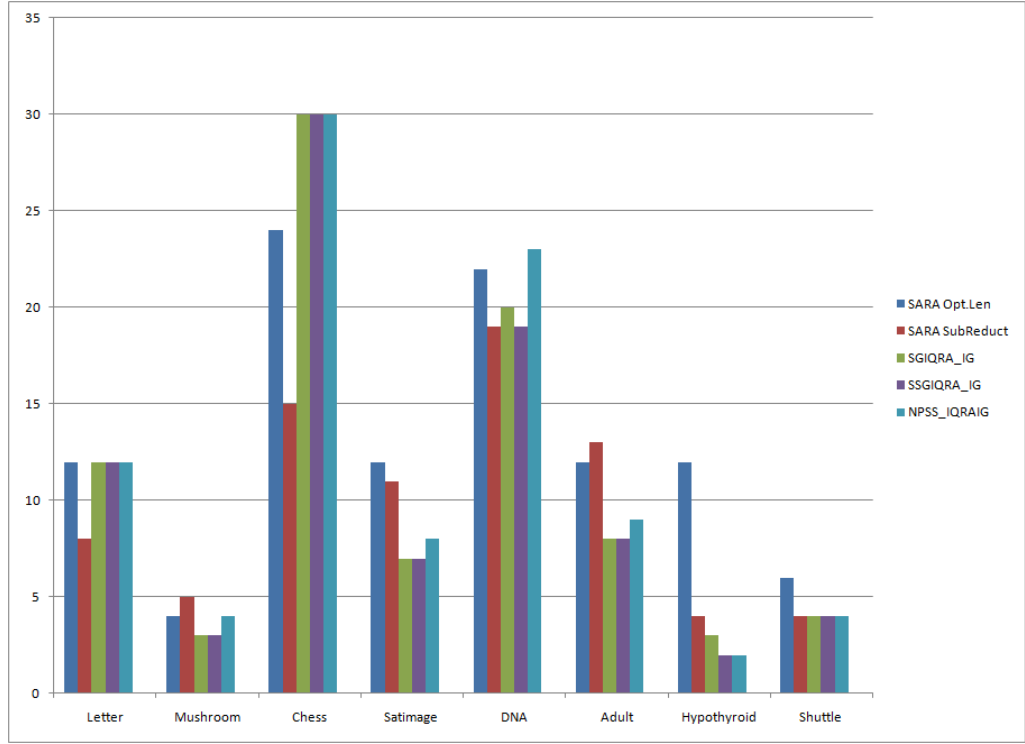
In SARA experiment results in Table 4.5, *Opt.Len* column refers to length of the minimum size reduct of the dataset, either obtained by exhaustive enumeration of reducts or by a heuristic algorithm. SARA algorithm aims to obtain a sub reduct of *Opt.Len* reduct. Experiment results pertaining to reduct length, comparatively displayed in Fig. 4.16, depict that much smaller reduct than reported as *Opt.Len* in SARA experiment results are obtained by proposed algorithms in all datasets except in Chess dataset, DNA dataset (only for NPSS_IQRAIG). Significantly proposed algorithms have obtained reducts or super reducts (with maximum EC of 1 in some cases), in comparison to sub reducts obtained in SARA experiment, using much fewer number of sample datasets. The proposed algorithms have resulted in less CT than SARA algorithm.

Relatively less computations times incurred for datasets having large number of objects like Adult, Letter, Shuttle by proposed algorithms establish the relevance of definition of large (computation intensive) dataset as that having large number of objects and attributes. Hence even though Adult, Letter, Shuttle like datasets contain many objects, as they contain fewer number of attributes, the task of computing reduct is feasible in less computational time.

4.8 Summary

Sampling based scalable approaches for reduct computation were developed in this chapter for complete symbolic decision systems. These approaches are categorized into persistent and non persistent approaches. Persistent approaches using random sampling, stratified sampling, namely SGIQRA_IG, SSGIQRA_IG, are able to compute super reduct with good computational gains over IQRA_IG with relatively more Excess Count. This is mostly because of influence of the sample on the reduct. To circumvent the problem of Excess Count, a non persistent strategy of using a new stratified sample, generated with replacement, for selection of each

Figure 4.16: Reduct Length Comparison of Comparative Results with SARA



attribute is developed. NPSS_IQRAIG algorithm uses non persistent strategy using stratified sampling. Empirical analysis conducted using large decision systems have proved the utility of NPSS_IQRAIG in obtaining better computational gains along with ability to reduce Excess Count.

Both persistent and non persistent approaches embed global validation step, which is empirically demonstrated to be computationally feasible. Global validation has given the proposed algorithms ability to find super reduct overcoming the serious limitation in knowledge representation due to sub reduct obtained by existing sampling approaches. Compared to SARA algorithm, proposed approaches uses much fewer number of sample datasets.

Chapter 5

Reduct Computation in Incomplete Decision Systems

The methods developed for reduct computation in chapters 3 and 4 are for complete symbolic decision systems (CDS). The classical rough sets work under the assumption that the decision system contain complete information. But many real world decision systems have imprecision and uncertainty. One primary occurrence of imprecision in decision tables is because of missing values others being erroneous values, possibility distribution or probability distribution etc. The present study confines to imprecision arising out of decision tables containing missing values which are referred as incomplete decision systems (IDS). The current chapter deals with arriving at an efficient reduct computation algorithm for incomplete decision systems using a novel granular framework.

Section 5.1 gives the review of reduct computation approaches in IDS using rough sets and extensions to rough sets. Kryszkiewicz's extension of rough sets for IDS and limitation of existing approaches for reduct computation in IDS are described in Section 5.2. Section 5.3 presents the proposed novel granular framework, algorithm RefineGranularSpace and new formulas for lower approximation, positive region, conditional information entropy using granular framework. Algorithm IQRAIG_Incomplete, proposed algorithm for reduct computation in IDS, is given in Section 5.4 along with theoretical validation of positive region removal incorporated in IQRAIG_Incomplete. Experiments and results using benchmark IDS, comparative experiments with IFSPA-IPR, IPR algorithms are given in Section 5.5 and analysis of results is presented in Section 5.6.

5.1 Introduction and Literature Review

An Incomplete Information System (IIS) or Incomplete Decision System (IDS) has objects with missing attribute values. One of the approaches for dealing with IIS/IDS is imputation of missing value to convert the system into CIS/CDS respectively. Several approaches for imputing missing value are described in [36]. The simplest of these approaches is removal of incomplete objects (objects with

missing attribute values), not preferred in practice, as it leads to losing available information in incomplete objects. There are methods for imputation of missing value by using probabilistic approaches [84, 87]. But the performance of the induced system using the obtained CIS/CDS depends highly on the preprocessing strategy adopted in imputation methodology.

Kryszkiewicz [66] proposed a similarity relation for IIS, which allows to compute reduct and rule induction without performing imputation of missing value. In this approach, missing value is called as null value and symbolized as “*”. Here the missing value of an attribute for an object is assumed to be from the domain of the attribute, but is unknown and indiscernible (equal) to all possible attribute domain values. Variations are proposed for similarity relation [107] that treat missing value as don’t care condition (equivalent to null value), a lost value (not related to any other domain value). In [132, 148] extensions are proposed for IIS based on limited tolerance relation, where a known attribute value is indiscernible with missing value but converse is not true.

Approaches for finding reduct for IIS/IDS can be broadly categorized into three ways, namely 1) Method of possible worlds, 2) Discernibility Matrix based approaches, 3) Information Entropy or Positive region based heuristic algorithms. Method of possible worlds [85] constructs several possible CDS for a given IDS and using conventional rough set methods, reducts for all possible CDS are evaluated and aggregated. This approach is computationally expensive and the similar method of possible equivalence relations given in [85] is also computationally expensive. Approaches for finding reduct in IIS/IDS using Discernibility Matrix are given by [66, 68, 176]. In [68] an improvised approach compared to method in [66] is given for computation of Discernibility matrix for IIS using Maximal consistent blocks. Recently Zou et al. [176] have given more efficient method for computing Reduct using Discernibility matrix approach. Discernibility matrix based approaches involve comparison of all possible pair of objects of IIS/IDS. Compared with discernibility matrix based approaches, positive region or Information Entropy based heuristic algorithms are computationally efficient. Approaches for Reduct in IIS/IDS using Information Entropy are given in [59, 70, 137, 170].

Some of the positive region based heuristic algorithm for reduct calculation in IIS/IDS are [82, 112, 171]. In [171] a method for handling hybrid data with missing values using neighborhood granulation, dealing with missing values with don’t care and lost conditions in finding reduct, is given. The approach in [82] uses maximal tolerance classes for reducing computations. Qian et al. [112] proposed an efficient algorithm for reduct computation in IDS which incorporates Positive region removal (Positive acceleration). Both approaches in [82, 112] treat the similarity class of each object in IDS in isolation. Present chapter gives a novel granular framework for reducing computations in finding reduct for IDS which

incorporates a mechanism to compute similarity classes of a group of objects simultaneously, which makes finding positive region efficient. The methodology & philosophy of CDS reduct computation algorithm IQRA_IG given in Chapter 3 has been modified and tuned to arrive at reduct computation algorithm for IDS, which is named as IQRAIG_incomplete.

Next Section 5.2 describes the Kryszkiewicz's extension [66] of rough sets for dealing with incomplete information systems which is used in the current work.

5.2 Extension of Rough Sets for Incomplete Decision Systems

$DT = (U, C \cup \{d\}, (V_a, f_a)_{a \in C \cup \{d\}})$ represents an Incomplete Decision system (IDS) where U is a set of objects, C is the set of conditional attributes and d is the decision attribute. For each attribute $a \in C \cup \{d\}$, V_a is called the value set of a and f_a is a mapping from U to V_a assigning a value for each object of U for attribute a . The missing attribute values in IDS are denoted by a character *. Without loss of generality, it is assumed that the decision attribute d does not contain missing values i.e., for all objects the decision class is known.

In this work the approach used by [66] for the treatment of missing value is followed. Missing values are called as null values and as their value is unknown, it is assumed that they are equal to any other possible value in the domain. Similar to indiscernibility relation in complete decision or information systems, a similarity relation is used for incomplete decision or information systems. Given $A \subseteq C$, $SIM(A)$ [66] is a similarity relation defined on U using attributes A where

$$SIM(A) = \{(x, y) \in U \times U / \forall a \in A, f_a(x) = f_a(y) \vee f_a(x) = * \vee f_a(y) = *\} \quad (5.1)$$

$SIM(A)$ is a reflexive, symmetric relation and is not always a transitive relation and so it is a tolerance relation. The similarity class of an object x is denoted by $S_A(x)$ and is defined as,

$$S_A(x) = \{y \in U / (x, y) \in SIM(A)\} \quad (5.2)$$

The collection of similarity classes for objects in U form a cover for U . Let $X \subseteq U$ and $A \subseteq C$. The lower approximation of a concept $X \subseteq U$, denoted by $\underline{AX}$ using the knowledge of $SIM(A)$, is defined by set of objects whose similarity class is completely contained in X , i.e.,

$$\underline{AX} = \{x \in U / S_A(x) \subseteq X\} \quad (5.3)$$

The upper approximation of X , denoted by $\overline{AX}$, is defined as containing objects having their similarity class having non trivial intersection with X . Formally,

$$\overline{AX} = \{x \in U / S_A(x) \cap X \neq \emptyset\} \quad (5.4)$$

Extending the positive region definition for Incomplete decision systems as the union of lower approximations of all decision equivalence classes, the formula becomes,

$$POS_A(\{d\}) = \bigcup_{X \in U/\{d\}} \underline{AX} \quad (5.5)$$

where $U/\{d\}$ is the set of equivalence classes induced by the standard indiscernibility relation IND as it is assumed that decision attribute d has no missing values. The dependency measure gamma, giving the fraction of number of positive region objects out of U , denoted by $\gamma_A(\{d\})$, is defined as,

$$\gamma_A(\{d\}) = \frac{|POS_A(\{d\})|}{|U|} \quad (5.6)$$

where $|S|$, for any set S , denote the size of S .

Reduct for IDS is same as reduct for CDS using the new definitions of positive region and lower approximation. A set of attributes $R \subseteq C$ is a Reduct for IDS, represented as DT, if and only if $\gamma_R(\{d\}) = \gamma_C(\{d\})$ such that for any proper subset $Q \subset R$, $\gamma_Q(\{d\}) < \gamma_R(\{d\})$.

5.2.1 Limitations of Existing Approaches

There is a significant impact on computations because of the type of relation used for obtaining positive region. Indiscernibility relation is used in applying Rough sets for CDS having only symbolic attributes [97]. Similarity relation is used in IDS [66].

Consider the example CDS given in Table 5.1. Consider the approximation space (U, IND_B) where $B = \{a\}$. Consider the equivalence class $\{O1, O3, O4\} \in U/B$. $B(O1) = B(O2) = B(O3) = \{O1, O2, O3\}$ as IND_B is an equivalence relation. The equivalence class is shared by all its members in classical rough sets. Since objects of $\{O1, O3, O4\}$ belong to the same decision class, all members of the equivalence class belong to positive region as the equivalence class is shared

Table 5.1: Example of CDS

U	a	d
O1	5	1
O2	6	2
O3	5	1
O4	5	1
O5	6	2
O6	7	2
O7	6	2

Table 5.2: Example of IDS

U	b	d
O1	5	1
O2	6	2
O3	5	1
O4	5	1
O5	6	2
O6	*	2
O7	*	2

by all its members.

Consider the example IDS given in Table 5.2. Consider the approximation space (U, SIM_B) where $B = \{b\}$. Consider the similarity class $\{O2, O5, O7, O8\}$. Here, $S_B(O2) = S_B(O5) = \{O2, O5, O7, O8\}$. Hence in the approximation space defined for IDS, a similarity class is shared by some of its members. All objects of $\{O2, O5, O7, O8\}$ belong to the same decision class. But unlike CDS, only objects sharing the similarity class, i.e., $O2, O5$ belong to positive region.

The existing approaches for reduct finding in IDS do not use this sharing of similarity class property to reduce the computations. This aspect in existing approaches makes reduct finding in IDS having higher complexity than reduct computation in CDS. Hence in this chapter a novel granular framework is proposed for IDS that embed the objects sharing the similarity class within the structure of the granule with the objective of reducing the complexity of reduct computation in IDS. Next section describes the proposed granular framework.

5.3 Novel Granular Framework for Incomplete Decision Systems

In granular computing, a granule refers to a set of objects which are similar to each other by some criteria. In IDS, similarity classes form the granules. If the

property of some objects sharing the same similarity class is embedded into the granular structure then it has impact on reducing the computations, as the same computations need not be done with respect to individual objects. In any positive region based heuristic reduct algorithm the time complexity depends upon efficient computation of positive region which in turn depends on efficient computation of lower approximation for each of the decision classes. The definition of lower approximation in IDS (Eq. 5.3) is based on computing with the similarity class of each object separately. As the similarity class is shared by some objects, if the structure of the granule allows to find the associated objects then the repeated calculations with each one of them can be avoided. With this motivation following granular framework is constructed for IDS. Next sub section 5.3.1 gives the properties of proposed granular framework for Incomplete Decision Systems.

5.3.1 Properties

Granular space $Gr(U, A)$ formed using attribute set $A \subseteq C$ on universe of objects U consists of distinct similarity classes induced on U by $SIM(A)$. By distinct similarity classes, it is meant, all different similarity classes without repetition. A distinct similarity class is referred as granule.

A granule $g \in Gr(U, A)$ is defined as, a collection of objects i.e., $g \subseteq U$ such that if $x, y \in g$ then $x \in S_A(y)$ and $y \in S_A(x)$. Granule g is further partitioned into two portions *base*, *tail* where,

$$\begin{aligned} base &= \{x \in g / S_A(x) = g\} \\ tail &= g - base \end{aligned} \tag{5.7}$$

As the similarity class is shared by some of the objects belonging to the similarity class, *base* portion of a granule is used to specify the set of objects sharing the similarity class and the remaining objects in the similarity class are kept in *tail* portion. The notation $g.base$, $g.tail$ is used for referring to the *base* and *tail* portions of g respectively.

As every object $x \in U$ has a unique similarity class there exists an unique granule $g \in Gr(U, A)$ such that $x \in g.base$. As distinct similarity classes are overlapping, there may exist more than one granule containing x in their tail portion. Hence the following properties follow,

Property 1: For any two distinct granules $g, f \in Gr(U, A)$, $g.base \cap f.base = \phi$.

Property 2: $U = \bigcup_{g \in Gr(U, A)} g.base$

Using properties 1, 2 it follows that the *base* portions of all granules in $Gr(U, A)$ constitute a partition for U .

For formulating a reduct algorithm for IDS using the structure and philosophy of IQRA_IG algorithm, the following three operators are introduced.

1. Granular Space Refinement (GrSR): Given $Gr(U, A)$ deriving a new granular space $Gr(U, A \cup \{a\})$. For reflecting the changes in the existing granular space when an attribute $a \in C - A$ is included in A .
2. Positive Region of $Gr(U, A)$ (GrPOS): Computation of Positive Region given granular space $Gr(U, A)$, thereby enabling computation of gamma measure $\gamma_A(\{d\})$ using Eq. 5.6.
3. Information Entropy of $Gr(U, A)$ (GrEnt): Computation of Information Entropy of a granular space $Gr(U, A)$ for dealing with trivial ambiguous situations using Information Gain heuristic.

Next sub section gives the proposed procedure for obtaining Granular Refinement of $Gr(U, A)$ in the form of algorithm RefineGranularSpace.

5.3.2 Granular Refinement

In IQRA_IG, the recommended reduct is initialized to empty set and in every iteration a single attribute is included into recommended reduct. Hence the state of granular space when $A = \phi$ is to be defined and is given in Property 3.

Property 3: Granular space $Gr(U, A)$ when $A = \phi$ consists of a single granule, i.e., $Gr(U, \phi) = \{g\}$ where $g.base = U$ and $g.tail = \phi$.

Property 3 in other words says that in the absence of knowledge about the objects, all objects are similar to each other.

Given $A \subset C$ and an attribute $a \in C - A$ the granular refinement operator GrSR constructs the refined granular space, i.e.,

$$Gr(U, A \cup \{a\}) = GrSR(Gr(U, A), \{a\})$$

The GrSR operator is abstractly described here.

Let $g \in Gr(U, A)$. The refined granules $GrSR(g, \{a\})$ are evolved as given below.

1. Let $g/IND(\{a\}) = \{P_1 \dots P_r\}$ with P^* denoting the partition corresponding to missing value if exists.
2. $GrSR(g, \{a\}) = \{g'_i\}_{i=1}^r$ where, $g'_i = \phi$ in case $P_i \cap g.base = \phi$ otherwise

$$g'_i.tail = \begin{cases} g'_i.base = P_i \cap g.base \\ (P_i \cap g.tail) \cup P^* \text{ if } P_i \neq P^* \\ g - g'_i.base \text{ otherwise} \end{cases} \quad (5.8)$$

Algorithm RefineGranularSpace describes the procedure for granular refinement of $Gr(U, A)$ into $Gr(U, A \cup \{a\})$.

Algorithm 9: RefineGranularSpace

Input : Incomplete Decision System $DT = (U, C \cup \{a\}, (V_a, f_a)_{a \in C \cup \{a\}})$
Granular Space $Gr(U, A)$ where $A \subset C$
 a : New attribute to be included into A where $a \in C - A$
Output: Refined Granular Space $Gr(U, A \cup \{a\})$

```

1  $Gr(U, A \cup \{a\}) = \emptyset$ 
2 foreach granule  $g \in Gr(U, A)$  do
3    $[base1, \dots, basek] = \text{getEquivalenceClasses}(g.base, DT(g.base, a))$ 
4    $IncompleteObjects = \text{getIncompleteObjects}(g.tail, DT(g.tail, a))$ 
5    $g.tail = g.tail - IncompleteObjects, Startbase = base1$ 
6   if  $DT(base1(1), a) = *$  then
7     Add new granule  $g^*$  into  $Gr(U, A \cup \{a\})$  where
8      $g^*.base = base1$ 
9      $g^*.tail = (g.base - base1) \cup g.tail \cup IncompleteObjects$ 
10     $IncompleteObjects = IncompleteObjects \cup base1, Startbase = base2$ 
11  end
12  for  $b \leftarrow Startbase$  to  $basek$  do
13     $tailportion = \text{find}(DT(g.tail, a) = DT(b(1), a))$ 
14    Add new granule  $g'$  into  $Gr(U, A \cup \{a\})$  where
15     $g'.base = b$ 
16     $g'.tail = tailportion \cup IncompleteObjects$ 
17     $g.tail = g.tail - tailportion$ 
18  end
19 end
20 Return  $Gr(U, A \cup \{a\})$ 

```

In algorithm RefineGranularSpace a pseudo matlab scripting notation is employed. As DT is a two dimensional array, the notation $DT(\text{objects}, \text{attribute})$ refers to the ‘attribute’ values for ‘objects’ in DT. The method `getEquivalenceClasses` result in the equivalence classes for the ‘objects’ specified in first argument using ‘values’ given in the second argument by using the standard indiscernibility relation IND. In obtaining these equivalence classes the missing value “*” is also treated as a distinguishable value in forming the partition. The function ‘getMissingPortion’ select the objects from the first argument having missing value “*” by using the corresponding values provided in the second argument. The first object of the base portion is referenced using the notation ‘base(1)’. find operation returns the objects satisfying the condition specified as the argument.

With the addition of new attribute into A , the similarity class of each object is refined. The base portion of a granule $g \in Gr(U, A)$ contain the set of objects

sharing the similarity class $g.base \cup g.tail$. The base portions of granules getting added into $Gr(U, A \cup \{a\})$, based on granule g , are going to form a partition of $g.base$. Hence,

$$g.base = \bigcup_{g' \in Gr(U, A \cup \{a\}) \wedge g'.base \subseteq g.base} g'.base \quad (5.9)$$

So, in algorithm `RefineGranularSpace` for every granule $g \in Gr(U, A)$, new granules are added into the refined granular space $Gr(U, A \cup \{a\})$. Equation 5.9 is used to form the base portions and the tail portions for each of them are identified accordingly by picking up the objects in $g.tail$ containing same value for attribute a and also includes those objects having missing value for attribute a . Special treatment is taken for the sub granule g^* from g which refers to the missing values from attribute a . The tail portion of g^* will contain all the remaining objects of the granule g . The refined granular space $Gr(U, A \cup \{a\})$ formed using this procedure satisfies all the properties of granular space detailed in the earlier section.

Based on earlier discussion and from Eq. 5.9, there exists a similarity between standard IND based refinement and the proposed granular refinement in IDS. The refinement that is happening to base portions of the granules is exactly the same refinement that occurs in the IND based refinement in CDS using the missing value “*” as a special discernible domain value. Hence the following property is valid in proposed granular refinement.

Property 4: The cardinality of $Gr(U, A)$ is equal to the cardinality of equivalence classes of $IND(A)$ treating the missing value “*” as a special discernible domain value.

Property 5, given below, summarizes the granular refinement and uses the notation $\leq$ for representing the granular refinement property between two granular spaces.

Property 5: Let $A \subset B \subseteq C$. The granular space $Gr(U, B)$ is said to be a granular refinement of $Gr(U, A)$ represented as $Gr(U, B) \leq Gr(U, A)$, satisfying the property that for every $g' \in Gr(U, B)$ there exists an unique granule $g \in Gr(U, A)$ such that $g'.base \subseteq g.base$.

Algorithm `RefineGranularSpace` assures of satisfying Property 5 and hence the following property.

Property 6: Let $A \subset C$ and let $a \in C - A$. If $Gr(U, A \cup \{a\})$ is obtained by applying algorithm `GranularRefineSpace` on $Gr(U, A)$ then $Gr(U, A \cup \{a\}) \leq Gr(U, A)$.

The next sub section 5.3.3, gives a new definition for positive region using the proposed novel granular framework.

5.3.3 Positive Region using Granular Framework

The positive region definition given in Eq. 5.5 which in turn depends on the lower approximation definition given in Eq. 5.3 uses object wise computations. Having obtained a granular framework for IDS this section gives a new definition of positive region which is granular based.

Given $A \subseteq C$, $X \subseteq U$, the new definition for lower approximation of X using $Gr(U, A)$ becomes,

$$\underline{AX} = \bigcup_{g \in Gr(U, A) \wedge (g.base \cup g.tail) \subseteq X} g.base \quad (5.10)$$

The positive region definition given in Eq. 5.5 will become new definition for positive region where $\underline{AX}$ is obtained using Eq. 5.10. At implementation level, to avoid many subset finding operations the following definition for positive region is given which is in line with the definition of positive region given in Chapter 3. The definition of positive region which has lesser practical complexity is,

$$POS_A(\{d\}) = \bigcup_{g \in Gr(U, A) \wedge |(g.base \cup g.tail)/IND(\{d\})|=1} g.base \quad (5.11)$$

Hence positive region using $Gr(U, A)$ is the union of base portions of those granules g having the same decision class for objects in $g.base \cup g.tail$.

From now onwards, in this chapter, whenever the gamma measure $\gamma_A(\{d\})$ is used it refers to obtaining gamma using Eq. 5.6 where positive region is computed using Eq. 5.11.

Next sub section 5.3.4 describes a new way of computing Information Entropy of a given granular space for an Incomplete Decision System.

5.3.4 Information Entropy of $Gr(U, A)$ in IDS

The conditional information entropy for a set of attributes $A \subseteq C$ is defined for Incomplete Decision System as [137]

$$H(A, \{d\}) = \sum_{i=1}^{|U|} H_{u_i}(A, \{d\}) \quad (5.12)$$

where for each object $u_i \in U$, $H_{u_i}(A, \{d\})$ is defined as [137]

$$H_{u_i}(A, \{d\}) = -\frac{1}{|U|} \sum_{j=1}^m \frac{|S_A(u_i) \cap X_j|}{|S_A(u_i)|} \log \frac{|S_A(u_i) \cap X_j|}{|S_A(u_i)|} \quad (5.13)$$

where for each j , X_j denote the equivalence class induced by decision attribute d , i.e., $U/IND(\{d\}) = \{X_1, \dots, X_m\}$.

Similar to positive region computation, existing information entropy computations for Incomplete decision systems depend on object wise computations leading to lot of redundancy in computations. Using the proposed granular framework an efficient way of computing information entropy is arrived.

Given $Gr(U, A)$ let a granule $g \in Gr(U, A)$. Then the following property follows from the definition of granule.

$$S_A(x) = (g.base \cup g.tail) \quad \forall x \in g.base \quad (5.14)$$

Hence it follows that

$$H_x(A, \{d\}) = H_y(A, \{d\}) \quad \forall x, y \in g.base \quad (5.15)$$

Using Eq. 5.15, conditional information entropy of a granule $g \in Gr(U, A)$ can be defined and is given below.

$$H_g(A, \{d\}) = -\frac{|g.base|}{|U|} \sum_{j=1}^m \frac{|S_A(g) \cap X_j|}{|S_A(g)|} \log \frac{|S_A(g) \cap X_j|}{|S_A(g)|} \quad (5.16)$$

where $S_A(g) = (g.base \cup g.tail)$. Now the equivalent definition of conditional information entropy given in Eq. 5.12 is given using granular space $Gr(U, A)$ as,

$$H(Gr(U, A), \{d\}) = \sum_{g \in Gr(U, A)} H_g(A, \{d\}) \quad (5.17)$$

Comparing Eq. 5.16 with Eq. 5.13 it is obvious to see that several repeated computations are avoided in the new formulation. Equations 5.17, 5.16 are used to arrive at IG heuristic in the proposed algorithm for reduct computation in IDS using IQRA_IG semantics.

Having obtained the required procedure for granular refinement and new way of finding positive region, gamma measure, conditional information entropy next section 5.4 gives the proposed reduct algorithm IQRAIG_Incomplete using the granular framework.

5.4 IQRAIG_Incomplete: Reduct algorithm using Granular Framework

The proposed reduct finding algorithm for Incomplete Decision Systems follows the structure and philosophy of IQRA_IG algorithm. The recommended reduct R is initialized to empty set and corresponding granular space $Gr(U, R)$ is defined using Property 3. In every iteration the current granular space is refined using every available attribute $a \in C - R$ using algorithm RefineGranularSpace. For each refined granular space $Gr(U, R \cup \{a\})$, positive region is computed using Eq. 5.11 and gamma measure $\gamma_{R \cup \{a\}}(\{d\})$ is computed using Eq. 5.6. Similar to IQRA_IG algorithm, that attribute giving the maximum gamma gain is included into recommended reduct R . In case of trivial ambiguous situation, Information Entropy of refined granular space is computed using Eq. 5.17 and the attribute which gives maximum information gain is included into recommended reduct R . Similar to IQRA_IG algorithm, positive region removal is applied before proceeding to the next iteration. The resulting algorithm is named as IQRAIG_Incomplete and is given in the next sub section 5.4.1.

5.4.1 Algorithm IQRAIG_Incomplete

Algorithm IQRAIG_Incomplete is suitable for both consistent and inconsistent incomplete decision systems as the end condition is based on obtaining $\gamma_C(\{d\})$ using reduct attributes. Hence the algorithm is applicable whether $\gamma_C(\{d\})$ is 1 (for consistent IDS) or less than 1 (for inconsistent IDS).

Algorithm 10: IQRAIG_Incomplete

Input : Incomplete Decision System $DT = (U, C \cup \{d\}, (V_a, f_a)_{a \in C \cup \{d\}})$
Output: Recommended Reduct: R

```
1 Compute  $Gr(U, C)$ ,  $POS_C(\{d\})$  and  $\gamma_C(\{d\})$ 
2 Set  $R = \emptyset$ 
3 Set  $Gr(U, R)$  as  $\{g\}$  where  $g.base = U$  and  $g.tail = \emptyset$ 
4 Set  $\gamma_R(\{d\}) = 0$ 
5  $POSCOUNT = 0$  /* for denoting number of positive region objects */
6  $DT'(U', C \cup \{d'\}) = DT$  /* For differentiating reduced decision system and original decision system. */
7 while  $\gamma_R(\{d\}) < \gamma_C(\{d\})$  do
8    $T = R$ ,  $\gamma_T(\{d'\}) = 0$  /*  $d'$  indicates computations being done using  $DT'$  */
9   foreach  $a$  in  $C - R$  do
10      $Gr(U', R \cup \{a\}) = RefineGranularSpace(DT', Gr(U', R), a)$ 
11     Compute  $POS_{R \cup \{a\}}(\{d'\})$ ,  $\gamma_{R \cup \{a\}}(\{d'\})$ 
12     if  $\gamma_{R \cup \{a\}}(\{d'\}) > \gamma_T(\{d'\})$  then
13        $T = R \cup \{a\}$ 
14     end
15   end
16   if  $T = R$  then
17     foreach  $a$  in  $C - R$  do
18       Compute  $H(Gr(U', R \cup \{a\}))$ 
19       if  $H(Gr(U', R \cup \{a\})) > H(Gr(U', T), \{d'\})$  then
20          $T = R \cup \{a\}$ 
21       end
22     end
23      $R = T$ 
24   end
25   else
26      $R = T$ ,  $POSCOUNT = POSCOUNT + |POS_R(\{d'\})|$ 
27      $\gamma_R(\{d\}) = \frac{POSCOUNT}{|U|}$ ,  $DT' = DT'(U' - POS_R(\{d'\}), C \cup \{d'\})$ 
28      $Gr(U', R) = cleanGranularSpace(Gr(U', R))$ 
29   end
30 end
31 Return  $R$ 
```

In next two sub sections, two important aspects of IQRAIG_Incomplete are discussed namely computation of $Gr(U, C)$ as in line 1 of Algorithm 10 and validation of positive region removal step in IDS as done in step 27.

5.4.2 Computation of $Gr(U, C)$

The end condition of IQRAIG_Incomplete requires computation of $Gr(U, C)$ so that $POS_C(\{d\})$ can be computed with which required $\gamma_C(\{d\})$ is obtained. There are two ways to compute $Gr(U, C)$. The first method is a sequential approach and the second method is global approach.

In sequential approach starting with $Gr(U, \phi)$, the granular space is refined using algorithm RefineGranularSpace on adding one conditional attribute at a time sequentially. In global approach, initially the equivalence classes of $IND(C)$ are obtained using stable quick sort algorithm assuming missing value “*” is a special discernible value with respect to other domain values. These equivalence classes form the base portions of the granules of $Gr(U, C)$. The tail portions

of each granule are identified by comparing, using missing value semantics of similarity relation, only with the attribute values of base portion of other granules. The global approach is computationally efficient as long as number of granules of $Gr(U, C)$ is much smaller than number of objects of the dataset otherwise the sequential method is preferred.

A heuristic is adopted based on experimental analysis for determining which method to be followed in IQRAIG_Incomplete algorithm. The heuristic is to use global approach for computing $Gr(U, C)$ whenever $|U/IND(C)| \leq \frac{|U|}{3}$ otherwise use sequential approach to compute $Gr(U, C)$.

5.4.3 Validation of Positive Region Removal

The computational efficiency of IQRA_IG over QRA is primarily due to the incorporation of positive region removal aspect. IQRAIG_Incomplete algorithm also contains this step as it is based on the structure of IQRA_IG. In IQRA_IG and IQRA related algorithms removal of positive region granules does not affect the future computations as the granules are non overlapping being equivalence classes arising out of IND relation. But the granules formed in IQRAIG_Incomplete algorithms are based on tolerance relation SIM and hence the granules are possibly overlapping. Hence the obtained positive region objects in an iteration of IQRAIG_Incomplete can be present in other granules in non positive region. Hence validation is required to establish that the positive region objects removal does not affect, in future iterations of IQRAIG_Incomplete, the computations of other granules. The theory for validation of positive region removal is given here in the form of lemmas and theorem.

Lemma 5.4.1. *Let $Gr(U, A)$ be the granular space induced by attribute set $A \subseteq C$. Let $g, g' \in Gr(U, A)$ be two distinct granules. If $g.base \subseteq g'.tail$ then $g'.base \subseteq g.tail$.*

Proof. Let an object $o \in g'.base$. Since $g.base \subseteq g'.tail$, $\forall x \in g.base$, $x \in S_A(o)$. Since, $SIM(A)$ is symmetric it follows, $\forall x \in g.base$, $o \in S_A(x)$. Hence, $o \in g.tail$.

$\therefore g'.base \subseteq g.tail$. This concludes the proof.

□

Importance of Lemma 5.4.1 is that the base objects of a granule g are present only in those granules whose base portions are part of $g.tail$. Next lemma proves the monotonic behavior of Positive Region as iterations progress in IQRAIG_Incomplete algorithm.

Lemma 5.4.2. *Let $Gr(U, A)$ be the granular space induced by attribute set $A \subseteq C$. For any attribute $a \in C - A$, $POS_A(\{d\}) \subseteq POS_{A \cup \{a\}}(\{d\})$.*

Proof. Let an object $o \in POS_A(\{d\})$. Then there exists a unique granule $g \in Gr(U, A)$ such that $o \in g.base$ and by definition of positive region, $S_A(o) = (g.base \cup g.tail)$ is subset of some $X \in U/IND(\{d\})$. Using granular refinement property given in Property.5, there exists a unique granule $g' \in Gr(U, A \cup \{a\})$ such that $o \in g'.base$ and $g'.base \subseteq g.base$ and $g'.tail \subseteq (g.base \cup g.tail)$.

Hence, $S_{A \cup \{a\}}(o) = (g'.base \cup g'.tail) \subseteq (g.base \cup g.tail) = S_A(o)$. It follows that $S_{A \cup \{a\}}(o) \subseteq S_A(o) \subseteq X$. So, $o \in POS_{A \cup \{a\}}(\{d\})$.

$\therefore POS_A(\{d\}) \subseteq POS_{A \cup \{a\}}(\{d\})$. This concludes the proof. □

The validity of positive region removal in IQRAIG_Incomplete, using Lemmas 5.4.1, 5.4.2, is established in the following theorem.

Theorem 5.4.3. *Let $DT = (U, C \cup \{d\}, (V_a, f_a)_{a \in C \cup \{d\}})$ be an incomplete decision system. Let $DT' = (U', C \cup \{d'\}, (V_a, f_a)_{a \in C \cup \{d'\}})$ be the reduced decision system owing to removal of positive region objects after an iteration of IQRAIG_Incomplete algorithm when an attribute $a \in C - R$ is added into recommended reduct R . Then,*

$$POS_{R \cup \{a\}}(\{d\}) = POS_R(\{d\}) \cup POS_{R \cup \{a\}}(\{d'\})$$

Proof. Using Lemma 5.4.2, $POS_R(\{d\}) \subseteq POS_{R \cup \{a\}}(\{d\})$. If B denote the set of objects included into positive region because of addition of 'a' into R then, $B = POS_{R \cup \{a\}}(\{d\}) - POS_R(\{d\})$. Let $o \in B$. Then two cases are possible.

Case I: Similarity class of 'o' is disjoint with $POS_R(\{d\})$, i.e., $S_{R \cup \{a\}}(o) \cap POS_R(\{d\}) = \phi$. In which case the computation for positive region belongingness of object 'o' is not affected by presence or absence of $POS_R(\{d\})$. The computations in DT' for positive region calculation of 'o' are as same as computing in DT .

Case II: Similarity class of 'o' is having non trivial intersection with $POS_R(\{d\})$, i.e., $S_{R \cup \{a\}}(o) \cap POS_R(\{d\}) \neq \phi$. Let $o_p \in S_{R \cup \{a\}}(o) \cap POS_R(\{d\})$. Since $o_p \in S_R(o)$ ($\because S_{R \cup \{a\}}(o) \subseteq S_R(o)$) and also that $o \in POS_{R \cup \{a\}}(\{d\})$, $o \notin POS_R(\{d\})$ there exists two distinct granules $g, g' \in Gr(U, R)$ such that $o_p \in g.base$ and $o \in g'.base$. Hence g is a positive region granule and g' is a non positive region granule in $Gr(U, R)$. Since, $o_p \in S_R(o)$, $o \in S_R(o_p) = (g.base \cup g.tail)$ and it follows that $o \in g.tail$. Since g is a positive region granule we have $(g.base \cup g.tail) \subseteq X$ for some $X \in U/IND(\{d\})$. Hence the decision classes of o and o_p are same. Hence in computing the positive region membership of o in DT' is not affected by removal of

objects belonging to the same decision class. That is as $o \in B \subseteq POS_{R \cup \{a\}}(\{d\})$ we have $S_{R \cup \{a\}}(o) \subseteq X$ and presence of objects like o_p belonging to same decision class lead to no contradiction.

Hence calculating B objects as part of $POS_{R \cup \{a\}}(\{d\})$ is with or without the $POS_R(\{d\})$ objects being present in DT. If $POS_R(\{d\})$ objects removed from DT resulting in reduced decision table DT' the $B = POS_{R \cup \{a\}}(\{d'\})$. This concludes the proof. □

5.4.4 Complexity Analysis of IQRAIG_Incomplete

Time and space complexities of IQRAIG_Incomplete are same as that of IQRA_IG algorithm excepting linear overheads in time complexities because of *tail* computation in RefineGranularSpace, CleanGranularSpace.

5.5 Experiments and Results

In this section the experiments are divided into two parts. In the first part, given in sub section 5.5.1, proposed algorithm is applied on benchmark IDS to analyze the importance of primary and secondary heuristics in dealing with trivial ambiguous situations and also to analyze the classification performance resulting from reduct obtained using IQRAIG_Incomplete. In the second part, given in sub section 5.5.4, a comparative analysis of IQRAIG_Incomplete is done with IFSPA-IPR, IPR algorithms given in [112] following the experimental setup given in [112]. Two derived versions of the proposed IQRAIG_Incomplete are used in these experiments. IQRAIG_Incomplete without the positive region removal aspect is named as QRAIG_Incomplete, legible as the structure of the algorithm without positive region removal is that of QRA still dealing with trivial ambiguous situations with IG heuristic. Similarly IQRAIG_Incomplete algorithm without dealing with trivial ambiguous situations with secondary heuristic IG is named IQRA_Incomplete. These extensions are used for understanding the impact of these aspects of positive region removal, dealing with trivial ambiguous situations in the experimental analysis.

Table 5.3: Details of Incomplete Decision Systems

Dataset	Objects	Conditional Attributes	Decision Classes	Missing Values Count
Mushroom	8124	22	2	2480
Vote	435	16	2	392
Lung Cancer	32	56	3	5
Backup-large-test	376	35	19	1625
Shuttle	43500	9	7	6042
Adult	32561	14	2	4262
Hypothyroid	2108	25	2	3532
Audiology Standardized	200	69	24	291
Soybean-large	307	35	19	712
Dermatology	366	34	6	8
Breast Cancer Wisconsin	699	9	2	16

Table 5.4: IQRAIG_Incomplete Results

Dataset	Reduct	CT (in Seconds)	$\gamma_C(\{d\})$
Mushroom	5	7.76	1.0
Vote	12	5.46	0.326
Lung Cancer	5	0.48	1.0
Backup-large-test	11	7.48	0.907
Shuttle	4	86.20	1.0
Adult	8	185.76	0.999
Hypothyroid	4	7.26	1.0
Audiology Standardized	21	3.94	1.0
Soybean-large	12	7.39	0.928
Dermatology	7	2.10	1.0
Breast Cancer Wisconsin	5	0.65	1.0

5.5.1 Experiments on benchmark Incomplete Decision Systems

The details of the benchmark incomplete decision systems used for experimentation are given in Table 5.3. Along with the number of objects, conditional attributes, and number of decision classes, the number of missing values present in the IDS is specified in Table 5.3. Applying IQRAIG_Incomplete algorithm on these datasets, the results obtained are given in table 5.4. Table 5.4 along with giving reduct length and computational time (CT) contains the gamma measure for all conditional attributes $\gamma_C(\{d\})$.

5.5.2 Classification Analysis using 3NN and Stability Analysis

The main objective of the proposed work is for obtaining reduct without going for the imputation of missing value. With similar objective to obtain a classifier which can work with IDS without imputation or neglecting or assuming as a

discernible value of the missing values, a literature survey is conducted. The literature survey has not resulted in identifying a classifier dealing with missing values without doing imputation as preprocessing step. Weka [34] provides only imputation of missing values using “ReplaceMissingValues”, imputing with mean and mode for qualitative and quantitative attributes respectively, before using the IDS for building classifier models. Similar approach is used in Rough Set Exploration System (RSES) [4]. Hence the following procedure has been adopted for building a classifier using IDS.

As KNN or specifically 3NN is depending on computing distance (usually cartesian distance) between the test object attribute vector with train data objects, the symbolic attribute needs to be converted suitable for computing distance. One procedure usually followed is to convert a symbolic attribute $a \in C$ into $|V_a|$ number of boolean attributes such that each boolean attribute corresponds to one domain value of attribute ‘a’. In converting IDS in similar lines, in case the attribute $a \in C$ is having ‘*’ in its domain of values then the attribute is converted only to $|V_a| - 1$ number of boolean attributes such that for object having missing value for attribute ‘a’ all $|V_a| - 1$ column values are set to 1. This in accordance to the semantic of missing value that it is equal to all other domain values. In computing distance the missing value is equi distant from all other objects having other domain values of ‘a’ and hence in computing distance the other known values of the object makes an important contribution. This is much better representation for missing value than imputing missing value to one particular domain value or totally discarding the incomplete object. Every IDS is converted into CDS as described and 3NN classifier is applied on the dataset. The results of tenfold analysis performed using this conversion and average results of applying 3NN classifier is given in table 5.5.

Stability analysis refers to analyzing how similar the reducts found in fold experiments with reduct found with original dataset. Stability of reduct is considered as a measure of reliability of reduct for constructing DM models [3]. The procedure to perform stability analysis as given in [112] is reported here. The reducts using IQRAIG_Incomplete, QRAIG_Incomplete are found for the entire dataset and also for 10 sub datasets resulting from folds of tenfold cross validation. In each fold the nine remaining portions leaving out one portion, usually considered as the test dataset portion, constitute the dataset represented as X_i for i^{th} fold. Let R_i denote the reduct obtained by applying IQRAIG_Incomplete on X_i . The following procedure is for obtaining stability measure for a tenfold experiment.

The distance measure $D(R_1, R_2)$ measures the difference between two reducts R_1, R_2 and is equal to zero in case both arguments are the same reducts. $D(R_i, R_j)$ is defined as,

Table 5.5: Tenfold Experiment results on benchmark datasets

Dataset	Reduct Length	CT (in seconds)	3NN Performance		Stability
			Original	Reduct	
Mushroom	5±0	7.217±0.03	87.155±2.052	100±0	0.167±0.176
Vote	13.3±2.312	5.2078±1.089	93.953±3.669	91.86±2.741	0.185±0.086
Lung Cancer	5.2±0.632	0.44875±0.009	46.667±23.307	36.667±18.922	0.442±0.366
Backup-large-test	10.8±0.789	7.0689±0.469	58.378±7.341	76.486±5.983	0.164±0.100
Shuttle	4±0	70.248±0.38	48.96±10.255	99.52±0.097	0.040±0.127
Adult	7.8±0.422	149.57±0.415	54.917±7.709	71.075±0.936	0.069±0.095
Hypothyroid	4±0	6.9182±0.048	95.429±1.274	95.524±1.254	0±0
Audiology Standardized	21.1±1.969	3.639±0.458	70.421±10.802	66.842±10.556	0.2443±0.1032
Soybean-large	12.1±0.738	6.8157±1.429	53.333±6.849	69.667±4.289	0.3611±0.1859
Dermatology	7±0	0.8377±0.0746	75±8.282	84.444±6.954	0.1889±0.1789
Breast Cancer Wisconsin	4.7±0.483	0.4202±0.0251	85.797±5.499	95.507±3.013	0.0400±0.0843

$$D(R_1, R_2) = 1 - \frac{|R_1 \cap R_2|}{|R_1 \cup R_2|} \quad (5.18)$$

The mean value between distances between fold reducts R_i for $i = 1 \dots 10$ and the reduct for the entire dataset represented as R_0 is given by,

$$\mu = \frac{\sum_{i=1}^{10} D(R_i, R_0)}{10} \quad (5.19)$$

and the standard deviation of these distances is given by,

$$\sigma = \sqrt{\frac{\sum_{i=1}^{10} (D(R_i, R_0) - \mu)^2}{10}} \quad (5.20)$$

The stability of a tenfold experiment is characterized by μ, σ values. Lower the mean and standard deviation values higher the stability of the algorithm. Table 5.5 contain the stability values reported using μ, σ values.

5.5.3 Experiments with Trivial Ambiguous Incomplete Decision Systems

Three incomplete decision systems Vote, Backup-large-test and Soybean-large described in table 5.3 are having trivial ambiguity. Hence at least once in the execution of IQRAIG_Incomplete the computations of secondary heuristic of information gain using Information entropy given in Eq. 5.17, for resolving trivial ambiguity, are involved with these decision systems. In this section the impact of resolving trivial ambiguity using secondary heuristic is investigated. As described in the earlier section, the comparison is performed between IQRAIG_Incomplete algorithm with IQRA_Incomplete (IQRAIG_Incomplete algorithm without handling trivial ambiguity) algorithm and results are reported in Table 5.6.

Order invariance analysis is done by performing hundred trails of experiment

Table 5.6: Trivial Ambiguity Experiment Results

Dataset	Reduct Sizes		Computational Time (in seconds)		$\gamma_C(\{d\})$
	IQRAIG_Incomplete	IQRA_Incomplete	IQRAIG_Incomplete	IQRA_Incomplete	
Vote	12	11	5.46	5.03	0.326
Backup-large-test	11	12	7.48	8.21	0.907
Soybean-large	12	13	7.39	6.96	0.928

Table 5.7: Order Variance Experiment Results on Trivial Ambiguous Datasets

Dataset	Reduct Lengths		Distinct Reduct Count	
	IQRAIG_Incomplete	IQRA_Incomplete	IQRAIG_Incomplete	IQRA_Incomplete
Vote	12,13,16	11,12,13,14,15,16	3	27
Backup-large-test	12,13,14,15	12,13,14,15,16,17	25	85
Soybean-large	11,12,13	11,12,13,14,15,16,17	31	96

with IQRAIG_Incomplete and IQRA_Incomplete algorithms. In each trail, the decision system's order of conditional attributes is randomly permuted and both algorithms are applied to derive the reducts. The logical mapping of the attributes in each trail, with original attributes is used to convert all reducts into one single representation of attribute labels. The distinct reducts obtained in hundred trails is reported for both algorithms in Table 5.7.

5.5.4 Comparative Experiments

The proposed algorithms are compared against IFSPA-IPR, IPR algorithms given in [112]. The reason for this comparison is that IFSPA-IPR algorithm is a single reduct finding algorithm which is a positive region based quick reduct like algorithm for IDS incorporating positive region removal (called as positive approximation). IPR algorithm is similar to IFSPA-IPR but without having the feature of positive region removal. Hence IPR algorithm is similar in nature to QRAIG_Incomplete algorithm and IFSPA-IPR is similar to IQRAIG_Incomplete algorithm. In order to perform comparative analysis with results of IPR, IFSPA-IPR, experiments are conducted with QRAIG_Incomplete, IQRAIG_Incomplete following the design of the experiments in [112]. The design of experiments is again repeated here for completeness.

The datasets used in [11] are from UCI Machine learning repository and details of them (Audiology Standardized, Soybean-large, Dermatology, Breast-cancer-wisconsin) are given in the Table 5.3 along with the datasets that are used in the sub section 5.5.1. There are two experiments conducted on these datasets.

The first experiment is for the purpose of understanding the impact of positive region removal and hence the comparison is done between QRAIG_Incomplete and IQRAIG_Incomplete from computation time perspective. In the first experiment, each dataset the rows are randomly permuted. The random ordered dataset is divided into 20 equal portions. In the first trail only the first portion is

Figure 5.1: CT Comparison of QRAIG_Incomplete vs IQRAIG_Incomplete for various sizes of dataset

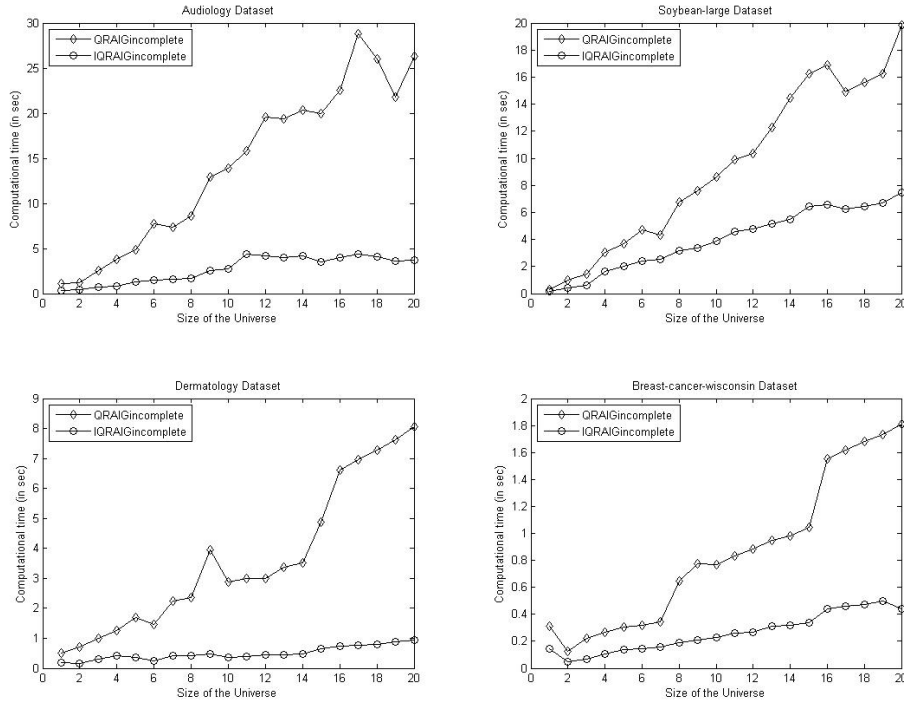


Table 5.8: CT Comparison of existing and proposed algorithms in tenfold experiment

Dataset	IPR	QRAIG_Incomplete	IFSPA-IPR	IQRAIG_Incomplete
Audiology Standardized	72.2859±5.2078	24.363±2.526	17.205±1.1261	3.639±0.458
Soybean-large	34.2984±3.8632	18.207±1.8313	10.8922±1.4842	6.8157±1.429
Dermatology	43.0281±1.9503	7.2694±0.137	15.6281±0.5176	0.8377±0.0746
Breast-cancer Wisconsin	12.5109±1.7053	1.667±0.244	4.4656±0.8354	0.4202±0.0251

taken and reducts are found using IQRAIG_Incomplete and QRAIG_Incomplete (obviously the reducts found will be same excepting having difference in computational time). In the second trail first two portions are considered and this is continued for 20 trails. Computational time (in seconds) in each trail for IQRAIG_Incomplete and QRAIG_Incomplete algorithms are given in Fig. 5.1 for the four datasets considered.

The second experiment is to perform a tenfold cross validation experiment to compare the proposed algorithms with IFSPA-IPR, IPR algorithms using parameters computational time and stability.

Table 5.8 gives the mean and standard deviation values of computational time (in seconds) of obtaining reduct using QRAIG_Incomplete, IQRAIG_Incomplete algorithms along with the results reported in [112] for IPR and IFSPA-IPR algorithms. Table 5.9 contains the results for stability analysis.

Figure 5.2: CT (range $\mu - \sigma$ to $\mu + \sigma$) Comparison of IPR, QRAIG_Incomplete, IFSPA-IPR, IQRAIG_Incomplete

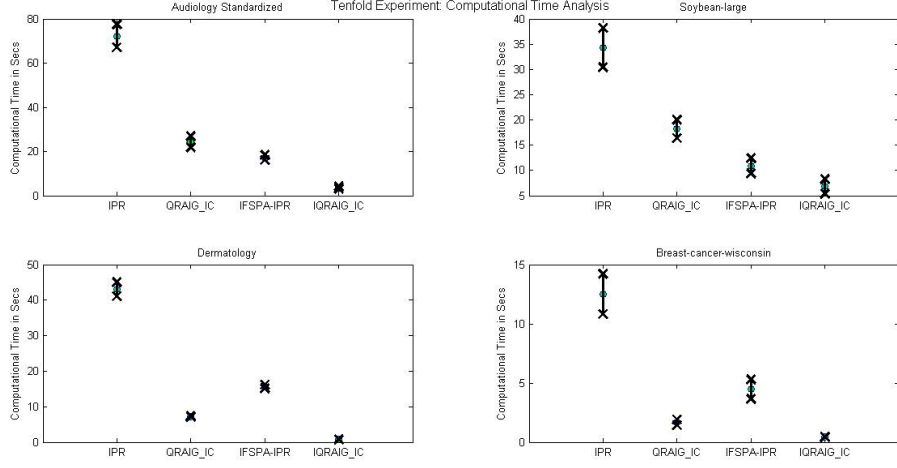
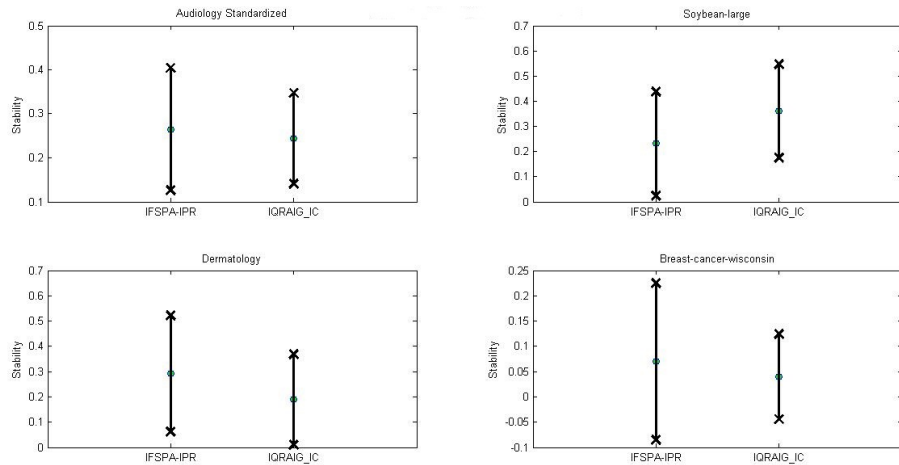


Table 5.9: Stability Comparison of IFSPA-IPR with IQRAIG_Incomplete in tenfold experiment

Dataset	IFSPA-IPR	IQRAIG_Incomplete
Audiology Standardized	0.2650±0.1393	0.2443±0.1032
Soybean-large	0.2312±0.2071	0.3611±0.1859
Dermatology	0.2921±0.2293	0.1889±0.1789
Breast-cancer Wisconsin	0.0700±0.1552	0.0400±0.0843

Figure 5.3: Stability (range $\mu - \sigma$ to $\mu + \sigma$) comparison in tenfold experiment



5.6 Analysis of Results

The datasets used in experiments contain significant incompleteness with some of the datasets like Shuttle, Adult containing thousands of missing values (See Table 5.3). It is also observed that due to the presence of missing values many of these datasets are inconsistent in nature (See Table 5.4). IQRAIG_Incomplete algorithm is computationally efficient in obtaining super reducts for these datasets in quick time (Table 5.4).

Tenfold experiment results in Table 5.5 establish that IQRAIG_Incomplete algorithm computes super reducts with good stability. Mean stability values of IQRAIG_Incomplete are near to zero for most of the datasets. 3NN classification performance using reduct attributes is at least comparable with that of all attributes in most datasets and significantly a huge jump in mean percentage of classification is observed in datasets such as Mushroom (87 to 100), Backup-large-test (58 to 76), Shuttle (48 to 99) etc. But in Lung Cancer, Audiology Standardized datasets, reduct attributes performance has decreased compared to all attributes. Overall reduct obtained using IQRAIG_Incomplete algorithm have proved to increase the generalization ability of the classifier significantly. These experiments establish the utility of reduct (dimensionality reduction) in enhancing the generalization ability of the classifier and in particular the importance of rough sets in dealing with uncertainty and vagueness arising out of incompleteness of the dataset.

Trivial ambiguity experiment results in Tables 5.6 & 5.7 are similar to results obtained for CDS in Chapter 3. Trivial ambiguity resolution resulted in lowering the distinct reduct count and reduced the range of the lengths of the reducts in 100 trails.

Fig. 5.1 evidently depicts the impact of positive region removal in reduction in computational time in IQRAIG_Incomplete over QRAIG_Incomplete algorithm. Unpaired t-test is administered on the experimental data arrived by tenfold experimentation on computational time and stability results given in Tables 5.8 & 5.9. Summary of results (range $\mu - \sigma$ to $\mu + \sigma$) for computational time and stability are depicted in Fig. 5.2, Fig. 5.3 respectively. The obtained results revealed IQRAIG_Incomplete has achieved statistically significant betterment over IFSPA-IPR for all datasets in computational time aspect, whereas no statistical significance is observed in stability aspect. The computational gain percentages of IQRAIG_Incomplete over IFSPA-IPR varies from 37% to 94%.

An important observation is that QRAIG_Incomplete (without positive region removal), using the proposed granular framework, achieved statistically significant betterment in computational time over IFSPA-IPR (with positive region removal)

for Dermatology and Breast cancer Wisconsin datasets. These computational gains are because of simultaneous computation of positive region for sub granule objects using proposed granular framework and establishes the relevance of proposed granular framework.

5.7 Summary

Present study introduced a novel granular framework for reduct computation in IDS. The proposed granular framework enabled the simultaneous computation of positive region for sub granule objects. New definitions of lower approximation, positive region, conditional information entropy are introduced using this granular framework.

Algorithm IQRAIG_Incomplete is proposed for reduct computation for IDS having the features of IQRA_IG algorithm such as positive region removal, trivial ambiguity resolution using IG heuristic. Theoretical validation of positive region removal in IQRAIG_Incomplete algorithm is provided. Experimental setups are designed to bring out the effectiveness of proposed method on benchmark IDS and also to compare with IFSPA-IPR algorithm. Experiments revealed the merits of IQRAIG_Incomplete algorithm in obtaining significant computational gains without compromising stability. Chapter 7 is devoted for seed based scalable approaches for IDS as well as hybrid decision systems.

Chapter 6

Reduct Computation in Hybrid Decision Systems

In previous chapters, ERCS are developed for symbolic decision systems. The real world decision systems, in general, possess qualitative (symbolic) as well as quantitative (numeric) attributes which are known as hybrid decision systems. Various extensions of rough sets are available in the literature for hybrid decision systems. One of the approaches, namely fuzzy rough sets, has been considered for developing ERCS in this chapter.

Section 6.1 presents detailed literature review of extensions of rough sets for hybrid decision systems and in particular about fuzzy rough sets and reduct computation in fuzzy rough sets. Section 6.2 gives an overview of Radzikowska Kerry's fuzzy rough set model, fuzzy decision reduct, and modified quick reduct algorithm for computing fuzzy decision reduct. A simplified computational model for Radzikowska-Kerry's fuzzy rough set model is proposed in Section 6.3. The philosophy of absolute positive region removal and their theoretical validation leading to improved modified quick reduct algorithm (IMQRA) and vector based modeling of operations of IMQRA resulting in algorithm IMQRA_MW are also described in Section 6.3. An illustration on a toy dataset is given in Section 6.4. In section 6.5, non persistent stratified sampling based IMQRA is developed as NPSS_IMQRA algorithm for scalable fuzzy decision reduct computation. Experiments are described in Section 6.6 and analysis of results is given in Section 6.7.

6.1 Hybrid Decision Systems and Extensions of Rough Sets

Hybrid decision systems (HDS) contain a mixture of qualitative and quantitative attributes and occur frequently in real world decision systems. The classical definitions of rough sets are based on an indiscernibility relation, which is an equivalence relation. Hence under indiscernibility relation using a quantitative attribute, two objects will be unrelated even though they have near values on a real-values scale.

A reduct computed thus would contain primary key like attributes and leads to classifiers with less generalization capacity.

Hence classical rough sets cannot be applied directly to hybrid decision systems for reduct computation. Traditionally indiscernibility relation using quantitative attribute was defined after discretization. The process of discretization converts a quantitative attribute into a qualitative attribute. A discretization algorithm places cuts in the domain of quantitative attribute and divides the continuous domain into non overlapping intervals (bins). Two objects having values in the same interval are assigned the same symbolic label. Optimal way of finding cuts was proved to be NP-Hard [88]. Hence heuristic based sub optimal discretization algorithms were used [134, 142] and a widely used approach which is based on rough set principles and boolean reasoning was given in [90].

Discretization approach provided a computationally effective way for dealing with hybrid decision systems using classical rough sets. But information loss is inevitable in any discretization approach. Slowinski et al., in [130], reported that the primary issue in data preprocessing for rough set analysis is with the uncertainty arising out of discretization of quantitative attributes. Based on repeated experiments using several discretization approaches, it was observed that no single discretization approach is appropriate for all decision systems [10]. In [51] it is empirically established that decision systems having attributes conforming to statistically highly skewed distributions or having high peaks are resulting in higher classification errors irrespective of discretization method used. It is also observed in [117] that discretization proves to be ineffective and leads to more information loss and results in significant classification error in inconsistent decision systems (having overlapping feature space of objects belonging to different decision classes). Approaches like soft cuts [89] and fuzzy discretization [117] were introduced to represent the ambiguity present in the neighborhood region of crisp cuts. These approaches were proved to be effective in reducing the classification error for inconsistent decision systems [92, 96].

The above mentioned improvements for discretization were intended at building efficient classifiers and not oriented at developing reduct computation approaches for hybrid decision systems. Two fundamental issues of rough sets, as stated in [173] are, representations of indiscernibility/discernibility and attribute reduction based on indiscernibility/discernibility. Hence attribute reduction approaches for hybrid decision systems were formulated by defining new representations of indiscernibility relation.

Alternative approaches for reduct computation in hybrid decision systems (without relying on discretization) were evolved by generalizing the classical rough set theory. Generalization was primarily meant to relax the equivalence properties of indiscernibility relation and using similarity relation which are more general bi-

nary relations [41, 164]. These approaches led to significant advancements in rough set theory both from theoretical and application view points. Similarity relations [72, 91, 129] allow a limited degree of variability in the quantitative attribute values and the degree is determined by a threshold value. The similarity class of an object is defined as the collections of related objects under similarity relation [128]. The rough approximations are defined based on similarity classes of objects. In particular, tolerance rough sets [80, 105, 123, 126] are based on reflexive and symmetric tolerance relations and reduct of hybrid decision system under tolerance rough sets is named as tolerance reduct. Determining appropriate threshold value is essential in tolerance rough sets for obtaining effective feature selection and classifier performance. Some of the approaches for threshold determination are using genetic algorithm [65] and EM algorithm [31].

Under tolerance relation, an object is related to all objects at varying distance as long as the threshold requirement is satisfied. A more semantically apt representation was obtained by bringing in fuzzy context. The fuzzy similarity relation [35] defines the degree of relatedness between any two objects based on the quantitative attribute values. Fuzzy similarity relation results in representation of approximate equality or graded indisernibility [20] of objects based on quantitative attribute values. Among the generalizations of rough sets for reduct computation [25, 26, 47, 71, 78, 86, 104], fuzzy rough sets [25, 26] evolved to be an important generalization of rough sets. Fuzzy rough sets are extensively used for reduct computation in hybrid decision systems.

Fuzzy rough sets and rough fuzzy sets were introduced by Dubois and Prade [25, 26] which have become standard approaches. The approximation of a crisp set in a fuzzy approximation space is called as a fuzzy rough set, and the approximation of a fuzzy set in a crisp approximation space is called as rough fuzzy set [57]. The modifications of this standard approach has been addressed in [45, 114, 152]. Radzikowska and Kerre [114] has given a generalized approach for Dubois-Prade fuzzy rough set model in the form of fuzzy rough set defined by an implication and a triangular norm (t-norm). Hu et al. [45] developed a new fuzzy rough set model using fuzzy generalized definitions of subset and intersection of sets in classical notions of lower and upper approximations. Wang et al. [152] developed a fuzzy rough set model based on thresholds on similarity degree for inclusion into lower and upper approximations. W. Wei et al. [154] has given a comparative study of these fuzzy rough set models. W. Wei et al. [154] established that Hu's, Wang's models are derivable from Dubois-Prade's model whereas Dubois-Prade's model is derivable from Radzikowska-Kerry's model. Thus Radzikowska-Kerry's model is a generalized model in the class of fuzzy rough set models.

Research has been advanced in fuzzy rough set models and also in reduct computation (feature selection) using these models in hybrid decision systems. Fuzzy

rough feature selection (FRFS) was introduced by Jensen et al. [56] based on Dubois-Prade's fuzzy rough set model. In FRFS approach every quantitative attribute is represented by a set of linguistic variables (fuzzy sets) resulting in fuzzy partition of the quantitative attribute [130]. The reduction algorithm was of positive region based heuristic approach similar to quick reduct algorithm [18] called as fuzzy rough quick reduct algorithm. The computational efficiency of fuzzy rough quick reduct algorithm is improved using the concept of compact computational domain by Bhatt et al. [7]. Later Jensen et al. [58] proposed reduct algorithms like B-FRFS, L-FRFS, FDM which are computationally more efficient than FRFS approach. Hu et al. [46] provided a reduct algorithm based on fuzzy conditional entropy. Hu et al. [45] developed FAR-VPFRS algorithm (Forward attribute reduction based on variable precision fuzzy rough model) using Hu's Fuzzy Rough Set model [45] which is a positive region heuristic based algorithm. Qian et al. [109] has given an improved feature selection algorithm based on forward approximation (FA-FSCE). FA-FSCE algorithm incorporates positive region removal (positive approximation) in an iteration of reduct finding process using Hu's fuzzy rough set model and fuzzy conditional entropy. Cornelis et al., [20] introduced the concept of fuzzy decision reduct and provided a modified quick reduct algorithm (MQRA) for finding fuzzy decision super reduct (FDSR) based on Radzikowska-Kerry's fuzzy rough set model. MQRA is an extended version of quick reduct algorithm [18] for hybrid decision systems and is a positive region heuristic based algorithm.

Finer and apt knowledge representation in hybrid decision system is achieved monotonically from classical rough sets to tolerance rough sets to fuzzy rough sets. The effective enhancement of representation incurs more computational and space complexities. Reduct computation using fuzzy rough sets incurs higher time and space complexities compared to classical rough sets. The present study aims at effective fuzzy decision reduct computation by reducing practical space and time complexities of MQRA algorithm.

The aspect of positive region removal embedded in approaches developed for complete symbolic decision systems in Chapters 3 & 4, for incomplete symbolic decision systems in Chapter 5 is essential and contributes significantly for reduction of space and time complexities. The approach is evident in classical rough sets as positive region equivalence classes are disjoint from non positive region equivalence classes in the partition induced by indiscernibility relation. But in fuzzy rough set approach computing positive region membership of an object requires computations with respect to every object of the universe. This necessitates theoretical validation of removal of objects which belong absolutely in positive region in fuzzy rough set model based reduct computation approaches. There exists generalized rough set models like variable precision rough set model [175] in which monotonic criteria of positive region membership of an object is not satisfied [172]. Hence it

is required to establish that in the generalized model of fuzzy rough sets monotonic criteria of positive region membership holds which is the precondition for positive region removal aspect. The positive region removal (positive approximation) was used in [109] for Hu's fuzzy rough set model. Present study establishes theoretically positive region removal aspect in Radzikowska-Kerry's fuzzy rough set model.

This study addresses mainly four aspects. The first is developing a simplified computational model for Radzikowska-Kerry's fuzzy rough set model and addressing the positive region removal issues. Second is to incorporate the simplified computational model and positive region removal in MQRA resulting in Improved MQRA (IMQRA). The third part involves modeling the computations in IMQRA with vector based operations so that further computational efficiency is obtained. The Matlab implemented versions of MQRA, IMQRA are referred as MQRA_MW, IMQRA_MW respectively. Based on compact support approaches for fuzzy decision reduct computation incorporated into IMQRA_MW algorithm, the final part of the chapter deals with developing sampling based scalable approaches for fuzzy decision reduct computation. In view of analysis done in Chapter 4, a non persistent stratified sampling based approach for IMQRA_MW is developed as NPSS_IMQRA algorithm. The computational efficiency of MQRA_MW, IMQRA_MW, NPSS_IMQRA is established empirically by performing comparative analysis with existing FSCE and FA-FSCE algorithms [109] and algorithms FRFS, B-FRFS, L-FRFS, FDM [58].

6.2 Overview of Fuzzy Decision Reduct

Let $HDT = (U, C^h = C^c \cup C^n, \{d\}, \{V_{a_c}, f_{a_c}\}_{a_c \in C^c \cup \{d\}}, \{V_{a_n}, f_{a_n}\}_{a_n \in C^n})$ be a hybrid decision system (table). Here U is the set of objects, C^c is set of qualitative (categorical) attributes and C^n is set of quantitative (numerical) attributes and C^h is the set of hybrid conditional attributes comprising of C^c and C^n . For every qualitative attribute $a_c \in C^c \cup \{d\}$, V_{a_c} denotes set of finite domain of values of attribute a_c and $f_{a_c} : U \rightarrow V_{a_c}$ is a mapping of assigning a symbol in V_{a_c} to every object in U . For every quantitative attribute $a_n \in C^n$, V_{a_n} has a range, a finite interval on real line with normalized range between 0 and 1 and $f_{a_n} : U \rightarrow V_{a_n}$ is a mapping assigning a value in V_{a_n} to every object in U . Notation $a_n(x)$ in place of $f_{a_n}(x)$ for quantitative attributes, $a_c(x)$ in place of $f_{a_c}(x)$ for qualitative attributes being used for simplicity. This study deals with decision systems and hence decision attribute d is taken as qualitative attribute. There are approaches for computation of reduct for hybrid regression systems where in d is a quantitative attribute [20] and will be explored in future work.

This section provides associated definitions, terminology, concepts for fuzzy rough sets and fuzzy decision reduct, for completeness, based on [19, 20, 114]. In classical rough sets, the approximation space is defined as (U, R) , where U is the set of objects and R is the indiscernibility relation defined on U satisfying the properties of an equivalence relation. A concept $A \subseteq U$ is approximated with the knowledge of R as lower and upper approximations. The equations for lower and upper approximation in classical rough sets given in Eq. (2.1), Eq. (2.2) can be specified from the view of object belonging into lower and upper approximations. An object $y \in U$ belongs to lower approximation of A ($R \downarrow A$) iff the equivalence class of y with respect to R is contained in A and belongs to upper approximation of A ($R \uparrow A$) iff the equivalence class of y with respect to R has nontrivial intersection with A . Hence,

$$y \in R \downarrow A \iff [y]_R \subseteq A. \quad (6.1)$$

$$y \in R \uparrow A \iff [y]_R \cap A \neq \phi. \quad (6.2)$$

The equivalent forms of Eq. (6.1), Eq. (6.2) are [19] given in Eq. (6.3), Eq. (6.4).

$$y \in R \downarrow A \iff (\forall x \in U) ((x, y) \in R \Rightarrow x \in A). \quad (6.3)$$

$$y \in R \uparrow A \iff (\exists x \in U) ((x, y) \in R \wedge x \in A). \quad (6.4)$$

6.2.1 Fuzzy Rough Set Theory

Similarity relation has been introduced [25, 26, 47, 78, 128, 129] for addressing the issues associated with discretization instead of indiscernibility relation through fuzzification. The theory developed in this direction is known as fuzzy rough set theory. In fuzzy rough set theory the similarity relation is a fuzzy tolerance relation or a fuzzy equivalence relation. A fuzzy relation R on $U \times U$ is said to be a fuzzy tolerance relation if,

$$R(x, x) = 1 \quad \forall x \in U. \quad (6.5)$$

$$R(x, y) = R(y, x) \quad \forall x, y \in U. \quad (6.6)$$

R becomes a fuzzy equivalence relation with an additional requirement of Γ -transitivity using a given t-norm Γ . The Γ -transitivity property is

$$\Gamma(R(x, y), R(y, z)) \leq R(x, z) \quad \forall x, y, z \in U. \quad (6.7)$$

In modeling fuzzy rough set based system for HDT , the fuzzy tolerance or equivalence relation is defined as R_{a_n} ($\forall a_n \in C^n$). Hence $R_{a_n}(x, y)$ for objects x, y denote the degree of similarity between x and y using the attribute values of a_n . If a_c is a qualitative attribute then the resulting similarity relation is taken as the classical indiscernibility relation and hence is defined as,

$$R_{a_c}(x, y) = \begin{cases} 1 & \text{if } a_c(x) = a_c(y), \\ 0 & \text{if } a_c(x) \neq a_c(y). \end{cases} \quad (6.8)$$

The similarity relation is extended for a set of attributes using a specified t-norm Γ , i.e. given $B \subseteq C^h \cup \{d\}$, fuzzy tolerance or equivalence relation is extended as,

$$R_B(x, y) = \Gamma \left(\underbrace{R_a(x, y)}_{a \in B} \right) \quad \forall x, y \in U. \quad (6.9)$$

In classical rough sets, concept to be approximated is a subset of U corresponding to an equivalence class associated with a decision value. In fuzzy rough set theory the concept to be approximated is generalized into a fuzzy set A . Each object x has a degree of membership into concept A with value $A(x)$. In HDT , $R_{d,y}$ of each object y defines a fuzzy concept where,

$$R_{d,y}(x) = R_d(x, y) \quad \forall x \in U. \quad (6.10)$$

Definition of $R_{d,y}$ (R_d -foreset) is general and not restricted only to decision variable. It is to be noted that as d is a qualitative attribute, $R_{d,y}$ represents the characteristic function of set of y 's decision class objects.

6.2.1.1 Radzikowska-Kerry's Fuzzy Rough Set Model

In a fuzzy rough set model, the concepts of lower and upper approximation of a fuzzy set A defined on U are defined using a fuzzy Γ -equivalence relation (or fuzzy tolerance relation) R in U . In Dubois-Prade's fuzzy rough set model [25, 26], given a fuzzy Γ -equivalence relation (or a fuzzy tolerance relation) in U , the lower and upper approximations of a fuzzy concept A are defined $\forall y \in U$ as,

$$\mu_{\underline{R}(A)}(y) = \inf_{x \in U} \max(1 - R(x, y), A(x)). \quad (6.11)$$

$$\mu_{\overline{R}(A)}(y) = \sup_{x \in U} \min(R(x, y), A(x)). \quad (6.12)$$

Eq. (6.11) and Eq. (6.12) are fuzzy generalized versions of lower and upper approximations in classical rough set model given in Eq. (6.3) and Eq. (6.4) using Kleene-Dienes Implication [1] and minimum t-norm [29]. Radzikowska-Kerry's model [114] generalizes Dubois-Prade's model by a pair of implication $\Im$ and t-norm Γ as,

$$R \downarrow A(y) = \inf_{x \in U} \Im(R(x, y), A(x)). \quad (6.13)$$

$$R \uparrow A(y) = \sup_{x \in U} \Gamma(R(x, y), A(x)). \quad (6.14)$$

The above definitions treat lower and upper approximations as fuzzy sets and for each object its degree of membership into lower and upper approximation is defined. Similarly positive region is defined. Given $B \subseteq C^h$ the fuzzy B -positive region of an object $y \in U$ is defined as,

$$POS_B(y) = \left(\bigcup_{x \in U} R_B \downarrow R_{d,x} \right)(y). \quad (6.15)$$

Hence the positive region of object y is the maximum membership degree of lower approximation of R_B into R_d -foreset of each object in U . As the decision attribute is qualitative, using proposition-1 given in [20], the fuzzy B -positive region computation requires computing lower approximation of $R_{d,y}$ only. Hence Eq. (6.15) simplifies to,

$$POS_B(y) = R_B \downarrow R_{d,y}(y). \quad (6.16)$$

6.2.1.2 Fuzzy Decision Reduct

Reduct for HDT using the definitions of lower approximation and positive region given in section 2.1.1 is called as fuzzy decision reduct. Fuzzy decision reduct [20] is defined as given below.

Let M be monotonic function from $\wp(U) \rightarrow [0, 1]$ ($\wp(U)$ denotes power set of U) such that $M(C^h) = 1$. A set of attributes B subset of C^h is

Algorithm 11: Modified Quick Reduct

Input : $HDT = (U, C^h = C^c \cup C^n, \{d\}, \{V_{a_c}, f_{a_c}\}_{a_c \in C^c \cup \{d\}}, \{V_{a_n}, f_{a_n}\}_{a_n \in C^n})$,
Measure M ,
Degree α
Output: Fuzzy M -decision super reduct B

```
1  $B = \phi$ 
2 repeat
3    $T = B$ 
4    $best = -1$ 
5   foreach  $a \in C^h - B$  do
6     if  $M(B \cup \{a\}) \geq best$  then
7        $T = B \cup \{a\}$ 
8        $best = M(B \cup \{a\})$ 
9     end
10  end
11   $B = T$ 
12 until  $M(B) \geq \alpha$ 
13 Return  $B$ .
```

said to be a fuzzy M -decision super reduct to degree α if $M(B) \geq \alpha$. It is called a fuzzy M -decision reduct to degree α if $\forall B^* \subset B, M(B^*) < \alpha$.

The measure M which satisfies the required properties and in extension to the classical rough set based gamma measure is,

$$\gamma_B = \frac{|POS_B|}{|POS_{C^h}|}. \quad (6.17)$$

where $|POS_A| = \sum_{x \in U} POS_A(x)$ for $A \subseteq C^h$.

6.2.2 Modified Quick Reduct Algorithm

Modified quick reduct algorithm (MQRA) [20] is formulated for finding fuzzy M -decision super reduct (FDSR) following the structure of QRA [18]. For completeness, the algorithm is reproduced from [20] in Algorithm 11 using the notations of this chapter.

The time complexity of computing $M(B)$ in HDT is $O(|U|^2)$ based on Eq. (6.17). Hence the time complexity of MQRA is $O(|C^h|^2 |U|^2)$.

Quick Reduct algorithm has been modified for giving better computational performance in IQuickReduct algorithm (IQRA) given in Chapter 3. Inspired to improve MQRA in that direction the following section provides schematic development of an algorithm Improved Modified Quick Reduct (IMQRA).

6.3 Improvements to Modified Quick Reduct Algorithm

Two facets of IQRA related algorithms of Chapter-3 are resolution of ambiguous scenarios and positive region removal. Occurrence of ambiguous situations in MQRA is a rarity due to $M(B)$ being a continuous value which increases till the required $M(C^h)$ is obtained. Hence improvements to MQRA are aimed at obtaining positive region removal aspect in MQRA. Validation of positive region removal aspect is essential in fuzzy rough set models for proving the monotonic criteria of positive region membership and also that omission of objects does not affect the subsequent inferences. The later is essential as computation of positive region membership of an object involves fuzzy similarity values of the object with all members of U .

In Section 6.3.1 necessary theory is developed for simplified computation model for lower approximation and positive region in Radzikowska-Kerry's fuzzy rough set model for decision systems and validation of positive region removal in MQRA algorithm using Radzikowska-Kerry's fuzzy rough set model. Section 6.3.2 contains the proposed algorithm IMQRA and Section 6.3.3 describes the modeling of IMQRA computations using vector based operations suitable for environments such as Matlab [81]. Section 6.3.4 contains the time and space complexity analysis of IMQRA.

6.3.1 Theoretical foundations

The Radzikowska-Kerry's fuzzy rough set model described in Section 6.2 is characterized by a fuzzy tolerance or an Γ -equivalence relation R , an implication $\Im$, a t-norm Γ and the definition of M . The discussion in this study considers R as a fuzzy tolerance relation and M is the gamma measure defined in Eq. (6.17).

For any object $y \in U$ let $U_1(y)$ denote the set of objects belonging to the decision class of y and let $U_2(y)$ denote the remaining objects belonging to other decision classes. The natural negation generated by an implication $\Im$ according to [1] is as given below.

Let $\Im$ be any fuzzy implication. The natural negation of $\Im$, denoted by $N_{\Im}$, is given by $N_{\Im}(x) = \Im(x, 0) \forall x \in [0, 1]$.

Given an implication $\Im$, Lemma 6.3.1 derives the simplified formula for positive region POS_B .

Lemma 6.3.1. *For any $y \in U$ and $B \subseteq C^h$, using fuzzy tolerance relation R_B and implication $\Im$, if $U_2(y) \neq \phi$ then*

$$POS_B(y) = \min_{x \in U_2(y)} N_{\mathfrak{S}}(R_B(x, y))$$

else $POS_B(y) = 1$ where $N_{\mathfrak{S}}$ is the natural negation generated by $\mathfrak{S}$.

Proof. Using Eq. (6.16),

$$\begin{aligned} POS_B(y) &= R_B \downarrow R_{d,y}(y) \\ &= \inf_{x \in U} \mathfrak{S}(R_B(x, y), R_d(x, y)) \\ &= \min_{x \in U} \mathfrak{S}(R_B(x, y), R_d(x, y)) \text{ (since } U \text{ being finite, inf is equal to min)} \\ &= \min \left(\min_{x \in U_1(y)} \mathfrak{S}(R_B(x, y), R_d(x, y)), \min_{x \in U_2(y)} \mathfrak{S}(R_B(x, y), R_d(x, y)) \right) \\ &= \min \left(\min_{x \in U_1(y)} \mathfrak{S}(R_B(x, y), 1), \min_{x \in U_2(y)} \mathfrak{S}(R_B(x, y), 0) \right) \text{ (since } d \text{ being qualitative using Eq. (6.8))} \\ &= \min \left(\min_{x \in U_1(y)} 1, \min_{x \in U_2(y)} N_{\mathfrak{S}}(R_B(x, y)) \right) \text{ (Using neutrality of truth property [143]} \\ &\quad \forall x \in [0, 1] \mathfrak{S}(x, 1) = 1 \text{ and Definition of Natural Negation)} \\ &= \min \left(1, \min_{x \in U_2(y)} N_{\mathfrak{S}}(R_B(x, y)) \right). \end{aligned}$$

Hence if $U_2(y) \neq \phi$

then

$$POS_B(y) = \min_{x \in U_2(y)} N_{\mathfrak{S}}(R_B(x, y))$$

else

$$POS_B(y) = 1.$$

□

From Lemma 6.3.1 it can be observed that only $N_{\mathfrak{S}}$ instead of $\mathfrak{S}$ is needed in computing lower approximation and positive region while using Radzikowska-Kerry's $(\mathfrak{S}, \Gamma)$ -fuzzy rough set model for decision systems. As [1, 143] $N_{\mathfrak{S}}$ is a fuzzy negation, an useful conclusion is that for decision systems Radzikowska-Kerry's fuzzy rough set model is completely defined by a pair of fuzzy negation N and t-norm Γ and can be specified as (N, Γ) -fuzzy rough set model. This observation facilitates experimenter in varying only the fuzzy negation for different experiments of reduct computation using Radzikowska-Kerry's model as experimenting with different implications generating the same natural negation has no impact in the outcome.

Confining to similarity between objects defined over quantitative attributes, in practice, strong (and hence continuous) involutive [143] fuzzy negations are preferred and used. Considering standard negation $N(x) = 1 - x$ which is often used, the simplified equation for positive region based on Lemma 6.3.1 is given in Eq. (6.18).

$$POS_B(y) = R_B \downarrow R_{d,y}(y) = \begin{cases} \min_{x \in U_2(y)} (1 - R_B(x, y)) & \text{if } U_2(y) \neq \phi \\ 1 & \text{Otherwise} \end{cases} \quad (6.18)$$

It is observed that Eq. (6.18) coincides with lower approximation obtained in Dubois-Prade's model [27] for approximating a crisp set. In [154], while comparing fuzzy rough set models, it is established that Radzikowska-Kerry's model is equal to Dubois-Prade's model by using a pair of Kleene-Dienes's implication $\Im_{KD}$ and minimum t-norm Γ_M . In view of the above discussion, for decision systems using (N, Γ) -fuzzy rough set model it is further stated that, Radzikowska-Kerry's model becomes equal to Dubois-Prade's model by using a pair of standard negation and minimum t-norm Γ_M .

From now onwards Eq. (6.18) is used for computation of $POS_B(y)$. To derive a formula similar to positive region removal, monotonic criteria for $POS_B(y)$ is required. Lemma 6.3.2 proves the monotonic property of $POS_B(y)$.

Lemma 6.3.2. *If $B_1 \subseteq B_2 \subseteq C^h$, then for any $y \in U$, $POS_{B_1}(y) \leq POS_{B_2}(y)$.*

Proof. If $U_2(y) = \phi$ then trivially $POS_{B_1}(y) = POS_{B_2}(y) = 1$ using Lemma 6.3.1.

Because of usage of t-norm in construction of R_B in Eq. 6.9 ,

$$\begin{aligned} R_{B_2}(x, y) &\leq R_{B_1}(x, y) \text{ for all } x \in U \\ \Rightarrow 1 - R_{B_1}(x, y) &\leq 1 - R_{B_2}(x, y) \quad \forall x \in U \\ \Rightarrow 1 - R_{B_1}(x, y) &\leq 1 - R_{B_2}(x, y) \quad \forall x \in U_2(y) \\ \Rightarrow \min_{y \in U_2(x)} (1 - R_{B_1}(x, y)) &\leq \min_{y \in U_2(x)} (1 - R_{B_2}(x, y)) \\ \Rightarrow POS_{B_1}(y) &\leq POS_{B_2}(y) \text{ using Eq. 6.18.} \end{aligned}$$

□

If the degree of dissimilarity of y with all objects in $U_2(y)$ is '1' then $POS_B(y) = 1$. In which case object y belongs absolutely into positive region. $ABSOLUTE_POS_B$ denotes collection of all such absolute members of positive region, i.e.,

$$ABSOLUTE_POS_B = \{y \in U / POS_B(y)=1\} \quad (6.19)$$

The following properties regarding lower approximations in the four fuzzy rough set models discussed earlier are established in [154].

1. λ -cut of Dubois-Prade's fuzzy lower approximation with $\lambda = 1$ (absolute positive region membership) implies Hu's fuzzy lower approximation.
2. λ -cut of Dubois-Prade's fuzzy lower approximation implies Wang's fuzzy lower approximation with $\beta = \lambda$.
3. Wang's fuzzy lower approximation with $\beta = 1$ is equivalent to Hu's fuzzy lower approximation.

The above properties along with the property established in this chapter for decision systems, i.e.,

4. In Radzikowska-Kerry's (N, Γ) -fuzzy rough set model, fuzzy lower approximation using standard negation and minimum t-norm Γ_M is equivalent to Dubois-Prade's fuzzy lower approximation

the following synthesis can be arrived. Given the same external parameters such as fuzzy similarity relation R , t-norm used in extending the fuzzy similarity relation R for a set of attributes as given in equation 6.9, using standard negator and minimum t-norm in Radzikowska-Kerry's (N, Γ) -fuzzy rough set model and with $\beta = 1$ in Wang's fuzzy rough set model, given any $B \subseteq C^h$, all four fuzzy rough models compute the same $ABSOLUTE_POS_B$ and Wang's and Hu's models only return $ABSOLUTE_POS_B$ as their resultant lower approximation (as a subset of U) and both Radzikowska-Kerry's and Dubois-Prade's models assign the same positive region membership to $U - ABSOLUTE_POS_B$ objects.

In essence, Radzikowska-Kerry's and Dubois-Prade's models compute the absolute and non absolute positive region memberships and Hu's and Wang's deal only with absolute positive region membership. Hence reduct computation using Radzikowska-Kerry's and Dubois-Prade's models is based on complete knowledge available in the dataset as reduct computation based on Hu's and Wang's models does not consider the boundary region objects. Hence choosing Radzikowska-Kerry's model as the basis for reduct computation in current study is appropriate. As the gamma measure using equation 6.17 grows continuously because of non absolute positive region members it is very rare or never in practice that trivial ambiguous situations arise in MQRA. Hence only positive region removal aspect is considered in improvements to MQRA.

Here the possibility of removal of $ABSOLUTE_POS_B$ objects without affecting calculations of positive region for remaining objects is investigated. Let $HDT' = (U', C^h = C^n \cup C^c, \{d\}, \{V_a, f_a\}_{a \in C^h \cup \{d\}})$ denote the decision system after removal of $ABSOLUTE_POS_B$ objects in HDT, i.e.,

$$U' = U - ABSOLUTE_POS_B$$

Lemma 6.3.3 follows from the definition of HDT' .

Lemma 6.3.3. *For an object $y \in U'$,*

$$U'_1(y) = U_1(y) - ABSOLUTE_POS_B$$

$$U'_2(y) = U_2(y) - ABSOLUTE_POS_B.$$

Lemma 6.3.4 establishes an useful property of objects of HDT' .

Lemma 6.3.4. *For an object $y \in U'$, $U'_2(y) \neq \phi$.*

Proof. Assume in contrary, $U'_2(y) = \phi$.

$$U'_2(y) = \phi \Rightarrow U_2(y) \subseteq ABSOLUTE_POS_B \text{ (Using Lemma 6.3.3)}$$

$$\Rightarrow R_B(x, y) = 0 \quad \forall x \in U_2(y)$$

$$\Rightarrow POS_B(y) = 1$$

$$\Rightarrow y \in ABSOLUTE_POS_B$$

which is a contradiction as $y \in U' \iff y \in U - ABSOLUTE_POS_B$.

□

Let $B^* \supset B$. Let POS_{B^*} , POS'_{B^*} denote positive region with respect to HDT , HDT' respectively.

Theorem 6.3.5. $POS_{B^*}(y) = 1 \quad \forall y \in ABSOLUTE_POS_B$ and

$$POS'_{B^*}(y) = POS_{B^*}(y) \quad \forall y \in U - ABSOLUTE_POS_B.$$

Proof. Let $y \in ABSOLUTE_POS_B$. Using Lemma 6.3.2, since $B \subset B^*$,

$$POS_{B^*}(y) \geq POS_B(y) = 1. \text{ Hence, } POS_{B^*}(y) = 1.$$

Let $y \in U - ABSOLUTE_POS_B$, (i.e., $y \in U'$)

Using Lemma 6.3.4, $U'_2(y) \neq \phi \Rightarrow U_2(y) \neq \phi$.

Using Eq. (6.18),

$$POS_{B^*}(y) = \min_{x \in U_2(y)} (1 - R_{B^*}(x, y))$$

$$\begin{aligned}
&= \min \left(\min_{x \in U'_2(y)} (1 - R_{B^*}(x, y)), \min_{x \in U_2(y) \cap ABSOLUTE_POS_B} (1 - R_{B^*}(x, y)) \right) \\
&= \min \left(\min_{x \in U'_2(y)} (1 - R_{B^*}(x, y)), \min_{x \in U_2(y) \cap ABSOLUTE_POS_B} (1) \right) \\
&\quad \left(\begin{array}{l} \because x \in U_2(y) \cap ABSOLUTE_POS_B \\ \Rightarrow y \in U_2(x) \wedge x \in ABSOLUTE_POS_B \\ \Rightarrow R_B(x, y) = 0 \Rightarrow R_{B^*}(x, y) = 0 \end{array} \right)
\end{aligned}$$

Hence, $POS_{B^*}(y) = \min_{x \in U'_2(y)} (1 - R_{B^*}(x, y)) = POS'_{B^*}(y)$.

□

Theorem 6.3.5 establishes the redundancy of $ABSOLUTE_POS_B$ objects in the computation of POS_{B^*} for $U - ABSOLUTE_POS_B$ objects and also that for any $B^* \supset B$ it is the case that $ABSOLUTE_POS_B \subseteq ABSOLUTE_POS_{B^*}$. The presence or removal of objects belonging to $ABSOLUTE_POS$ does not alter subsequent ambiguity resolution process. Hence in MQRA, objects which belong to absolute positive region in an iteration continues to remain in absolute positive region and does not affect the positive region computations for remaining objects in the further iterations.

Assume that absolute positive region objects are removed after each iteration of MQRA. Let $\{B_i\}$ be the sequence of recommended reduct sets in forward selection based incremental reduct algorithm, in particular MQRA, such that $B_{i+1} - B_i$ is a singleton set representing the attribute selected in $i + 1^{th}$ iteration where $B_0 = \phi$. Similarly let $\{HDT_i\}$ denote the sequences of hybrid decision systems, and let $\{U_i\}$ denote the set of objects of HDT_i used in i^{th} iteration of MQRA satisfying, $HDT_1 = HDT$, $U_1 = U$ and for $i > 1$, $U_i = U - ABSOLUTE_POS_{B_{i-1}}$. Theorem 6.3.5 validates the computations involved in each iteration of MQRA working only on non absolute positive region objects U_i . Hence the following corollary,

Corollary 6.3.6. *The results of MQRA are unaffected by incorporating absolute positive region removal such that the computation of positive region of objects are performed in i^{th} iteration using Theorem 6.3.5 such that $B^* = B_i$, $B = B_{i-1}$ with $B_0 = \phi$ and $HDT' = HDT_i$, $HDT = HDT_{i-1}$.*

The removal of absolute positive region objects in each iteration reduces the computational complexity and space complexity of MQRA. The equation for gamma measure becomes,

$$\gamma_{B^*} = \frac{|ABSOLUTE_POS_B| + |POS'_{B^*}|}{|POS_{C^h}|} \quad (6.20)$$

Algorithm 12: Improved Modified Quick Reduct

Input : $HDT = (U, C^h = C^c \cup C^n, \{d\}, \{V_{a_c}, f_{a_c}\}_{a_c \in C^c \cup \{d\}}, \{V_{a_n}, f_{a_n}\}_{a_n \in C^n})$,
 R : Fuzzy similarity relation
 N : Fuzzy Negation
 Γ : t-norm
 α : Degree
Output: B : Fuzzy M - Super reduct with degree α

```
1 Compute  $R_{a,x}$  for all  $a \in C^h \cup \{d\}$  and  $x \in U$ .
2 Compute  $POS_{C^h}$ ,  $|POS_{C^h}|$ .
3  $B = \phi$ 
4  $\gamma_B = 0$ 
5  $ABSOLUTE\_POS_B = \phi$ ,
6  $\gamma_{ABSOLUTE} = 0$ 
7  $\gamma'_B = 0$ 
8 while  $\gamma_B < \alpha$  do
9    $T = B$ ,
10   $best = -1$ 
11  foreach  $a \in C^h - B$  do
12    if  $\gamma'_{B \cup \{a\}} > best$  then
13       $T = B \cup \{a\}$ 
14       $best = \gamma'_{B \cup \{a\}}$ 
15    end
16  end
17   $B = T$ 
18   $\gamma_B = \gamma_{ABSOLUTE} + \gamma'_B$ 
19  Update  $ABSOLUTE\_POS_B$ ,  $\gamma_{ABSOLUTE}$ .
20  Construct  $HDT'$  by removal of  $ABSOLUTE\_POS_B$  objects
21 end
22 Return  $B$ .
```

In Eq. (6.20) taking the individual terms will give rise to Eq. (6.21) useful for implementation perspective. Denoting $\frac{|ABSOLUTE_POS_B|}{|POS_{C^h}|}$ as the gamma component arising out of absolute positive region objects as $\gamma_{ABSOLUTE}$, Eq. (6.20) becomes,

$$\gamma_{B^*} = \gamma_{ABSOLUTE} + \gamma'_{B^*} \quad (6.21)$$

Section 6.3.2 gives the algorithm Improved Modified Quick Reduct (IMQRA) based on the results obtained in this section.

6.3.2 Improved Modified Quick Reduct Algorithm (IMQRA)

IMQRA algorithm (given in Algorithm 12) is MQRA algorithm with incorporation of absolute positive region removal at each iteration of MQRA. Eq. (6.21) is used for updating of gamma measure.

6.3.3 Modeling of IMQRA computations using Vector Operations

The implementation of IMQRA is done in Matlab [81] environment. Using Eq. (6.16) for computation of positive region implies that the computations need to be

done with each object in isolation. In this section theory is developed so that the required computations for positive region are modeled as matrix and sub matrix operations to obtain computational gains using Matlab like environments.

The implementation level aspects of IMQRA has been demonstrated by using following fuzzy similarity relation R_{a_n} for any quantitative attribute $a_n \in C^n$ given in Eq. (6.22) as defined in [20] (examples of more fuzzy similarity relations can be found in [65]) and its equivalent form is provided in Eq. (6.23). Generalized Lukasiewicz t-norm [155] given in Eq. (6.24) is used for applying Eq. (6.9).

$$R_{a_n}(x, y) = \max \left(\min \left(\frac{a_n(x) - a_n(y) + \sigma_{a_n}}{\sigma_{a_n}}, \frac{a_n(y) - a_n(x) + \sigma_{a_n}}{\sigma_{a_n}} \right), 0 \right) \forall x, y \in U \quad (6.22)$$

where σ_{a_n} is the standard deviation of $a_n(.)$ of quantitative attribute a_n .

$$R_{a_n}(x, y) = \begin{cases} 1 & \text{if } a_n(x) = a_n(y), \\ 1 - \frac{|a_n(x) - a_n(y)|}{\sigma_{a_n}} & \text{if } |a_n(x) - a_n(y)| < \sigma_{a_n}, \\ 0 & \text{if } |a_n(x) - a_n(y)| \geq \sigma_{a_n}. \end{cases} \quad (6.23)$$

$$\Gamma(x_1, x_2, \dots, x_n) = \max \left(0, \left(\sum_{i=1}^n x_i \right) - n + 1 \right) \quad x_i \in [0, 1]. \quad (6.24)$$

As specified earlier, standard fuzzy negation is used for positive region computation as per the result of Lemma 6.3.1. The fuzzy similarity matrix for R_a for $a \in C^h \cup \{d\}$ is represented as a matrix of dimensions $|U| \times |U|$ representing $R_a(x, y) \forall x, y \in U$. It is to be noted that row of R_a corresponding to an object $x \in U$ represents the fuzzy set $R_{a,x}(R_{a,x} - \text{foreset})$. Computation of R_a either by using Eq. (6.22) for quantitative attributes or by using Eq. (6.8) for qualitative attributes and computation of R_B using Eq. (6.24) with collection of R_a ($a \in B$) matrices are implemented as matrix operations.

The task of evaluating $POS_B(x) \forall x \in U$ is based on another matrix named as Implicator value matrix $\mathfrak{S}_B$. $\mathfrak{S}_B$ contains for any pair of objects $x, y \in U$ the value of $\mathfrak{S}(R_B(x, y), R_d(x, y))$ using Eq. (6.18). Let $(D_1, \dots, D_{|V_d|})$ represent the decision equivalence classes of d using indiscernibility relation IND. For any $i \in 1 \dots |V_d|$, and for any $x, y \in D_i$, $U_1(x) = U_1(y) = D_i$ and $U_2(x) = U_2(y) = U - D_i$. Hence the nature of computations involved for implicator values using Eq. (6.18) is same for all objects belonging to a decision class. Hence computing $\mathfrak{S}_B$ can be done in stages with each decision class objects D_i for i from 1 to $|V_d|$ and by performing simultaneous operations for objects of D_i at once using sub

matrix operations. For each D_i , a portion of matrix $\mathfrak{S}_B$ is constructed using two operations given below.

$$\mathfrak{S}_B(D_i, D_i) = 1 \quad (6.25)$$

$$\mathfrak{S}_B(D_i, U - D_i) = 1 - R_B(D_i, U - D_i) \quad (6.26)$$

After constructing $\mathfrak{S}_B$ in $|V_d|$ stages using two operations in Eq. (6.25), Eq. (6.26), the required $POS_B(y)$ for all $y \in U$ is constructed by applying minimum operation on each column of $\mathfrak{S}_B$ resulting in a row vector of $POS_B(y)$ for all $y \in U$. In any iteration of IMQRA, matrix R_B contains the extended similarity matrix of selected B attributes. The memory occupied by individual similarity matrices of $a \in B$ are recovered as and when attribute a is included into B . The removal of *ABSOLUTE_POS_B* objects from decision system resulting in reduced decision system is achieved by removing the rows and columns corresponding to new *ABSOLUTE_POS_B* objects obtained in an iteration from current similarity matrices R_B and R_a for all $a \in C^h - B$.

Hence the process of computing similarity matrices, positive region are modeled as either matrix or sub matrix based vector operations. Algorithms MQRA, IMQRA implemented in Matlab environment using these vector operations are named as MQRA_MW, IMQRA_MW respectively.

6.3.4 Analysis of IMQRA Algorithm

The space complexity of IMQRA algorithm is $O(|C^h| |U|^2)$ up to first iteration of IMQRA used for storing fuzzy similarity matrices R_a for $a \in C^h \cup \{d\}$, R_B , and the impicator value matrix $\mathfrak{S}_B$. In each of the remaining iterations of IMQRA, space complexity reduces as the memory occupied by similarity matrix of attribute selected into recommended reduct B is released. The space complexity further reduces in each iteration as the objects belonging to *ABSOLUTE_POS_B* are removed resulting in smaller size similarity matrices. Hence using the notations developed for Corollary 6.3.6 the space complexity of i^{th} iteration of IMQRA is $O((|C^h| - i + 1) |U_i|^2)$.

The time complexity for computing $POS_{\{a\}}$ is $O(|U|^2)$ in the first iteration for each attribute $a \in C^h$ and hence time complexity of first iteration of IMQRA becomes $O(|C^h| |U|^2)$. Owing to removal of absolute positive region objects, the time complexity of i^{th} iteration of IMQRA is $O((|C^h| - i + 1) |U_i|^2)$ using the notations developed for Corollary 6.3.6. Hence the theoretical time complexity of

Table 6.1: Example Decision System

Object	A	B	C	D	E	d
x1	3.59	3.52	2.86	0.76	1.30	1
x2	1.97	8.31	7.57	0.54	5.69	2
x3	2.51	5.85	7.54	5.31	4.69	2
x4	6.16	5.50	3.80	7.79	0.12	1
x5	4.73	9.17	5.68	9.34	3.37	1

Table 6.2: Fuzzy sets $R_{\{A\},x1}$, $R_{d,x1}$

	x1	x2	x3	x4	x5	
$R_{\{A\},x1}$	1	0.099	0.417	0	0.2757	
$R_{d,x1}$	1	0	0	1	1	

IMQRA is $O\left(|C^h|^2 |U|^2\right)$ but in practice it will be much smaller.

6.4 Illustration

This section contains an illustration of the concepts proposed using a simple decision system consisting of quantitative conditional attributes. Consider the decision system given in Table 6.1.

The standard deviation values (σ) of conditional attributes are needed for fuzzy tolerance relation given in Eq. (6.22). The σ values for A, B, C, D, E attributes are 1.6981, 2.2750, 2.1411, 4.0084, 2.3134 respectively. The similarity matrix for attribute A is given below summarizing the fuzzy tolerance relation values using A attribute.

$$R_{\{A\}} = \begin{bmatrix} 1 & 0.099 & 0.417 & 0 & 0.2757 \\ 0.099 & 1 & 0.682 & 0 & 0 \\ 0.417 & 0.682 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0.1579 \\ 0.2757 & 0 & 0 & 0.1579 & 1 \end{bmatrix}$$

6.4.1 Positive Region computation using conventional approaches

In this section computation of $POS_{\{A\}}$ is demonstrated using the theory given in section 6.2. Considering y as $x1$, required fuzzy sets $R_{\{A\},x1}$, $R_{d,x1}$ for applying Eq.(6.16) to compute $POS_{\{A\}}(x1)$ are depicted in Table 6.2.

Using Kleene-Dienes's implication [19], the computation of $POS_{\{A\}}(x1)$ is

Table 6.3: Implication Value Matrix $\mathfrak{S}_{\{A\}}$ Computation

Step	State of $\mathfrak{S}_{\{A\}}$
$\mathfrak{S}_{\{A\}}(D_1, D_1) = 1$ <i>and</i> $\mathfrak{S}_{\{A\}}(U - D_1, D_1) = 1 - R_{\{A\}}(U - D_1, D_1)$	$\begin{bmatrix} 1 & 0 & 0 & 1 & 1 \\ 0.901 & 0 & 0 & 1 & 1 \\ 0.583 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 \end{bmatrix}$
$\mathfrak{S}_{\{A\}}(D_2, D_2) = 1$ <i>and</i> $\mathfrak{S}_{\{A\}}(U - D_2, D_2) = 1 - R_{\{A\}}(U - D_2, D_2)$	$\begin{bmatrix} 1 & 0.901 & 0.583 & 1 & 1 \\ 0.901 & 1 & 1 & 1 & 1 \\ 0.583 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$

given below.

$$\begin{aligned}
 POS_{\{A\}}(x1) &= R_{\{A\}} \downarrow R_{d,x1}(x1) \\
 &= \inf_{x \in U} \mathfrak{S}(R_{\{A\}}(x, x1), R_{d,x1}(x)) \\
 &= \inf_{x \in U} \max(1 - R_{\{A\}}(x, x1), R_{d,x1}(x)) \\
 &= \min(1, 1 - R_{\{A\}}(x2, x1), 1 - R_{\{A\}}(x3, x1), 1, 1) \\
 &= \min(0.901, 0.583, 1, 1) \\
 &= 0.583.
 \end{aligned}$$

Similarly positive region membership values for other objects are computed as,

$$POS_{\{A\}}(x2) = 0.901$$

$$POS_{\{A\}}(x3) = 0.583$$

$$POS_{\{A\}}(x4) = 1.0$$

$$POS_{\{A\}}(x5) = 1.0$$

6.4.2 Positive region computation using vector operations

The computation of $POS_{\{A\}}$ using vector operation modeling developed in Section 6.3.3 is demonstrated in this section. The natural negation generated by Kleene-Diene's implication is standard negation. Hence based on the theory developed in Section 6.3.3, computing $POS_{\{A\}}$ using Eq. (6.25), Eq. (6.26) will result in the same results obtained in Section 6.4.1. The equivalence classes of decision attribute d are $D_1 = \{x1, x4, x5\}$ and $D_2 = \{x3, x4\}$. Implication values matrix $\mathfrak{S}_{\{A\}}$ is initially taken as a zero matrix. Steps involved in applying Eq. (6.25), Eq. (6.26) for D_1, D_2 objects as sub matrix operations are given in Table 6.3.

Column wise minimum values in resulting $\mathfrak{S}_{\{A\}}$ are the required $POS_{\{A\}}$ membership values. Hence $POS_{\{A\}}$ obtained using vector operation approach is,

$$POS_{\{A\}}(x1) = 0.583$$

$$POS_{\{A\}}(x2) = 0.901$$

$$POS_{\{A\}}(x3) = 0.583$$

$$POS_{\{A\}}(x4) = 1.0$$

$$POS_{\{A\}}(x5) = 1.0$$

It can be seen that the result is same as obtained in Section 6.4.1.

6.4.3 Illustration of IMQRA_MW algorithm

The input parameters for IMQRA_MW algorithm are specified in Section 6.3.3. The first step in IMQRA_MW algorithm is to compute similarity matrices for all conditional attributes. The similarity matrix for C^h is computed using attribute similarity matrices by applying Eq. (6.9) using generalized Lukasiewicz t-norm. The resulting R_{C^h} is given below.

$$R_{C^h} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Computing POS_{C^h} as demonstrated in Section 6.4.2 results in,

$$POS_{C^h}(x1) = 1$$

$$POS_{C^h}(x2) = 1$$

$$POS_{C^h}(x3) = 1$$

$$POS_{C^h}(x4) = 1$$

$$POS_{C^h}(x5) = 1$$

Hence, $|POS_{C^h}| = 5$.

In the first iteration of IMQRA_MW the gamma measure for each conditional attribute is evaluated. As an example consider the evaluation of $\gamma_{\{A\}}$. Using $POS_{\{A\}}$ evaluated in Section 6.4.2, $\gamma'_{\{A\}} = \frac{|POS_{\{A\}}|}{|POS_{C^h}|} = \frac{4.067}{5} = 0.8134$. Initially, $\gamma_{ABSOLUTE} = 0$ and hence it follows that $\gamma_{\{A\}} = 0.8134$ using Eq. (6.21). Similar

computation result in gamma measure for other attributes as $\gamma_{\{B\}} = 0.4127, \gamma_{\{C\}} = 0.924, \gamma_{\{D\}} = 0.8687, \gamma_{\{E\}} = 0.8282$.

Hence attribute C is added into recommended reduct B . Since $POS_C(x1) = 1, POS_C(x2) = 0.8827, POS_C(x3) = 0.8687, POS_C(x4) = 1, POS_C(x5) = 0.8687$, objects $\{x1, x4\}$ are added to $ABSOLUTE_POS_B$ and $\gamma_{ABSOLUTE} = 0.4$.

The memory allotted for similarity matrix of C attribute is released and rows and columns corresponding to $ABSOLUTE_POS_B$ objects are removed from similarity matrix of recommended reduct B and similarity matrices of the available attributes $\{A, B, D, E\}$. In the next iteration all computations are performed on the reduced hybrid decision system and attribute A is added into recommended reduct B as it gives the maximum gamma gain. The positive region memberships in the reduced HDT i.e., HDT' by $\{C, A\}$ is $POS_{\{C,A\}}(x2) = 1, POS_{\{C,A\}}(x3) = 1, POS_{\{C,A\}}(x5) = 1$. Hence, $\gamma'_{\{C,A\}} = \frac{3}{5} = 0.6$ and using Eq. (6.21), $\gamma_{\{C,A\}} = 1$. As the terminating criteria of IMQRA_MW is satisfied algorithm returns $\{C, A\}$ as the FDSR for HDT .

6.5 Non persistent Sampling based Scalable approach for Fuzzy Decision Reduct

This section develops a scalable algorithm for fuzzy decision reduct suitable for very large datasets. The complexity analysis of IMQRA in subsection 6.3.4 establishes that space and time complexity requirements of IMQRA are much higher than that of reduct algorithms for complete and incomplete symbolic decision systems. Particularly space requirements of $|C^h| + 2$ fuzzy similarity matrices of $|U| \times |U|$ dimensions needed in first iteration of IMQRA can become a bottleneck for the system for large and very large HDS. Scalability becomes a pertinent issue when dealing with reduct computation in HDT. In this section reduct algorithm for IMQRA is proposed using sampling based approach with the objective achieving scalability. In Chapter 4 it is established that non persistent stratified sampling based approach has resulted in better reduct computation algorithm for symbolic decision systems in terms of reducing Excess Count, Computational Time, Reduct Length. Hence using non persistent stratified sampling approach is used in this section for developing algorithm NPSS_IMQRA for enabling scalable fuzzy decision reduct computation with much lesser time and space complexities than algorithm IMQRA.

6.5.1 Relevance of Sampling

In Chapter-4 relevance of sampling approach for reduct computation in symbolic decision system is discussed. It is pointed that sampling based attribute selection can result in an appropriate attribute selection for the decision system as long as the sampling approach has the ability to limit the occurrence of *false POS* granules and lost granules. In this subsection the relevancy of sampling for attribute selection using fuzzy rough set approach is discussed.

The basic blocks of computation in fuzzy rough set model are fuzzy similarity matrices. To achieve comparability in the computations in sample based attribute selection with computations in the whole dataset, similar computation of similarity matrices need to be obtained in sample datasets. i.e., the fuzzy similarity membership value of a pair of objects in sampled dataset need to be same as it is in global dataset. The similarity between objects based on qualitative attributes are unaltered in sampled datasets. For quantitative attributes, this is possible only if all the other parameters present in the fuzzy similarity relation are evaluated at global level and are used in computing fuzzy similarity relation for sampled dataset objects. For example in the fuzzy similarity relation for quantitative attributes given in Eq. (6.22), the global parameter is standard deviation value σ . As computing σ for an attribute in HDT is of linear complexity and computationally feasible such global parameters are evaluated for the given decision system and are used in computing fuzzy similarity matrices in sampled datasets.

Let $SHDT = (U^s, C^h = C^c \cup C^n, \{d\}, \{V_{a_c}, f_{a_c}\}_{a_c \in C^c \cup \{d\}}, \{V_{a_n}, f_{a_n}\}_{a_n \in C^n})$ be the sample hybrid decision system generated using stratified sampling from HDT . Since the parameters required for fuzzy similarity relation are computed for each quantitative attribute $a \in C^n$ at global level, the fuzzy similarity matrix $R_{\{a\}}^s$ in $SHDT$ is going to be sub matrix of $R_{\{a\}}$ in HDT . As this property is trivially holds for similarity matrices of categorical attributes $a \in C^c$, the fuzzy similarity matrix R_B^s for $B \subseteq C^h$ in $SHDT$ is a sub matrix of R_B in HDT , i.e.,

$$R_B^s(x, y) = R_B(x, y) \quad \forall x, y \in U^s \quad (6.27)$$

Let $o \in U^s$ be an object of $SHDT$. The following properties are satisfied because $SHDT$ is a sampled dataset from HDT .

$$U_1^s(o) \subseteq U_1(o) \quad (6.28)$$

$$U_2^s(o) \subseteq U_2(o) \quad (6.29)$$

Now consider the impact of sampling on positive region membership computation of o in *SHDT*. For any qualitative attribute $a \in C^c$, $POS_{\{a\}}^s(o)$ can be ‘1’ but $POS_{\{a\}}(o)$ can be actually ‘0’. This can happen when there exists an inconsistent object $o' \in U_2(o)$, i.e., $f_a(o) = f_a(o')$ but their decision classes are different, but $o' \notin U_2^s(o)$. This corresponds to occurrence of *false POS* granules discussed in Chapter 4. But such discrete jump of 1 to 0 in positive region membership between sampled and original dataset does not occur for quantitative attributes. Let $a \in C^n$ and using Eq. (6.18) for computing $POS_{\{a\}}^s(o)$,

$$\begin{aligned}
POS_{\{a\}}^s(o) &= \min_{y \in U_2^s(o)} (1 - R_{\{a\}}^s(o, y)) \\
&= \min_{y \in U_2^s(o)} (1 - R_{\{a\}}(o, y)) \\
&\geq \min_{y \in U_2(o)} (1 - R_{\{a\}}(o, y)) \\
&= POS_{\{a\}}(o)
\end{aligned}$$

Hence the positive region membership of an object in *SHDT* is higher than that of *HDT* but the change is gradual in nature owing to a being quantitative attribute. As the proposed non persistent sampling approach is based on stratified sampling, there is a good chance that the value range of a decision class objects for attribute a in *HDT* is preserved or approximately preserved in *SHDT*. Hence it is highly probable that $POS_{\{a\}}^s(o), POS_{\{a\}}(o)$ are nearby values. Extending this property to a subset of attributes, $B \subseteq C^h$ results in,

$$POS_B^s(o) \geq POS_B(o) \quad (6.30)$$

As B is a mix of quantitative and qualitative attributes, the values of $POS_B^s(o), POS_B(o)$ are expected to be similar in nature. Hence the γ_B^s can be a good approximation of γ_B . The nature of *HDT* is represented more precisely in *SHDT* also because of non persistent nature as the sampled datasets for each attribute selection are generated with replacement from *HDT*. Hence sampling based approach is relevant for reduct computation in hybrid decision systems using fuzzy rough set model. Next subsection gives the proposed NPSS_IMQRA algorithm.

6.5.2 NPSS_IMQRA algorithm

NPSS_IMQRA algorithm follows the structure of IMQRA algorithm and uses sampled hybrid decision systems for selection of attributes. Algorithm starts with B initialized to empty set. In an iteration of NPSS_IMQRA, NoS number of stratified sample hybrid decision systems ($SHDT$ s) are generated from HDT where sampling of objects is done with replacement. The `getBestAttr` algorithm is applied on each $SHDT$ to perform a single iteration of IMQRA for selection of next best attribute into B . `getNextAttr` algorithm uses already selected attributes B to compute R_B^s , POS_B^s , $ABSOLUTE_POS_B^s$. After removal of $ABSOLUTE_POS_B^s$ objects if $SHDT$ becomes empty, indicating B as FDSR for $SHDT$, then algorithm returns -1 as no new attribute can be selected. Similarly after performing the iteration of IMQRA if no gamma gain is observed then also algorithm returns -1 as no decision is reached in the process. Practically this situation also refers to the almost possibility of (even though theoretical exceptions can be there) B being $FDSR$ for $SHDT$. In the presence of gamma growth a new attribute giving the maximum gamma growth is returned by algorithm `getBestAttr` to be included into B in algorithm NPSS_IMQRA.

Hence in an iteration of NPSS_IMQRA, at most NoS number of attributes are selected into B^s . In case of at least a single attribute inclusion into B^s the global validation step is performed. In the global validation step, fuzzy similarity matrix R_{B^s} is computed in HDT using equation (6.9). The attributes B^s are included into B and R_B is updated by combining existing R_B with R_{B^s} using equation (6.9). Positive region POS_B is computed and the new set of absolute positive region objects are determined and included into $ABSOLUTE_POS_B$ objects. The newly included absolute positive region objects are removed from HDT and the corresponding rows and columns of R_B are removed. The memory used for R_{B^s} is released. Algorithm continues till the required terminating criteria is reached and finally returns B as the fuzzy M-decision super reduct with degree α .

Algorithm 13: NPSS_IMQRA

Input : $HDT = (U, C^h = C^c \cup C^n, \{d\}, \{V_{a_c}, f_{a_c}\}_{a_c \in C^c \cup \{d\}}, \{V_{a_n}, f_{a_n}\}_{a_n \in C^n})$,
 R : Fuzzy similarity relation,
 N : Fuzzy Negation,
 Γ : t-norm for computing R_B for any $B \subseteq C^h$ using equation (6.9)
 α : Degree for terminating criteria
 SS : Sample Size (in Percentage)
 NoS : Number of Samples Datasets in an Iteration
 $POSFlag$: Boolean flag for computing $|POS_{C^h}|$
Output: B : Fuzzy M -Super reduct with degree α

```
1 Compute parameters required for  $R_a$  for all  $a \in C^n$ 
2 if  $POSFlag = true$  then
3   | Compute  $POS_{C^h}, |POS_{C^h}|$ .
4 end
5 else
6   |  $|POS_{C^h}| = |U|$ 
7 end
8  $B = \emptyset$ 
9  $\gamma_B = 0$ 
10  $ABSOLUTE\_POS_B = \emptyset$ ,
11  $\gamma_{ABSOLUTE} = 0$ 
12  $\gamma'_B = 0$ 
13 while  $\gamma_B < \alpha$  do
14    $B^s = \emptyset$  /* Attributes selected in the current iteration */
15   for  $i = 1$  to  $NoS$  do
16      $SHDT =$ 
17        $(U^s, C^h = C^n \cup C^c, \{d\}, \{V_{a_c}, f_{a_c}\}_{a_c \in C^c \cup \{d\}}, \{V_{a_n}, f_{a_n}\}_{a_n \in C^n}) = \text{getStratifiedSample}(HDT, SS)$ 
18      $a = \text{getBestAttr}(SHDT, B \cup B^s)$ 
19     if  $a \neq -1$  then
20       |  $B^s = B^s \cup \{a\}$ 
21     end
22   end
23   if  $POSFlag = false$  and  $B^s = \emptyset$  then
24     | Return  $B$  /* Modified End Condition */
25   end
26   Compute  $R_{B^s}$ 
27    $R_B = \Gamma(R_B, R_{B^s})$ 
28    $B = B \cup B^s$ 
29   Compute  $POS_B$  and  $\gamma'_B$ 
30    $\gamma_B = \gamma_{ABSOLUTE} + \gamma'_B$ 
31   Update  $ABSOLUTE\_POS_B, \gamma_{ABSOLUTE}$ .
32   Reduce  $HDT$  by removal of  $ABSOLUTE\_POS_B$  objects
33 end
34 Return  $B$ .
```

An important aspect of `getBestAttr` algorithm is selection of the attribute is only based on $|POS_{B \cup \{a\}}^s|$ but not on $\gamma_{B \cup \{a\}}^s$. This results in computational gain as computation of $|POS_{C^h}^s|$ is omitted. This omission does not impact the final selection as a single iteration of IMQRA is involved. Hence the presence or absence of denominator value in $\gamma_{B \cup \{a\}}^s$ does not impact the selection.

Algorithm 14: getBestAttr

Input : $SHDT = (U^s, C^h = C^n \cup C^c, \{d\}, \{V_{a_c}, f_{a_c}\}_{a_c \in C^c \cup \{d\}}, \{V_{a_n}, f_{a_n}\}_{a_n \in C^n})$: Sampled Hybrid Decision Table
 R : Fuzzy similarity relation
 N : Fuzzy Negation
 Γ : t-norm for computing R_B for any $B \subseteq C^h$ using equation (6.9)
 B : Set of attributes included in recommended reduct
Output: $attr$: Best attribute in $C^h - B$ to be included into B

```
1 Compute  $R_B^s, POS_B^s, ABSOLUTE\_POS_B^s$ 
2 Reduce  $SHDT$  by removing  $ABSOLUTE\_POS_B^s$  objects
3 if  $|U^s| = 0$  then
    /*  $B$  is already a reduct for  $SHDT$  */
4      $attr = -1$ 
5     Return  $attr$ 
6 end
7  $T = B, best = -1, attr = -1$ 
8 foreach  $a \in C^h - B$  do
9     if  $|POS_{B \cup \{a\}}^s| > best$  then
10          $T = B \cup \{a\}$ 
11          $best = |POS_{B \cup \{a\}}^s|$ 
12          $attr = a$ 
13     end
14 end
15 Return  $attr$ 
```

6.5.3 Analysis of NPSS_IMQRA Algorithm

Computation of POS_{C^h}

The computation of POS_{C^h} is required in NPSS_IMQRA algorithm to be used in terminating criteria of the algorithm. The computation of POS_{C^h} needs computing fuzzy similarity matrix R_{C^h} . Constructing and storing fuzzy similarity matrices $R_{\{a\}}$ for each $a \in C^h$ makes space complexity of NPSS_IMQRA to be same as that of IMQRA and cannot be afforded as NPSS_IMQRA is intended for large HDT. The approach used is to initialize R_{C^h} to unity matrix and by sequentially computing $R_{\{a\}}$ and integrating with R_{C^h} using t-norm. Hence in this approach the necessity arises to maintain two fuzzy similarity matrices at global level of dimensions $|U| \times |U|$. After computation of R_{C^h} and from it POS_{C^h} and $|POS_{C^h}|$, the memory occupied by R_{C^h} is released. In first iteration, computation of R_B and updation using B^s attributes can be done in a similar approach which requires two matrices. The space complexity reduces in the successive iterations owing to absolute positive region removal. Hence the space complexity of NPSS_IMQRA is $O(|U|^2)$ which is a huge reduction in comparison to space complexity of IMQRA i.e., $O(|C^h| |U|^2)$.

If depending on the size of the dataset and the processing environment's limitations if computation of POS_{C^h} is to be avoided then $POSFlag$ is set to false and it is assumed that HDT is consistent and $|POS_{C^h}|$ is set to the maximum pos-

sible value $|U|$. In the current approach because of usage of t-norm in updating fuzzy similarity matrix and the positive region membership of each object grows gradually from zero to one as iterations progresses. Hence the change in $|POS_B|$ is expected moving from one iteration to other iteration till the maximum possible $|POS_B|$ value is reached. Hence in a deterministic algorithm like IMQRA the terminating criteria#2 can be adopted as the situation with zero gamma gain with any of the available attributes. As the selection of the attributes in NPSS_IMQRA is done using SHDT's the terminating criteria#2 is taken as the situation when no attribute is included into B^s after invoking `getBestAttr` on NoS samples.

In case the algorithm NPSS_IMQRA is used for computing FDSR for HDT by setting degree $\alpha = 1$ then another computationally feasible option exists for computing POS_{C^h} . The possibility involved is a hybrid of previous two approaches discussed. Algorithm starts with $POSFlag$ set to false. After few iterations due to absolute positive region removal the size of HDT reduces. According to system constraints a threshold can be fixed such that after HDT size is less than the threshold, evolving from current R_B the computation of R_{C^h} can be computed and POS_{C^h} for non absolute positive region objects (current HDT objects) is computed and augmented with $ABSOLUTE_POS_B$.

Complexity analysis of NPSS_IMQRA

As discussed in earlier sub sub section, the space complexity of NPSS_IMQRA is $O(|U|^2)$. Let U_i denote the number of objects in HDT in i^{th} iteration of NPSS_IMQRA. Let $\epsilon = \frac{SS}{100}$. The size of stratified sampled dataset SHDT is $|U^s| = \epsilon |U_i|$. Hence the time complexity of local computations in i^{th} iteration is $O(NoS \times |C^h - i| (\epsilon |U_i|)^2)$. The time complexity of global validation step in i^{th} iteration is $O(|B^s| |U_i|^2)$. Hence the time complexity of i^{th} iteration of NPSS_IMQRA is $O(\max(NoS \times |C^h - i| (\epsilon |U_i|)^2, |B^s| |U_i|^2))$. If k denote the number of iterations of NPSS_IMQRA, i.e., $(|B| < NoS \times k)$, then the time complexity of NPSS_IMQRA is

$$\sum_{i=1}^k O(\max(NoS \times |C^h - i| (\epsilon |U_i|)^2, |B^s| |U_i|^2))$$

which is having $O(|C^h|^2 |U|^2)$ as a very high upper bound.

6.6 Experiments and Results

The details of hybrid decision systems used in the experiments are described in Table-6.4. All the datasets are from UCI Machine Learning repository [11] ex-

Table 6.4: Details of Hybrid Datasets

Dataset	Objects	Conditional Features			Decision Classes
		Total	Quantitative	Qualitative	
Image Segmentation	2310	19	16	3	7
Sonar, mines vs. rocks	208	60	60	0	2
Wisconsin diagnostic breast cancer (WDBC)	569	30	30	0	2
Cleveland	297	13	5	8	5
Glass	214	9	9	0	6
Ionosphere	230# 351*	34	33	1	2
Web	149	2556	2556	0	5
Wine	178	13	13	0	3
Heart	270	13	5	8	2
Olitos	120	25	25	0	4
Spambase	4601	57	57	0	2

#Size of the dataset used in [58] *Size of the dataset used in present work [11]

Table 6.5: Comparison Results of IMQRA_MW and MQRA_MW with FA-FSCE and FSCE algorithms

Dataset	FSCE Results		FA-FSCE Results		MQRA_MW Results		IMQRA_MW Results	
	Reduct size	Time	Reduct size	Time	Reduct size	Time	Reduct size	Time
Image Segmentation	17	1258.0468	17	900.5781	16	41.75	16	17.4
Sonar, Mines vs Rocks	41	300.5625	41	50	5	0.57	5	0.69
WDBC	27	228.9218	27	171.875	6	1.7257	6	1.3469

cepting Web dataset from [53] and Olitos from [21]. Sub section (6.6.1) describes the experiments performed using IMQRA algorithm. Sub section (6.6.2) contains the experiments performed using NPSS_IMQRA algorithm.

6.6.1 Experiments using IMQRA_MW algorithm

Experiments conducted using IMQRA_MW, obtained the same reducts (for example with Spambase dataset) and reduct lengths with the results for MQRA reported in [20]. The difference in results for MQRA_MW, IMQRA_MW with MQRA in [20] is in terms of computational time. But in results reported in [20] the computational time is not reported. Hence comparative analysis of proposed algorithms with results given in [20] for MQRA algorithm could not be performed.

The FA-FSCE algorithm given in [109] uses positive region object removal (forward acceleration) in Hu's fuzzy rough set model [45] and FSCE algorithm is the version of FA-FSCE without positive region removal. Hence FSCE is comparable to MQRA_MW and FA-FSCE is comparable to IMQRA_MW algorithm. A comparative analysis is done for IMQRA_MW and MQRA_MW algorithms with the results reported for FSCE and FA-FSCE algorithms in [109]. Table 6.5 contains the results for FSCE and FA-FSCE algorithms reported in [109] along with the results obtained for MQRA_MW and IMQRA_MW algorithms.

In [58] Jensen et al., have described new approaches for fuzzy rough feature selection in improvement to the earlier FRFS algorithm [56]. The algorithms described in [58] are fuzzy boundary region based feature selection (B-FRFS), fuzzy

Table 6.6: Reduct Size Comparison with FRFS and associated algorithms

Dataset	FRFS	B-FRFS	L-FRFS	FDM	MQRA_MW and IMQRA_MW
Cleveland	11	9	9	9	8
Glass	9	9	10	9	9
Heart	11	8	8	8	7
Ionosphere	11	9	9	8	7
Olitos	10	6	6	6	5
Web	24	20	21	18	20
Wine	10	6	6	6	5

Table 6.7: CT (in seconds) Comparison with FRFS and associated algorithms

Dataset	FRFS	B-FRFS	L-FRFS	FDM	MQRA_MW	IMQRA_MW
Cleveland	24.11	8.78	3.32	10.68	0.37	0.5
Glass	1.61	3.3	1.53	4.88	0.14	0.28
Heart	11.84	3.61	2.17	8.77	0.24	0.31
Ionosphere	61.8	8.53	3.77	17.54	1.08	0.84
Olitos	11.2	1.29	0.72	4.07	0.17	0.25
Web	5642.65	949.69	541.65	1782.69	89.62	62.17
Wine	1.42	1.69	0.97	4.6	0.11	0.23

lower approximation based feature selection (L-FRFS) and fuzzy discernibility matrix based algorithm (FDM). All these algorithms are forward selection based quick reduct like algorithms based on Dubois-Prade’s fuzzy rough set model[25]. Hence a comparative analysis is done with MQRA_MW, IMQRA_MW algorithms with these algorithms. The results reported in [58] along with results obtained for MQRA_MW, IMQRA_MW algorithms are given in Table 6.6 and Table 6.7. Table 6.6 contains the results pertaining to reduct sizes and Table 6.7 contains the results pertaining to computational time in seconds.

6.6.2 Experiments using NPSS_IMQRA algorithm

The result of NPSS_IMQRA and IMQRA_MW can be different as different selection procedures are used. The objective of the experiments in this section is to investigate the applicability of FDSR obtained using NPSS_IMQRA for classification and also the reduct size and computational time in comparison to the result of IMQRA_MW. The results are compared with the results reported for MQRA in [20] in the aspects of reduct size and classification performance. The design of experiments also follows the methodology used in [20].

In [20] a tenfold cross validation experiment is performed using MQRA algorithm using 1NN classifier. In order to perform comparative analysis the same experiment is used and described here. In each trail 9 folds are considered as training data and one fold is considered as test data. FDSR is obtained by applying MQRA, NPSS_IMQRA on the training data at degrees $\alpha = 1, 0.9$. Using

Table 6.8: Classification Results in Tenfold Experiment

	MQRA Results[20]			NPSS_IMQRA Results		
Dataset	Unred.	γ		Unred.	γ	
		1	0.95		1	0.95
Cleveland	51.56	49.23	45.85	50.34	48.27	49.04
Glass	70.52	70.52	70	74.76	74.76	71.43
Ionosphere	86.09	89.13	86.52	88	88.29	87.43
Pima Indians	73.45	71.15	75.24	70.79	70.79	67.76
Spambase	*	*	*	90.85	90.35	88.78
Web	40.95	42.95	42.86	47.85	42.85	40.71
Wine	94.97	92.71	91.57	95.88	91.18	90

*: Not Reported in [20].

Table 6.9: Reduct Sizes in Tenfold Experiment

	MQRA Results[20]			NPSS_IMQRA Results		
Dataset	Unred.	γ		Unred.	γ	
		1	0.95		1	0.95
Cleveland	13	7.7	5.00	13	8	5.5
Glass	9	9	7	9	8.9	6.6
Ionosphere	34	7.1	5.1	34	7.5	5.4
Pima	8	7.5	5	8	8	5.5
Spambase	57	*	*	57	48.1	16.4
Web	2556	18.4	14.9	2556	21.8	18.4
Wine	13	5	3.9	13	5.1	4

*: Not Reported in [20].

1NN classifier the performance is evaluated over test data using all attributes, FDSR attributes from each algorithm and the average results are obtained. As the FDSR obtained using MQRA, MQRA_MW, IMQRA_MW are same, in this section comparative analysis is only done between MQRA results reported in [20] and NPSS_IMQRA results. The parameters used for NPSS_IMQRA algorithm are NoS=3 and SS=20%. Table 6.8 and Table 6.9 contains the tenfold experiment results with NPSS_IMQRA along with the results reported in [20]. Table 6.8 contains the average classification performance results and Table 6.9 contain the average reduct sizes obtained in tenfold experiment.

A tenfold experiment using IMQRA_MW is conducted using the same folds used for NPSS_IMQRA and the obtained results are summarized in Table 6.10. Table 6.10 contains the results as (mean $\pm$ standard deviation) of computational time in seconds, reduct length.

In [20] detailed results were given for MQRA algorithm on spambase dataset. Table IV in [20] contains the attributes selected in each iteration and resulting gamma after each iteration of MQRA. Current implementation of MQRA (i.e., MQRA_MW) has given the same result as reported in [20]. Table 6.11 gives the

Table 6.10: CT in tenfold experiments for IMQRA_MW and NPSS_IMQRA algorithms

Dataset	Computational Time		Reduct Length	
	IMQRA_MW	NPSS_IMQRA	IMQRA_MW	NPSS_IMQRA
Image Segmentation	13.956±0.097	5.024±0.145	16.4±0.84	17.0±1.054
Sonar, Mines vs Rocks	0.4±0.01	0.246±0.013	5.2±0.422	6.1±0.32
WDBC	0.99±0.02	0.48±0.02	5.8±0.422	6.5±0.527
Cleveland	0.22±0.001	0.14±0.003	7.7±0.48	8.2±0.632
Glass	0.096±0.00	0.08±0.003	9.0±0.0	8.9±0.316
Ionosphere	0.58±0.03	0.26±0.006	7.2±0.633	7.3±0.68
Spambase	*	66.91±3.46	*	44.3±0.823
Web	50.64±2.82	45.44±3.22	18.7±1.06	22.3±1.49
Wine	0.063±0.001	0.062±0.002	5.0±0.0	5.1±0.32
Heart	0.135±0.007	0.078±0.005	7.1±0.32	8.0±0.47
Olitos	0.122±0.002	0.114±0.005	5.0±0.0	5.8±0.42

iteration wise result obtained with NPSS_IMQRA with that of MQRA reported in [20].

Table 6.11: Detailed comparative Results with Spambase dataset

Size	MQRA Result[20]		NPSS_IMQRA Result		Size	MQRA Results[20]		NPSS_IMQRA Result	
	Att.	γ	Att.	γ		Att.	γ	Att.	γ
1	26	0.05	26	0.057709	26	0	0.99	0	0.99115
2	24	0.1	24	0.10919	27	48	0.992	48	0.99245
3	11	0.16	6	0.141	28	7	0.993	7	0.99342
4	18	0.26	20	0.24407	29	23	0.994	23	0.99437
5	20	0.41	56	0.36814	30	17	0.995	17	0.99511
6	56	0.55	18	0.50186	31	34	0.995	34	0.99579
7	49	0.67	51	0.60586	32	47	0.996	47	0.99643
8	51	0.77	11	0.722	33	25	0.996	25	0.99695
9	4	0.82	49	0.81325	34	22	0.997	22	0.99741
10	44	0.87	44	0.86048	35	5	0.997	5	0.99778
11	45	0.9	45	0.88898	36	21	0.998	21	0.99809
12	2	0.91	4	0.916	37	53	0.998	53	0.99839
13	1	0.93	15	0.92858	38	8	0.998	8	0.99868
14	15	0.94	1	0.93985	39	38	0.998	38	0.99896
15	32	0.95	41	0.94638	40	19	0.999	19	0.9992
16	52	0.95	52	0.95504	41	40	0.999	40	0.99944
17	9	0.96	32	0.96321	42	12	0.999	12	0.99966
18	41	0.97	9	0.96877	43	30	0.999	30	0.99987
19	55	0.97	55	0.97333	44	35	0.999	35	0.99998
20	43	0.97	43	0.97693	45	27	1	27	1
21	50	0.98	2	0.98033					
22	36	0.98	50	0.98312					
23	13	0.98	36	0.98545					
24	54	0.98	13	0.9876					
25	6	0.98	54	0.98955					

Figure 6.1: Computational Gain Percentage of IMQRA_MW over FSCE, FA-FSCE, MQRA_MW algorithms

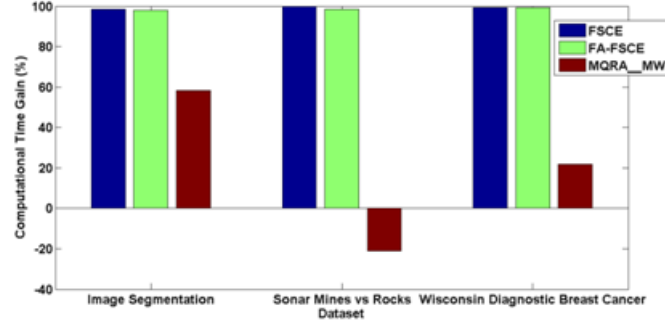
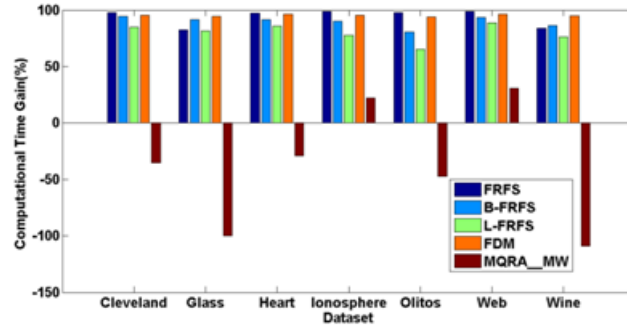


Figure 6.2: Computational Gain Percentage of IMQRA_MW over FRFS, B-FRFS, L-FRFS, FDM, MQRA_MW algorithms



6.7 Analysis of Results

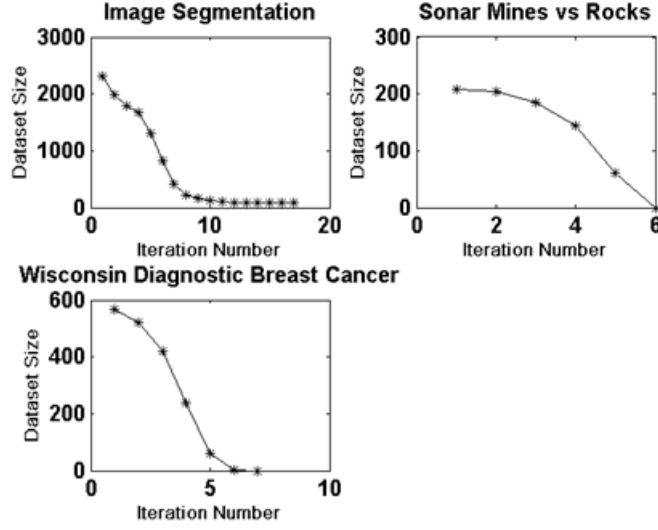
In this section results obtained with IMQRA_MW algorithm and NPSS_IMQRA algorithm are analyzed in subsections (6.7.1) and (6.7.2).

6.7.1 Analysis of IMQRA_MW Results

The computational gain percentages obtained in IMQRA_MW over FA-FSCE, FSCE and MQRA_MW algorithms is given in Fig.6.1 and over FRFS and associated algorithms is given in Fig.6.2. From Fig.6.1 it is inferred that IMQRA_MW achieved over 98% computational gain over FSCE, FA-FSCE algorithms. Similarly from Fig.6.2 it is inferred that IMQRA_MW has achieved over 60-99% computational gains over FRFS, B-FRFS, L-FRFS, FDM algorithms. The highly significant computational gains illustrate the importance of vector based implementation and absolute positive region removal incorporated in IMQRA_MW.

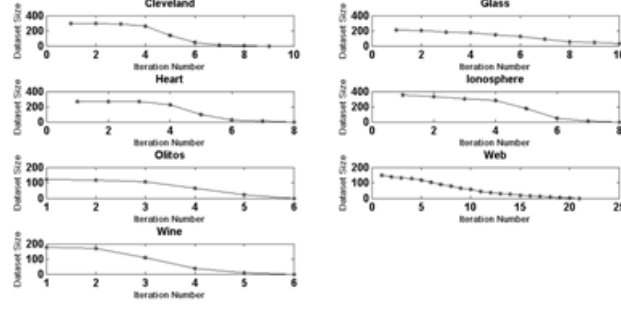
It is observed that incorporating vector based implementation in MQRA resulting in MQRA_MW algorithm has itself resulted in significant computational gains by MQRA_MW over FA-FSCE, FSCE algorithms and also FRFS associ-

Figure 6.3: Reduction in Dataset Size in IMQRA_MW iterations in FA-FSCE comparative Experiment



ated algorithms. In comparison between MQRA_MW, IMQRA_MW algorithms mixed results can be noticed. The gain due to removal of absolute positive region is sensitive to the nature of the datasets. Fig.6.3 and Fig.6.4 contain the iteration wise dataset size used in IMQRA_MW algorithm owing to absolute positive region removal for datasets used in FSCE comparative experiment and FRFS comparative experiment respectively. Datasets in which MQRA_MW obtained better computational gains than IMQRA_MW (such as Sonar Mines vs Rocks, Glass, Wine) it can be observed (from Fig.6.3 and Fig.6.4) that, the quantum of absolute positive region in the initial iterations is small and also that these datasets are smaller size in nature. Compared to MQRA_MW there is a computational overhead in IMQRA_MW associated with identification of absolute positive region objects and filtering them in fuzzy similarity matrices. Hence in these datasets the computational overhead has compensated the effect of absolute positive region removal resulting in more computational time for IMQRA_MW than MQRA_MW. The other reason being that the most computational intensive iterations of IMQRA_MW are the initial iterations and in these iterations amount of dataset reduction is small in these datasets. But for large decision systems such as Image Segmentation and Web datasets the computational overhead in IMQRA_MW is significantly compensated by gains obtained in absolute positive region removal and in these two datasets IMQRA_MW has achieved significant gains over MQRA_MW. Hence IMQRA_MW is proved to be computationally efficient version of MQRA algorithm especially for large datasets.

Figure 6.4: Reduction in Dataset Size in IMQRA_MW iterations in FRFS Comparative Experiment



6.7.2 Analysis of NPSS_IMQRA Results

The 1NN classification results for tenfold experiment reported in Table 6.8 establishes that NPSS_IMQRA resulted in similar classification accuracies compared to MQRA results reported in [20]. The variability in the results reported for *Unred (all)* attributes is because of the training and test datasets in ten folds are different from that of experiment in [20].

The average reduct sizes obtained in tenfold experiment (in Table 6.9, Table 6.10) also establish that the reduct size obtained in NPSS_IMQRA is similar to that of MQRA, IMQRA_MW and in some datasets has resulted in slightly higher size. Only in Web dataset, where the reduct is obtained from 2556 attributes, NPSS_IMQRA has resulted in approximately 3 more attributes compared to that of MQRA. In all other datasets at most difference observed is that of 1 attribute. As selection of attributes done in NPSS_IMQRA is based on sampled datasets, this result proves that NPSS_IMQRA also selects relevant attributes and the relevancy is comparable to attributes selected using whole dataset as in MQRA (and hence with IMQRA_MW).

Similar observation can be seen in results with Spambase dataset (in Table 6.11). Even though the order of the attributes obtained is NPSS_IMQRA is different, close observation reveals that same attributes are selected in NPSS_IMQRA and MQRA with slight difference in attribute order.

Based on results reported in Table 6.10, NPSS_IMQRA obtained significant computational gains over IMQRA_MW (in the order of 10 to 65% gain percentages). It is to be noted that IMQRA_MW could not be applied on the spambase dataset in our experimental environment due to space complexity issue of memory overflow. NPSS_IMQRA was able to get reduct on spambase dataset due to improvements in space and time complexities discussed earlier. This example illustrates the relevance of sampling based approach for scalable computation.

6.8 Summary

The study aims at efficient implementation of algorithms for FDSR of hybrid decision systems using fuzzy rough set theory. Simplified computation model for positive region computation in Radzikowska-Kerry's fuzzy rough set model is developed. It is identified that Radzikowska-Kerry's fuzzy rough set model for decision systems can be specified using a pair of fuzzy negation and t-norm instead of an implication and t-norm. Theoretical validation for redundancy of presence of absolute positive region objects for computation of positive region for remaining objects is provided. These improvements are incorporated into MQRA as algorithm Improved MQRA (IMQRA). The computations in MQRA, IMQRA are modeled as vector based operations resulting in MQRA_MW, IMQRA_MW algorithms. MQRA_MW, IMQRA_MW have achieved significant computational gains over several existing approaches for reduct computation using fuzzy rough set theory.

A sampling based scalable approach for FDSR computation based on non persistent stratified sampling strategy described in Chapter 4 is developed as NPSS_IMQRA algorithm. NPSS_IMQRA has obtained significant computational gains over IMQRA_MW and also overcomes limitations associated with memory requirements in IMQRA_MW.

Algorithm NPSS_IMQRA takes a good stride in achieving scalability but has its own limitations. The memory requirement of at least two matrices of size $|U| \times |U|$ can still be a bottleneck without which global validation and thereby achieving reduct or super reduct becomes infeasible. This issue directs the researches to propose relevant parallel and distributed approaches retaining the computational framework given by IMQRA_MW, NPSS_IMQRA at individual node computations as future work.

Chapter 7

Seed based Fuzzy Decision Reduct for Hybrid Decision Systems

This chapter explores the adoption of classical rough set technology as one of the key components in ERCS for scalable reduct computation in incomplete symbolic and hybrid decision systems.

The motivation and overview of seed based reduct computation is given in Section 7.1. Section 7.2 describes the seed based reduct computation approach for IDS including IQRAIG_Incomplete_Seed algorithm (in Sub Section 7.2.3), experiments, results and analysis of results. Seed based approach for fuzzy decision reduct in HDS is described in Section 7.3. Section 7.3 details the approach followed for seed based reduct computation, proposed algorithm IMQRA_Seed (in Sub Section 7.3.4), experiments, results and analysis of results. Summary of the chapter is given in Section 7.4.

7.1 Complexity Hierarchy of Reduct Computation

Figure 7.1 summarizes the complexity hierarchy of reduct computation in complete symbolic, incomplete symbolic and hybrid decision systems. The complexity of reduct algorithm highly depends on the complexity of computation of positive region and in turn computation of lower approximation. In complete symbolic decision systems, positive region computation is least complex as the computation occurs on equivalence granules resulting from classical indiscernibility relation IND. As noted in Chapter 5, the complexity of computation of positive region in incomplete symbolic decision systems, can be improved by the proposed novel granular approach and a sub granule's (base of granule) belongingness to positive region can be simultaneously computed. Hence reduct computation complexity of incomplete symbolic decision systems is higher than that of complete symbolic decision systems.

Reduct computation in hybrid decision systems, using fuzzy rough set model, has the highest complexity, owing to mandatory computation of positive region belongingness for each individual object. By incorporating vector based operations in IMQRA_MW algorithm, there is a reduction in complexity but the space and

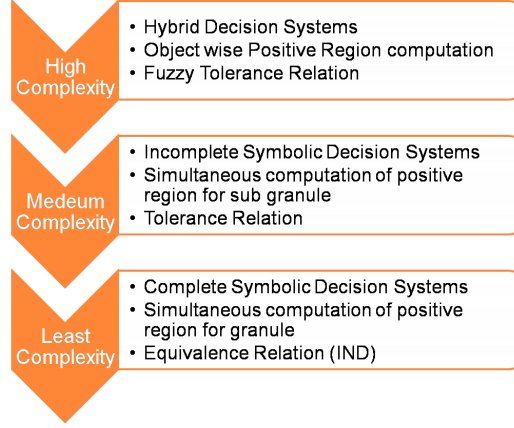


Figure 7.1: Complexity of Reduct Computation in Hybrid, Incomplete and Complete Symbolic Decision Systems

time complexities of reduct computation in HDS are much higher than those of other decision systems.

In literature, the problem of reduct computation for a type of decision system is considered and approached independent of approaches for other types of decision systems. In the methodologies proposed in this chapter, combination of approaches for different types of decision systems are proposed which effectively compute super reduct for a decision system of higher complexity.

In Chapter 3, IQRA_IG algorithm has been established as the efficient reduct computation algorithm for complete symbolic decision systems and in case size of the dataset is large then in Chapter 4, algorithm NPSS_IQRAIG proved to be a viable scalable approach for reduct computation. For medium size datasets IQRA_IG and for large decision systems NPSS_IQRAIG can result in reduct or super reduct in reasonable amount of time. In this chapter an approach is developed for computing reduct for incomplete symbolic decision systems and for hybrid decision systems by using the efficiency of IQRA_IG or NPSS_IQRAIG for complete symbolic decision systems. The proposed approaches are named as seed based approaches. In this approach, the given decision system is transformed into complete symbolic decision system. Depending on the size of the CDS either IQRA_IG or NPSS_IQRAIG is applied to obtain a subset of attributes. This subset of attributes is named as seed reduct and is given to the respective algorithm (in case DT is IDS then IQRAIG_Incomplete or HDS then IMQRA or NPSS_IMQRA) to continue till the required super reduct is obtained. The detailed aspects of the approach with respect to IDS and HDS are discussed in the further sections.

7.2 Seed based Reduct Computation in Incomplete Decision Systems

Let $DT = (U, C \cup \{d\}, (V_a, f_a)_{a \in C \cup \{d\}})$ represents an Incomplete Decision system (IDS) as described in Chapter 5. In novel granular framework for IDS described in Chapter 5, the granular space refinement occurs from $Gr(U, A)$ into $Gr(U, A \cup \{a\})$ using GrSR operator (Eq. (5.8)) where $A \subset C$ and $a \in C - A$. If a projection of refinement process occurring from $Gr(U, A)$ to $Gr(U, A \cup \{a\})$ is confined to only *base* portions of granules of $g \in Gr(U, A)$ it is observed that exactly same refinement happens as in CDS using classical indiscernibility relation. The phenomenon is extendible to successive application of GrSR operator. Hence for a given sequence of conditional attributes $a_1, \dots, a_k$, and let $g \in Gr(U, \{a_1, \dots, a_k\})$ then the following observations are arrived.

1. If $g.base$ is in positive region and $g.tail$ is empty then set of objects $g.base$ is in positive region both in CDS, IDS reduct computation approaches.
2. If $g.base$ is in positive region and g (i.e., $(g.base \cup g.tail)$) in positive region then $g.base$ objects are considered in positive region in CDS, IDS approaches.
3. If $g.base$ is in positive region and g (i.e., $(g.base \cup g.tail)$) is not in positive region then $g.base$ objects are considered in positive region in CDS approach and not considered in IDS approach. Using VPRS semantics, $g.base$ belongs to positive region with $\beta \geq \frac{|g.base|}{|g.base \cup g.tail|}$.

Hence positive region belongingness of $g.base$ objects in IDS can be at least approximately determined by CDS approach. Using successive application of GrSR operator, tail portion gradually becoming smaller or empty resulting in β value in step 3 above becoming near to 1. Hence reduct attributes obtained using CDS approach become ϵ -approximate reduct for IDS and may require further iterations of IDS reduct computation algorithm for becoming a reduct or super reduct for IDS.

7.2.1 Transforming IDS into CDS

The first step of seed based approach for reduct computation is conversion of IDS into CDS. The conversion is done by dropping the semantics given to missing value $*$ of IDS. Assuming that $*$ is a domain value discernible from other domain values of an attribute, IDS is converted to CDS.

7.2.2 IQRA_IG for seed computation

The second step of seed based approach involves identification of effective reduct computation algorithm for CDS. From the results of Chapter 3 and Chapter 4, IQRA_IG, NPSS_IQRAIG are two effective reduct computation algorithms for moderate and large datasets respectively. In this chapter theory and experiments were dealt using IQRA_IG for seed computation in CDS and the whole approach can be applied using NPSS_IQRAIG in case of large IDS.

In order to appreciate the computational gains obtained by using IQRA_IG, two random experiments are conducted. In the first experiment incomplete datasets are generated randomly in 26 trails. In each trail, dataset is generated with 50 conditional attributes and 1 decision attribute. The size of the universe of dataset is varied from 500 to 3000 with an increase of 100 objects in each trail. 5% of total number of object values for conditional attributes ($|U| \times |C|$) are randomly selected and set as missing values. On the dataset thus generated in a trail, both IQRAIG_Incomplete and IQRA_IG algorithms are applied. The computational times incurred for both algorithms are plotted in Figure 7.2.

In the second experiment, 15 trails were conducted. In all trails, random dataset is generated with a fixed size of 3000 objects, 50 conditional attributes and 1 decision attribute. But in each trail starting with 1% to 29% with a step increase of 2%, the percentage of missing values present in dataset is randomly modified. On each dataset thus generated, IQRA_IG and IQRAIG_Incomplete algorithms are applied and computational times obtained by both algorithms are plotted in Figure 7.3. In the first experiment, range of CT for IQRA_IG is 1 to 14 seconds whereas that of IQRAIG_Incomplete is from 16 to 123 seconds. In the second experiment, range of CT for IQRA_IG is between 12 and 15 seconds whereas range of IQRAIG_Incomplete is 78 to 626 seconds. Hence IQRA_IG obtains significant computational gains over IQRAIG_Incomplete. Gains obtained by IQRA_IG will be relevant if the seed attributes selected through IQRA_IG yields near to gamma measure of '1' for IDS, i.e., ϵ -approximate reduct with ϵ value near to zero.

7.2.3 Algorithm IQRAIG_Incomplete_Seed

IQRAIG_Incomplete_Seed algorithm is IQRAIG_Incomplete algorithm with pre-processing using IQRA_IG algorithm. Algorithm IQRAIG_Incomplete_Seed is given in Algorithm 15. As discussed in previous two subsections, seed attributes are initially obtained by applying IQRA_IG algorithm on given IDS. IQRA_IG algorithm treats IDS as CDS by considering the missing value as a domain value. Starting with seed R as the selected attributes, using novel granular framework for IDS, $\gamma_R(\{d\})$ is computed. In case the end condition of $\gamma_R(\{d\}) = \gamma_C(\{d\})$

Figure 7.2: Computational Time Comparison of IQRA_IG and IQRAIG_Incomplete Algorithms for varying random dataset sizes

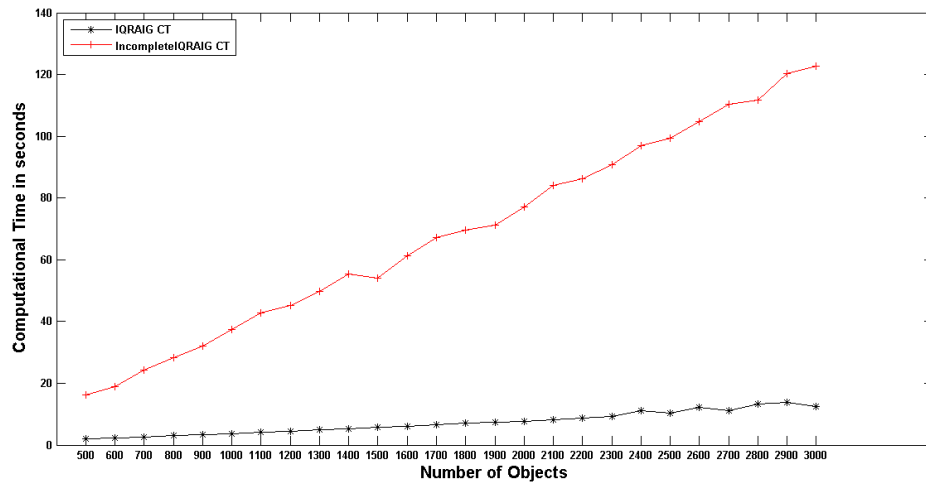
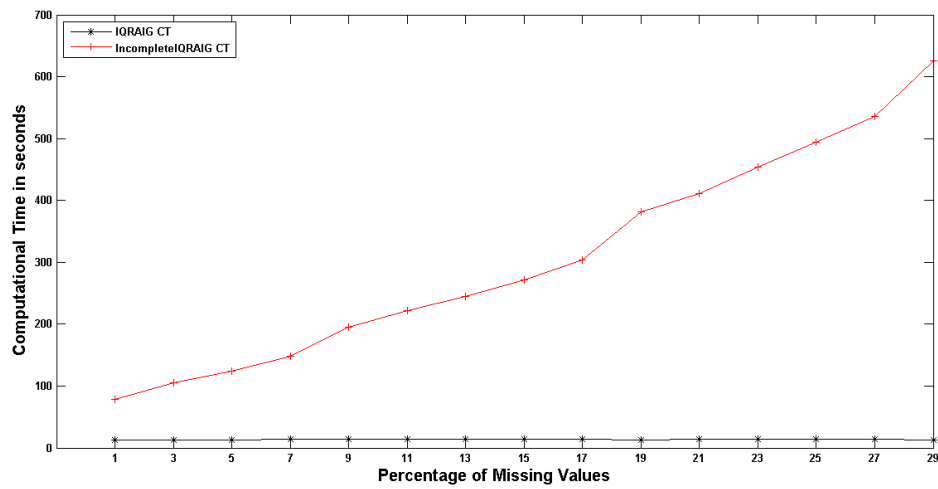


Figure 7.3: Computational Time Comparison of IQRA_IG and IQRAIG_Incomplete for varying percentages of missing values



is not reached, then iterations of IQRAIG_Incomplete are continued till the end condition is satisfied.

Theoretical complexities of IQRA_IG and IQRAIG_Incomplete are same excepting the linear overhead involved in RefineGranularSpace and CleanGranularSpace modules of IQRAIG_Incomplete. Hence theoretical time complexity of IQRAIG_Incomplete_Seed does not change but linear overheads involved in computation of initial seed attributes is not there as they were computed using IQRA_IG algorithm. Next section of Experiments and Results analyzes the nature of gains obtained by the seed based reduct computation for IDS.

Algorithm 15: IQRAIG_Incomplete_Seed

```

Input : Incomplete Decision System  $DT = (U, C \cup \{d\}, (V_a, f_a)_{a \in C \cup \{d\}})$ 
Output: Recommended Reduct:  $R$ 
1 Set  $R = IQRA\_IG(DT)$ 
2 Compute  $Gr(U, R)$ ,  $POS_R(\{d\})$  and  $\gamma_R(\{d\})$ 
3  $POSCOUNT = |POS_R(\{d\})|$  /* for denoting number of positive region objects */
4 Compute  $Gr(U, C)$ ,  $POS_C(\{d\})$  and  $\gamma_C(\{d\})$ 
5  $DT'(U', C \cup \{d'\}) = DT$  /* For differentiating reduced decision system and original decision system. */
6  $U' = U - POS_R(\{d\})$ 
7 while  $\gamma_R(\{d\}) < \gamma_C(\{d\})$  do
8    $T = R$ ,  $\gamma_T(\{d'\}) = 0$  /*  $d'$  indicates computations being done using  $DT'$  */
9   foreach  $a$  in  $C - R$  do
10      $Gr(U', R \cup \{a\}) = RefineGranularSpace(DT', Gr(U', R), a)$ 
11     Compute  $POS_{R \cup \{a\}}(\{d'\})$ ,  $\gamma_{R \cup \{a\}}(\{d'\})$ 
12     if  $\gamma_{R \cup \{a\}}(\{d'\}) > \gamma_T(\{d'\})$  then
13        $T = R \cup \{a\}$ 
14     end
15   end
16   if  $T = R$  then
17     foreach  $a$  in  $C - R$  do
18       Compute  $H(Gr(U', R \cup \{a\}))$ 
19       if  $H(Gr(U', R \cup \{a\})) > H(Gr(U', T), \{d'\})$  then
20          $T = R \cup \{a\}$ 
21       end
22     end
23      $R = T$ 
24   end
25   else
26      $R = T$ ,  $POSCOUNT = POSCOUNT + |POS_R(\{d'\})|$ 
27      $\gamma_R(\{d\}) = \frac{POSCOUNT}{|U|}$ ,
28      $U' = U - POS_R(\{d'\})$ 
29      $Gr(U', R) = cleanGranularSpace(Gr(U', R))$ 
30   end
31 end
32 Return  $R$ 

```

7.2.4 Experiments and Results

The incomplete datasets used in Chapter 5 are used for experimental analysis. For completeness, the details of datasets are again reported in Table 7.1. Primary objective of the experimentation is to perform comparative analysis of IQRAIG_Incomplete_Seed and IQRAIG_Incomplete algorithms. Similar to ex-

Table 7.1: Details of Incomplete Decision Systems

Dataset	Objects	Conditional Attributes	Decision Classes	Missing Values Count
Mushroom	8124	22	2	2480
Vote	435	16	2	392
Lung Cancer	32	56	3	5
Backup-large-test	376	35	19	1625
Shuttle	43500	9	7	6042
Adult	32561	14	2	4262
Hypothyroid	2108	25	2	3532
Audiology Standardized	200	69	24	291
Soybean-large	307	35	19	712
Dermatology	366	34	6	8
Breast Cancer Wisconsin	699	9	2	16

Table 7.2: IQRAIG_Incomplete and IQRAIG_Incomplete_Seed Results

Dataset	$\gamma_C(\{d\})$	IQRAIG_Incomplete		IQRAIG_Incomplete_Seed				
		Reduct	CT (in Seconds)	Seed	CT	$\gamma_{Seed}(\{d\})$	Seed Reduct	CT
Mushroom	1.0	5	7.76	5	0.56	1.0	5	0.68
Vote	0.326	12	5.46	10	0.122	0.317	11	1.84
Lung Cancer	1.0	5	0.48	5	0.099	1.0	5	0.11
Backup-large-test	0.907	11	7.48	11	0.288	0.729	14	1.19
Shuttle	1.0	4	86.20	4	9.122	1.0	4	20.592
Adult	0.999	8	185.76	8	16.304	0.999	8	111.08
Hypothyroid	1.0	4	7.26	2	0.212	0.871	4	4.026
Audiology Standardized	1.0	21	3.94	15	0.916	0.895	18	1.689
Soybean-large	0.928	12	7.39	13	0.294	0.56	16	2.67
Dermatology	1.0	7	2.10	7	0.388	0.98	8	0.768
Breast Cancer Wisconsin	1.0	5	0.65	4	0.072	0.994	5	0.21

periments in Chapter 5, comparative analysis is also done with results reported for IPR, IFSPA_IPR algorithms in [112].

Table 7.2 summarizes the results obtained using IQRAIG_Incomplete_Seed, IQRAIG_Incomplete algorithms on incomplete datasets. Table 7.2 contains computational time, reduct length for seed and seed reduct (CT for seed reduct includes seed CT) and also gamma measure obtained by using seed attributes as part of results for IQRAIG_Incomplete_Seed algorithm.

Tenfold experiment is conducted for analyzing classification performance of the seed, seed reduct using 3NN classifier designed for incomplete datasets as described in Chapter 5. The average and standard deviation of results obtained for reduct length, computation time, 3NN performance and stability are reported in Table 7.3. It is to be noted that exactly same folds are considered as used in tenfold experiment for IQRAIG_Incomplete algorithm in Chapter 5 and hence the results reported in Table 7.3 are comparable to results reported for IQRAIG_Incomplete algorithm in Table 5.5.

Tenfold experiments results for last four datasets given in Tables 7.3 and 5.5 for IQRAIG_Incomplete_Seed and IQRAIG_Incomplete algorithms are compared with results reported for IPR, IFSPA_IPR algorithms in [112]. Computational time comparison results are given in Table 7.4 and stability comparison is given in Table 7.5.

Table 7.3: IQRAIG_Incomplete_Seed Tenfold Experiment results on benchmark datasets

Dataset	Seed Length	Seed Reduct Length	Seed CT	Seed Reduct CT	3NN Performance			Stability
					Original	Seed	Seed Reduct	
Mushroom	5±0	5±0	0.47±0.01	0.58±0.01	87.155±2.052	100±0	100±0	0.17±0.18
Vote	10.1±0.74	12±1.25	0.105±0.01	1.75±0.41	93.953±3.669	93.02±4.65	91.63±3.33	0.09±0.10
Lung Cancer	5±0	5.1±0.32	0.093±0.01	0.11±0.02	46.667±23.307	33.33±22.22	33.33±22.22	0.49±0.38
Backup-large-test	11±0.82	13.9±0.88	0.273±0.02	2.19±0.49	58.378±7.341	82.70±6.14	83.24±5.06	0.20±0.13
Shuttle	4±0	4±0	7.61±0.05	17.39±0.11	48.96±10.255	99.52±0.10	99.52±0.10	0.04±0.17
Adult	7.8±0.42	7.8±0.42	13.65±0.12	92.55±4.19	54.917±7.709	71.08±0.93	71.08±0.93	0.07±0.10
Hypothyroid	2±0	4±0	0.19±0.01	3.66±0.09	95.429±1.274	95.52±1.33	95.57±1.21	0.08±0.17
Audiology Standardized	15.6±2.32	18.7±2.06	0.85±0.15	1.59±0.18	70.421±10.802	69.37±11.10	69.87±11.49	0.24±0.14
Soybean-large	12.4±0.84	15.2±0.92	0.29±0.03	2.27±0.55	53.333±6.849	80.33±6.93	80.67±5.62	0.26±0.12
Dermatology	6.7±0.48	7.7±0.48	0.31±0.06	0.64±0.09	75±8.282	73.88±8.09	73.06±8.59	0.24±0.25
Breast Cancer Wisconsin	4±0	4.9±0.32	0.066±	0.19±0.01	85.797±5.499	95.07±3.63	94.64±1.94	0.12±0.16

Table 7.4: CT (in Seconds) Comparison of existing [112] and proposed algorithms in tenfold experiment

Dataset	IPR	IFSPA-IPR	IQRAIG_Incomplete	IQRAIG_Incomplete_Seed
Audiology Standardized	72.2859±5.2078	17.205±1.1261	3.639±0.458	1.59±0.18
Soybean-large	34.2984±3.8632	10.8922±1.4842	6.8157±1.429	2.27±0.55
Dermatology	43.0281±1.9503	15.6281±0.5176	0.8377±0.0746	0.64±0.09
Breast-cancer Wisconsin	12.5109±1.7053	4.4656±0.8354	0.4202±0.0251	0.19±0.01

7.2.5 Analysis of Results

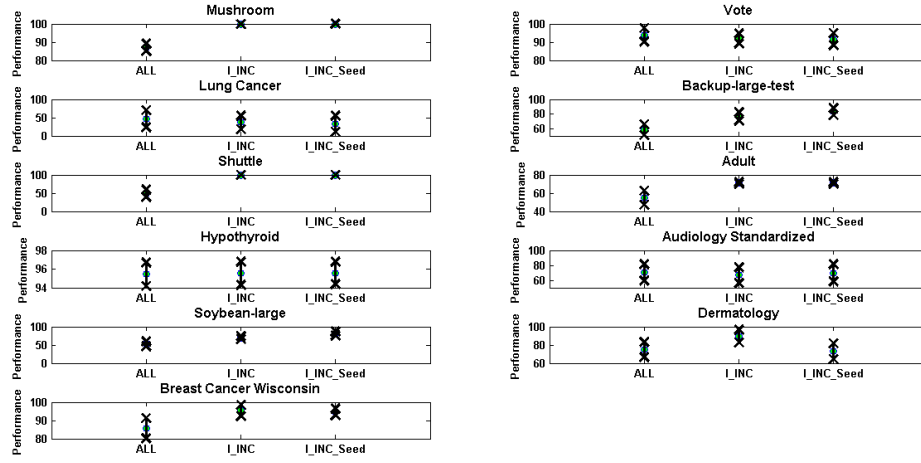
Comparative analysis results between IQRAIG_Incomplete (abbreviated as I_INC), IQRAIG_Incomplete_Seed (abbreviated as I_INC_Seed) are depicted in Figures 7.4, 7.5, 7.6, 7.7. Results of IQRAIG_Incomplete_Seed in tenfold experiment for 3NN performance evaluation (refer to Figure 7.4) have re-established the results found in Chapter 5 for IQRAIG_Incomplete. Based on unpaired t-test results, IQRAIG_Incomplete_Seed has obtained highly significant performance improvement over using all attributes in Mushroom, Backup-large-test, Shuttle, Adult, Soybean-large, Breast Cancer Wisconsin datasets. Obtaining 20 to 50% improvements in classification performance in datasets such as Shuttle, Backup-large-test establishes the advantage of using rough sets for feature selection in datasets with vagueness arising out of presence of missing values.

Both IQRAIG_Incomplete_Seed, IQRAIG_Incomplete resulted in similar results for 3NN performance evaluation, except on three datasets. In eight datasets comprising of Mushroom, Vote, Lung Cancer, Shuttle, Adult, Hypothyroid, Audiology Standardized, Breast Cancer Wisconsin both algorithms resulted in similar performance. IQRAIG_Incomplete_Seed has obtained significant, highly significant improvement over IQRAIG_Incomplete algorithm on Backup-large-test,

Table 7.5: Stability Comparison of IFSPA-IPR with IQRAIG_Incomplete in tenfold experiment

Dataset	IFSPA-IPR	IQRAIG_Incomplete	IQRAIG_Incomplete_Seed
Audiology Standardized	0.2650±0.1393	0.2443±0.1032	0.24±0.14
Soybean-large	0.2312±0.2071	0.3611±0.1859	0.26±0.12
Dermatology	0.2921±0.2293	0.1889±0.1789	0.24±0.25
Breast-cancer Wisconsin	0.0700±0.1552	0.0400±0.0843	0.12±0.16

Figure 7.4: 3NN Performance Comparison (range $\mu - \sigma$ to $\mu + \sigma$) of Tenfold Experiment



Soybean-large datasets respectively. In dermatology dataset, IQRAIG_Incomplete obtained highly significant betterment over IQRAIG_Incomplete_Seed algorithm. In summary, both IQRAIG_Incomplete_Seed, IQRAIG_Incomplete algorithms resulted in obtaining significant performance improvement in 3NN classification compared to using all attributes for classifier construction.

In reduct length comparison in tenfold experiment between IQRAIG_Incomplete, IQRAIG_Incomplete_Seed algorithms (refer to Figure 7.5), a significantly shorter reduct compared to IQRAIG_Incomplete resulted by IQRAIG_Incomplete_Seed algorithm in Audiology Standardized dataset. In Backup-large-test, Soybean-large, Dermatology datasets, IQRAIG_Incomplete obtained highly significant shorter length reducts than IQRAIG_Incomplete_Seed algorithm. In all other datasets both methods obtained similar length reducts. Overall, IQRAIG_Incomplete_Seed obtained similar length reducts and slight increase in reduct length using seed based IQRAIG_Incomplete_Seed algorithm is expected in view of seed attributes obtained without cognizing presence of missing values.

IQRAIG_Incomplete, IQRAIG_Incomplete_Seed algorithms have obtained similar stability values (refer to Figure 7.6), except in Vote dataset in which IQRAIG_Incomplete_Seed has performed significantly better.

In computational time comparison in tenfold experiment (refer to Figure 7.7), IQRAIG_Incomplete_Seed algorithm obtained highly significant reduction in computational time compared to IQRAIG_Incomplete in all datasets. These significant results are owing to computational savings in using IQRA_IG algorithm for seed attributes and also the fact that seed attributes giving significant gamma measure values in IDS. The advantage of seed based approach is due to the fact

Figure 7.5: Reduct Length Comparison (Mean $\pm$ Standard Deviation) of Tenfold Experiment

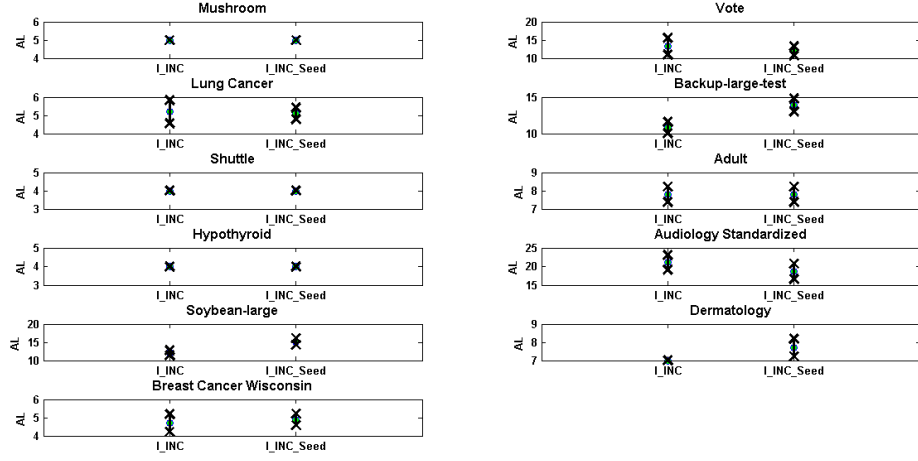


Figure 7.6: Stability Comparison (Mean $\pm$ Standard Deviation) of Tenfold Experiment

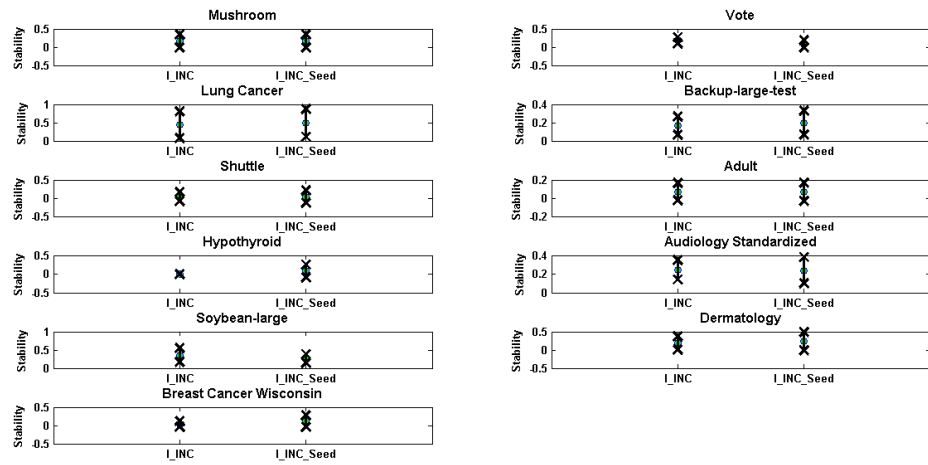
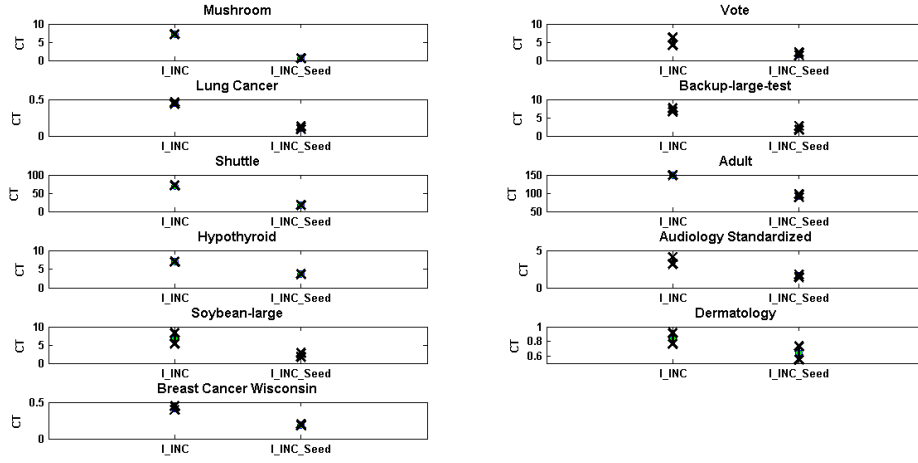


Figure 7.7: CT Comparison (range $\mu - \sigma$ to $\mu + \sigma$) of Tenfold Experiment



that important attributes having the ability to contribute to gamma measure were obtained using less complex algorithm like IQRA_IG.

In comparative analysis between IQRAIG_Incomplete_Seed and IFSPA-IPR, IPR algorithms (refer to Figures 7.8 and 7.9) and based on unpaired t-test results, it is established that IQRAIG_Incomplete_Seed obtained highly significant computational gains over IFSPA-IPR, IPR algorithms while obtaining similar stability values.

In summary, IQRAIG_Incomplete_Seed algorithm is recommended in scenarios in which computational time needs to be optimized and slight increase in reduct length is acceptable.

7.3 Seed based Reduct Computation in Hybrid Decision Systems

The computational complexity of IMQRA_MW is very high compared to that of IQRA used for symbolic decision systems. In fuzzy rough set model based positive region computation of an object, computations with all objects of the universe are involved. Whereas in classical approach, the computations are not done on object level but at a granular level where a granule is an equivalence class induced by indiscernibility relation. Traditionally, i.e. without using the fuzzy rough set model, the quantitative attributes are discretized for obtaining a symbolic decision table suitable for applying algorithm like QRA for feature selection. For instance, Rough set exploration system (RSES) [4] employs supervised discretization algo-

Figure 7.8: CT Comparison (range $\mu - \sigma$ to $\mu + \sigma$) of Tenfold Experiment with IFSPA, IPR algorithms

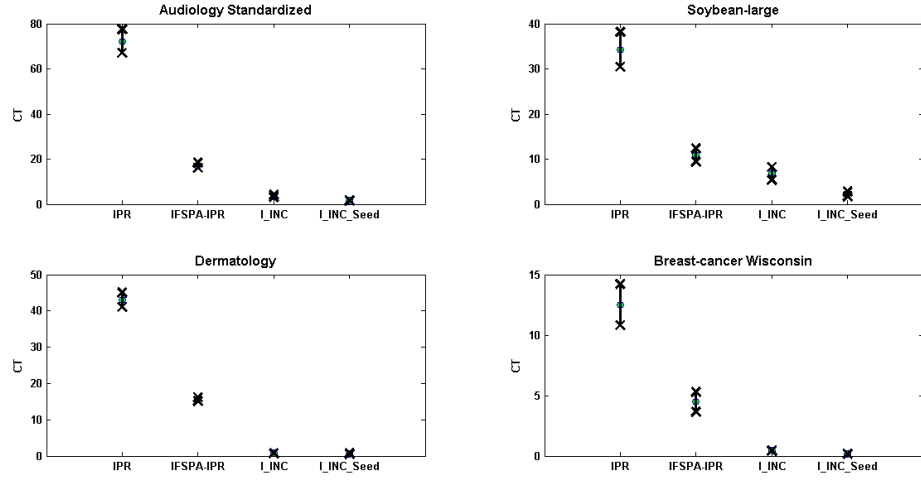
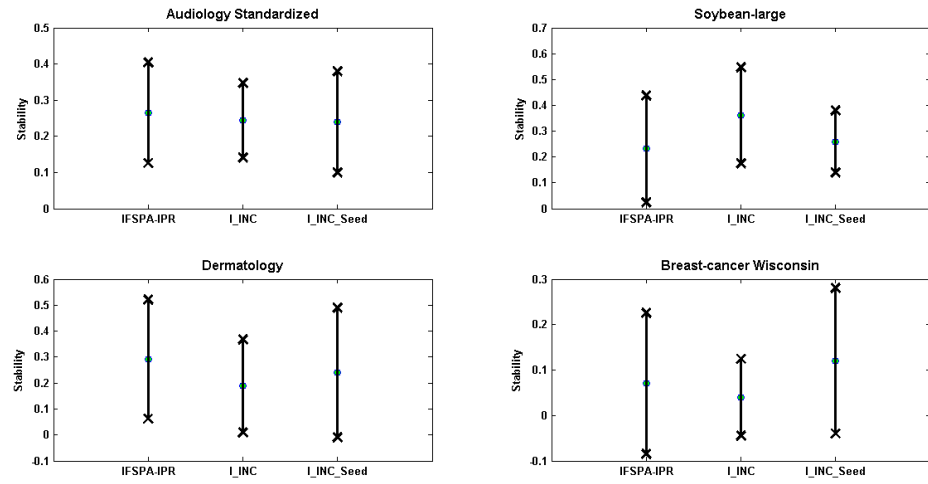


Figure 7.9: Stability Comparison of Tenfold Experiment (range $\mu - \sigma$ to $\mu + \sigma$) with IFSPA, IPR algorithms



rithm CUTS proposed by Pawlak for discretization of quantitative attributes. But the process of discretization leads to loss of information present in the original decision system but reduct algorithms on symbolic decision systems are computationally efficient. To avail the benefits of both fuzzy rough set model approach and classical rough set model approach, a hybrid approach named as algorithm IMQRA using Seed (IMQRA_Seed) is proposed.

The new approach uses a suitable unsupervised discretization method relevant for the given fuzzy similarity relation for obtaining symbolic decision system from hybrid decision system. IQRA_IG algorithm is applied on the obtained symbolic decision system to derive a starting set of features called as Seed Reduct. The Seed Reduct B induces gamma measure $M(B)$ and hence becomes fuzzy M-decision super reduct to degree $M(B)$ (FDSR- $M(B)$) [20]. If the obtained gamma measure is below the specified degree, further iterations of IMQRA_MW are used till degree limit is satisfied. This results in fuzzy M-decision reduct to specified degree. In the case that the degree is '1' the resulting reduct is called as fuzzy decision super reduct (FDSR) otherwise called as approximate FDSR. Hence the resulting output of this approach is a fuzzy decision super reduct utilizing all the information present in the hybrid decision system but selection of the seed reduct is based on symbolic decision system giving significant computational gains. Initially before conducting any experiments, the authors predicted that seed reduct may obtain gamma measure around 0.5 so that remaining classification accuracy can be obtained by selecting further features using IMQRA_Seed. But the results of the experiments indicate on an average a gamma measure of 0.95 for the seed reduct in all the experiments that were conducted on bench mark hybrid decision systems.

7.3.1 Proposed Approach for Seed based Reduct Computation in Hybrid Decision Systems

The time complexity of IQRA like algorithms for symbolic decision systems is much lesser than that of IMQRA_MW applied for hybrid decision systems. The approach of fuzzy rough sets enable to find reduct for hybrid decision systems without discretizing quantitative attributes. Discretization of quantitative attributes result in losing information present the hybrid decision table. But IMQRA_MW algorithm owing to its time complexity and also space requirements (usually similarity matrices representing fuzzy tolerance relation for all conditional attributes need to be maintained) is not suitable for large hybrid decision systems. Especially if feature selection needs to be done in an online environment, IMQRA_MW is not suitable. In such situations, to arrive at a FDSR in a less computational time the following approach is developed.

Even though discretization of HDT results in loss of information, selecting a suitable discretization method which has similarity with the fuzzy tolerance relation helps in an approximate way for mimicking the feature selection process involved in IMQRA_MW. Reduct attributes computed on discretized HDT using computationally efficient QRA like algorithm, will be relevant attributes for HDT also because of the similarity of discretization function with the fuzzy tolerance relation. The selected attributes B called as Seed Reduct, usually become approximate FDSR to HDT to degree $M(B)$. The reason for resulting in approximate reduct is because in discretized HDT obtaining a positive region membership of '1' for an object is less constrained than in HDT.

The proposed approach contains three steps. First is selection of relevant discretization method which has similarity with given fuzzy tolerance relation. Second step is to derive Seed Reduct by applying efficient reduct computation algorithm for symbolic decision systems on discretized HDT, discretized using the selected discretization method. Finally, using IMQRA starting with Seed Reduct as B evolving into required $FDSR - \alpha$. Section (7.3.2) describes the discretization method selected for a fuzzy tolerance relation considered in the study. Section (7.3.3) contains an empirical comparison of IQRA_IG algorithm with IMQRA_MW algorithm to further establish the relevance of choosing IQRA_IG for seed reduct computation. Section (7.3.4) contains the proposed IMQRA_Seed algorithm along with complexity analysis.

7.3.2 Selection of Relevant Discretization Method

The fuzzy tolerance relation R is defined for each attribute $a_n \in C^n$. The fuzzy tolerance relation in Chapter 6 (as given in Eq. (6.22)) is used for demonstration of proof of concept and for completeness is given below.

For $a_n \in C^n$ fuzzy tolerance relation R_a is defined for all $x, y \in U$ as

$$R_{a_n}(x, y) = \max \left(\min \left(\frac{a_n(x) - a_n(y) + \sigma_{a_n}}{\sigma_{a_n}}, \frac{a_n(y) - a_n(x) + \sigma_{a_n}}{\sigma_{a_n}} \right), 0 \right) \forall x, y \in U \quad (7.1)$$

In case of qualitative attributes, classical indiscernibility relation is used, i.e., for all $a_c \in C^c \cup \{d\}$, R_a is defined as,

$$R_{a_c}(x, y) = \begin{cases} 1 & \text{if } a_c(x) = a_c(y) \\ 0 & \text{Otherwise} \end{cases} \quad \forall x, y \in U \quad (7.2)$$

The semantics of the fuzzy tolerance relation given in equation (7.1) is that if two

object values are equal then the fuzzy relation membership of the objects is equal to ‘1’. If two object values are within σ_{a_n} the membership value is $0 < R_{a_n} < 1$. If two object values are within $0.5\sigma_{a_n}$ then membership value is more than 0.5. If the absolute difference between two object values is greater than or equal to σ_{a_n} then membership value is equal to zero. Hence considering a membership value of above 0.5 as significant nearness between objects based on attribute a and having absolute difference of $0.5\sigma_{a_n}$ as permissible the following discretization strategy is adopted. For each $a_n \in C^n$ let μ_{a_n}, σ_{a_n} denote the mean and standard deviation of object values of attribute a_n in HDT. The discretization function used is, for object $x \in U$,

$$discretevalue(a_n(x)) = floor\left(\frac{(a_n(x) - \mu_{a_n})}{0.5 * \sigma_{a_n}}\right) \quad (7.3)$$

The objects which are placed in the same bin using (7.3) are having object values with respect to a_n within $0.5\sigma_{a_n}$ and hence related by indiscernibility relation membership of ‘1’ using (7.2). Hence some of the pair of objects which get membership value more than 0.5 using (3) in HDT are assigned membership of ‘1’ in discretized HDT.

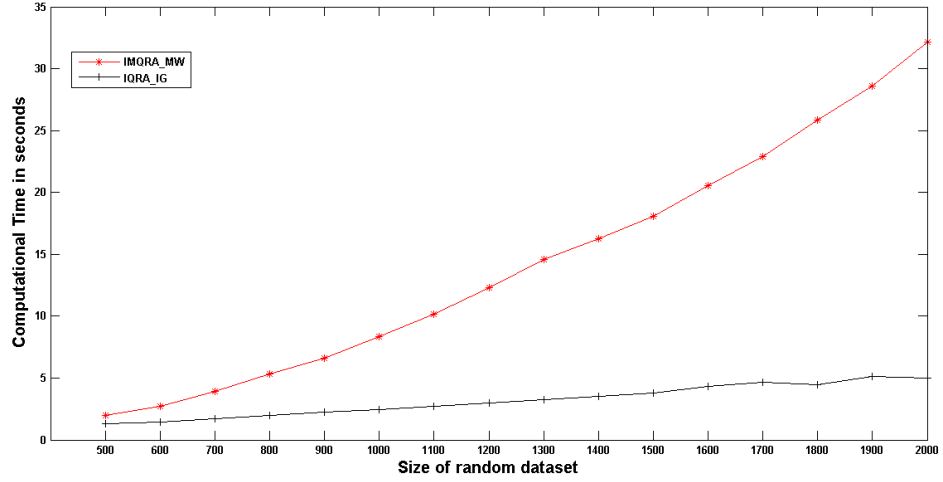
Having obtained a discretization approach which captures analogy with fuzzy tolerance relation, next sub section illustrates the scalability of IQRA_IG as a reduct finding algorithm for symbolic decision system discretized HDT in comparison to IMQRA_MW algorithm.

7.3.3 Empirical comparison of IQRA_IG and IMQRA_MW algorithms

Theoretically the complexity of IQRA_IG is very less compared to IMQRA_MW. The following experiment demonstrates empirically the computational gain in IQRA_IG over IMQRA_MW algorithm. Hybrid data sets (containing 35 quantitative attributes) are generated randomly with varying sizes from 500 to 2000 objects by increasing by 100 objects in each trail. In each trail the HDT is discretized using the discretized function given in Eq. (7.3) to obtain discretizedHDT. Fig. 7.10 depicts the computational time involved for IQRA_IG on discretized-HDT and IMQRA_MW on HDT for all trails in the increasing order of data set size.

In Fig. 7.10, as size of the data set size varies from 500 to 2000 objects the computational time of IQRA_IG varied from 1 to 4 seconds. But that of IMQRA_MW varied from 1 to 32 seconds. The experiment demonstrate that IQRA_IG can obtain significant computational gains compared to IMQRA_MW.

Figure 7.10: CT Comparison of IQRA_IG and IMQRA_MW Algorithms for varying random hybrid dataset sizes



Hence IQRA_IG is chosen as the reduct algorithm for applying on discretizedHDT to obtain Seed Reduct.

7.3.4 Seed based Fuzzy Decision Reduct

IMQRA_Seed algorithm is a modification to IMQRA_MW algorithm for obtaining FDSR using seed (FDSR_Seed). The discretized decision system HDT_Symbolic is constructed by discretizing the quantitative attributes in HDT using a discretization function suitable to fuzzy tolerance relation used in the Radzikowska Kerry's model in IMQRA_MW. Applying IQRA_IG on HDT_Symbolic, Seed Reduct B0 is obtained. If gamma measure obtained by Seed Reduct, $M(\text{Seed Reduct})$, is less than the specified degree α then starting with Seed Reduct as the already included attributes into recommended reduct B , IMQRA_MW iterations are continued till $FDSR - \alpha$ is obtained. Algorithm IMQRA_Seed is given in the next sub section.

7.3.4.1 Algorithm IMQRA_Seed

Algorithm 16: IMQRA_Seed	
Input	: $HDT = (U^s, C^h = C^n \cup C^c, \{d\}, \{V_a, f_a\}_{a \in C^h \cup \{d\}})$: Hybrid Decision Table
α :	Degree
R :	Fuzzy similarity relation
F :	Discretization Function
N :	Fuzzy Negation
Γ :	t-norm for computing R_B for any $B \subseteq C^h$ using equation (6.9)
Output:	B : FDSR_Seed
1	Discretize quantitative attributes C^n using F to obtain $HDT_Symbolic$.
2	$B0 = IQRA_IG(HDT_Symbolic) /* Seed Reduct Computation$ */
3	$B = B0$
4	Compute $\gamma(B)$
5	Compute $ABSOLUTE_POS_B, \gamma_{ABSOLUTE}$.
6	Construct HDT' by removal of $ABSOLUTE_POS_B$ objects
7	while $\gamma_B < \alpha$ do
8	$T = B$,
9	$best = -1$
10	foreach $a \in C^h - B$ do
11	if $\gamma'_{B \cup \{a\}} > best$ then
12	$T = B \cup \{a\}$
13	$best = \gamma'_{B \cup \{a\}}$
14	end
15	end
16	$B = T$
17	$\gamma_B = \gamma_{ABSOLUTE} + \gamma'_B$
18	Update $ABSOLUTE_POS_B, \gamma_{ABSOLUTE}$.
19	Construct HDT' by removal of $ABSOLUTE_POS_B$ objects
20	end
21	Return B .

Let n be the size of the obtained FDSR_Seed B . Let k denote the size of Seed Reduct $B0$. Then the number of IMQRA_MW iterations involved in IMQRA_Seed algorithm is $n - k$. The relevancy of discretization function used is determined by obtained $M(B0)$. In case the chosen discretization function is approximating well the fuzzy tolerance relation then obtained $M(B0)$ will be significant (near to 1) and $n - k$ becomes small indicating less iterations of IMQRA_MW are involved in IMQRA_Seed. In this case it is also expected that $M(B0)$ will be nearly equal to gamma measure M value obtained by first k attributes of FDSR (obtained by direct application of IMQRA_MW) and the sizes of FDSR_Seed and FDSR will be similar. In contrast if the discretization function is not related to fuzzy tolerance relation then $M(B0)$ is insignificant (near to zero) and size of FDSR_Seed will be higher than FDSR.

The total time complexity of IMQRA_Seed algorithm expressed as sum of iteration wise time complexity is

$$\sum_{i=1}^k O((|C^h| - i + 1) |U_i| \log |U_i|) + \sum_{I=k+1}^n O((|C^h| - i + 1) |U_I|^2)$$

Table 7.6: Details of Hybrid Decision Systems

Dataset	Objects	Attributes	Numeric Attributes	Symbolic Attributes	Decision Classes
Image Segmentation	2310	19	16	3	7
Sonar Mines vs Rocks	208	60	60	0	2
WDBC	569	30	30	0	2
Cleveland	297	13	5	8	5
Glass	214	9	9	0	6
Ionosphere	351	34	33	1	2
Web	149	2556	2556	0	5
Wine	178	13	13	0	3
Heart	270	13	5	8	2
Olitos	120	25	25	0	4
Spambase	4601	57	57	0	2

Universe of objects used in each iteration of IQRA_IG algorithm, because of positive region removal, are denoted by U_i ($1 \leq i \leq k$) and used in iterations of IMQRA_MW, because of absolute positive region removal, are denoted by U_I ($k+1 \leq I \leq n$). Computational gain obtained by IMQRA_Seed over IMQRA_MW depends on number of high complexity iterations of IMQRA_MW are replaced by less complexity iterations of IQRA_IG. As the iteration complexity of IQRA_IG is dominated by that of IMQRA_MW, asymptotically for large hybrid decision systems the time complexity of IMQRA_Seed becomes $O\left(\left(|C^h| - k\right)^2 |U|^2\right)$ yielding significant computational gains over IMQRA_MW algorithm.

7.3.5 Experiments and Results

In all experiments, Lukasiewicz t-norm [20] is used for extension of fuzzy tolerance relation for attribute set as in (6.9). The degree parameter α is taken as ‘1’. Hybrid datasets used in Chapter 6 are used for experiments. For completeness, details of datasets are again reported in Table 7.6 .

In the first experiment, algorithms IMQRA_MW, IMQRA_Seed are applied on all datasets. The results are summarized in Table 7.7. Column ‘M(SR)’ gives the M-measure obtained using seed reduct (SR) attributes. Reduct sizes and computational time (in seconds) obtained are given for seed reduct (SR) using IQRA_IG algorithm as part of IMQRA_Seed algorithm, FDSR_Seed (FS) using IMQRA_Seed algorithm, FDSR(F) using IMQRA_MW algorithm. It is to be noted that Seed Reduct computation time is included in computation time of FDSR_Seed. For example in Image Segmentation data set the CT required for evolving into FDSR_Seed from Seed Reduct is 2.03 seconds (CT of FDSR_Seed-CT of Seed Reduct).

In second experiment ten times repeated tenfold cross validation is performed for assessing the average behavior of reduct computation aspects and classification accuracy of classifier models constructed with all attributes, Seed Reduct,

Table 7.7: Results of Reduct computation (M(Seed Reduct), Size of reduct and computational time) using entire data sets for Seed Reduct (SR), FDSR_Seed (FS), FDSR (F)

Dataset	M(SR)	Reduct Size			CT (in Seconds)		
		SR	FS	F	SR	FS	F
Image Segmentation	0.996	9	16	16	0.56	2.59	17.4
Sonar Mines vs Rocks	0.980	5	6	5	0.55	0.62	0.69
WDBC	0.996	5	6	6	0.29	0.39	1.3469
Cleveland	0.983	6	8	8	0.18	0.21	0.5
Glass	0.983	8	9	9	0.07	0.09	0.28
Ionosphere	0.978	5	7	7	0.28	0.34	0.84
Web	0.961	19	23	20	18.07	23.76	62.17
Wine	0.981	4	5	5	0.05	0.07	0.23
Heart	0.961	5	8	7	0.16	0.18	0.31
Olitos	0.96	4	5	5	0.16	0.19	0.25
Spambase	0.9954	35	45	45	9.74	38.04	* ¹

FDSR_Seed, FDSR. In a tenfold cross validation experiment the data set is randomly divided into ten equal size parts. In each fold, one part is used for test data and remaining 9 parts are used as training data. On training data IMQRA_Seed, IMQRA algorithms are applied to derive Seed Reduct, FDSR_Seed and FDSR. Projected training and test data sets are constructed using these reducts. Classifier models are constructed on training data, and projected training data sets producing four classifier models. The accuracy of classifier modes is evaluated by testing with corresponding test data sets. Three classifiers are used for constructing classifier models, namely 3NN, Multilayer Perceptron Neural Network (MLP), Classification and Regression Tree (CART). Table 7.8 contains the summary of average results obtained in 10xRepeated tenfold cross validation experiment pertaining to reduct computation aspects. The average result of classification accuracies for three classifiers on four classifier models along with 95% confidence intervals is reported in Table 7.9.

7.3.5.1 Results with Spambase dataset

The results for Spambase dataset are reported as an example for a large hybrid decision system. A limitation of applying IMQRA_MW algorithm directly on Spambase dataset or such large hybrid decision systems arises due to space complexity. For example, for Spambase dataset, 57 similarity matrices needs to be maintained of size 4601×4601 along with matrices required for intermediate computations. Such large space requirements were not met by the environment in which experiments are performed. Because of this limitation, the results reported in Tables

Table 7.8: Average Result of Reduct computation (M(Seed Reduct), Size of reduct and computational time) in 10xrepeated tenfold cross validation for Seed Reduct (SR), FDSR_Seed (FS), FDSR(F)

Dataset	M(SR) (Mean±Std)	Reduct Size (Mean±Std)			CT (in Seconds)(Mean±Std)		
		SR	FS	F	SR	FS	F
Image Segmentation	0.994 ±0.003	8.56 ±0.57	16.11 ±0.45	16.16±0.55	0.51 ±0.04	2.12±0.09	13.96±0.09
Sonar Mines vs Rocks	0.95 ±0.021	4.18 ±0.39	5.75 ±0.44	5.26 ±0.44	0.5 ±0.02	0.55±0.03	0.40±0.01
WDBC	0.991 ±0.003	4.99 ±0.27	6.33 ±0.47	5.84 ±0.37	0.25 ±0.03	0.33±0.03	1.00±0.02
Cleveland	0.984 ±0.008	6.06 ±0.31	7.82 ±0.54	7.66 ±0.48	0.18 ±0.01	0.21±0.01	0.22±0.00
Glass	0.981 ±0.004	7.9 ±0.39	8.98 ±0.14	9 ±0.00	0.07 ±0.00	0.08±0.00	0.10±0.00
Ionosphere	0.975 ±0.016	5.11 ±0.45	6.92 ±0.46	7.22 ±0.56	0.25 ±0.02	0.29±0.02	0.56±0.02
Web	0.948 ±0.012	17.46 ±1.32	20.99 ±1.36	19.01±1.24	15.29 ±1.61	21.41±2.66	51.51±3.25
Wine	0.982 ±0.018	3.97±0.17	5 ±0.14	5 ±0.00	0.05 ±0.00	0.06±0.00	0.06±0.00
Heart	0.975 ±0.012	5.14 ±0.40	7.5 ±0.52	7.07 ±0.26	0.13 ±0.01	0.15±0.01	0.12±0.00
Olitos	0.957 ±0.013	4 ±0.00	5.15 ±0.36	5 ±0.00	0.13 ±0.01	0.15±0.01	0.12±0.00
Spambase	0.9932±0.0021	31.45±2.44	44.3±0.87	*	9.07±0.7	29.76±1.62	*

contain “*” corresponding to IMQRA_MW results for Spambase dataset. The results obtained by NPSS_IMQRA in Chapter 6, IMQRA_Seed in this chapter illustrate the advantage of such methodologies in dealing with large hybrid decision systems where in direct methodologies like IMQRA_MW fail to give reduct or give reduct after lot of computational delay.

An attempt towards reducing the space complexity of IMQRA_MW impacted the time complexity significantly. For example, in order to overcome space complexity issue of IMQRA_MW, IMQRA_MW was implemented using memory conserving operations. In this variant of IMQRA_MW, similarity matrices are not stored but computed whenever requirement arises. This variant of IMQRA_MW resulted in obtaining reduct for Spambase dataset in 457.23 seconds. As reported in Table 7.7, IMQRA_Seed algorithm obtained the same reduct in 38.04 seconds in which only 9.74 seconds were taken for arriving at seed attributes. The seed obtained is 24, 26, 11, 6, 20, 2, 18, 49, 51, 44, 45, 1, 4, 32, 15, 50, 17, 41, 48, 43, 22, 0, 9, 21, 13, 23, 36, 12, 34, 7, 25, 47, 54, 19, and 52. Comparing seed attributes with first 35 attributes of FDSR given in [20] it is found that 32 attributes are present in obtained Seed Reduct. The gamma measure of Seed Reduct is 0.9954 and is similar to gamma measure obtained by first 35 attributes of FDSR which was 0.997 [20]. The FDSR_Seed evolved over Seed using IMQRA_Seed algorithm resulted in as a 45 size reduct. The additional attributes included into Seed Reduct are 56, 55, 5, 53, 38, 8, 40, 30, 35, and 27. The obtained FDSR_Seed is exactly same as FDSR.

7.3.6 Analysis of Results

From the results reported for entire data sets in Table 7.7, and for 10xRepeated tenfold cross validation in Table 7.8 the significant result is that seed has obtained significant (near to 1) gamma measure value of above 0.95 on all data sets. The

Table 7.9: The accuracy results (Mean accuracy, 95% Confidence interval) for 3nn, MLP, cart classifiers in 10x repeated 10 fold cross validation using attributes (all), Seed Reduct (SR), FDSR_Seed (FS), FDSR (F)

Dataset	Attribute Set Type	3NN		MLP		CART	
		Mean(%)	95% CI	Mean(%)	95% CI	Mean(%)	95% CI
Image Segmentation	ALL	95.65	(95.40 , 95.89)	96.47	(96.25 , 96.69)	95.45	(95.15 , 95.76)
	S	93.40	(92.94 , 93.87)	96.35	(96.08 , 96.63)	95.92	(95.61 , 96.23)
	FS	95.65	(95.40 , 95.89)	96.81	(96.58 , 97.05)	95.62	(95.32 , 95.92)
	F	95.65	(95.40 , 95.89)	96.69	(96.47 , 96.91)	95.62	(95.31 , 95.92)
Sonar Mines vs Rocks	ALL	82.05	(80.36 , 83.74)	78.20	(76.43 , 79.97)	71.40	(69.45 , 73.35)
	S	72.40	(70.65 , 74.15)	69.80	(67.67 , 71.93)	67.85	(65.84 , 69.86)
	FS	75.05	(73.27 , 76.83)	69.10	(67.10 , 71.10)	69.05	(67.14 , 70.96)
	F	74.40	(72.70 , 76.10)	72.95	(70.90 , 75.00)	70.45	(68.49 , 72.41)
WDBC	ALL	92.73	(92.01 , 93.45)	93.73	(92.94 , 94.52)	92.80	(92.11 , 93.50)
	S	90.95	(90.16 , 91.73)	95.63	(95.03 , 96.22)	93.27	(92.52 , 94.01)
	FS	90.98	(90.22 , 91.75)	96.05	(95.51 , 96.60)	93.16	(92.44 , 93.88)
	F	93.75	(93.15 , 94.35)	96.77	(96.27 , 97.26)	93.70	(93.07 , 94.32)
Cleveland	ALL	44.59	(42.80 , 46.37)	51.28	(49.37 , 53.19)	49.14	(47.65 , 50.63)
	S	44.03	(42.26 , 45.81)	52.48	(50.50 , 54.46)	48.41	(46.57 , 50.26)
	FS	44.38	(42.59 , 46.17)	53.03	(51.19 , 54.88)	49.90	(48.36 , 51.43)
	F	44.38	(42.60 , 46.16)	54.38	(52.78 , 55.98)	49.90	(48.33 , 51.46)
Glass	ALL	68.90	(66.80 , 71.01)	63.43	(61.34 , 65.52)	67.71	(65.90 , 69.53)
	S	68.57	(66.46 , 70.69)	61.67	(59.45 , 63.88)	67.33	(65.58 , 69.09)
	FS	68.95	(66.84 , 71.07)	62.24	(60.30 , 64.18)	67.62	(65.81 , 69.42)
	F	68.90	(66.80 , 71.01)	63.00	(61.24 , 64.76)	67.62	(65.82 , 69.42)
Ionosphere	ALL	85.17	(83.97 , 86.37)	82.8	(81.44 , 84.16)	88.43	(87.43 , 89.42)
	S	88.69	(87.69 , 89.68)	87.77	(86.6 , 88.94)	87.23	(86.12 , 88.34)
	FS	88.31	(87.17 , 89.46)	88.37	(87.28 , 89.47)	88.4	(87.32 , 89.48)
	F	88.63	(87.48 , 89.78)	87.46	(86.35 , 88.56)	88.6	(87.59 , 89.61)
Wine	ALL	72	(69.73 , 74.27)	94.47	(93.42 , 95.52)	89.71	(88.49 , 90.93)
	S	85.88	(83.55 , 88.21)	91.29	(89.98 , 92.61)	90.88	(89.50 , 92.27)
	FS	85.53	(83.06 , 88.00)	92.76	(91.45 , 94.08)	91.53	(90.22 , 92.84)
	F	73.41	(71.15 , 75.67)	94.06	(93.00 , 95.11)	91.53	(90.12 , 92.94)
Heart	ALL	64.39	(62.64 , 66.13)	72.7	(70.92 , 74.49)	75.51	(73.95 , 77.06)
	S	63.76	(61.93 , 65.58)	68.15	(66.56 , 69.73)	60.34	(58.69 , 62.00)
	FS	64.24	(62.49 , 65.99)	72.67	(70.85 , 74.49)	72.97	(71.40 , 74.55)
	F	64.24	(62.49 , 65.99)	76.03	(74.36 , 77.71)	73.32	(71.74 , 74.91)
Olitos	ALL	62.87	(60.45 , 65.29)	73.02	(70.35 , 75.70)	63.39	(60.74 , 66.05)
	S	54.02	(50.97 , 57.07)	61.48	(58.23 , 64.73)	58.89	(55.9 , 61.87)
	FS	56.28	(53.37 , 59.19)	63.06	(60.10 , 66.02)	60.68	(57.92 , 63.44)
	F	52.57	(49.65 , 55.49)	64.17	(61.32 , 67.03)	61.94	(59.34 , 64.53)
Web	ALL	48.86	(45.91 , 51.81)	38.00	(35.53 , 40.47)	53.71	(51.69 , 55.74)
	S	45.57	(42.96 , 48.18)	43.43	(40.86 , 46.00)	47.43	(45.04 , 49.81)
	FS	48.5	(45.86 , 51.14)	41.79	(39.00 , 44.57)	48.86	(46.02 , 51.70)
	F	50.14	(47.36 , 52.93)	39.86	(36.53 , 43.18)	49.21	(46.38 , 52.05)
Spambase	ALL	81.13	(80.79 , 81.48)	92.92	(92.68 , 93.16)	91.83	(91.54 , 92.13)
	S	88.81	(88.05 , 89.57)	92.76	(92.53 , 92.99)	91.60	(91.34 , 91.87)
	FS	81.10	(80.77 , 81.45)	93.04	(92.81 , 93.29)	91.83	(91.57 , 92.08)
	F	*	*	*	*	*	*

sizes of FDSR_Seed and FDSR are also similar for all data sets in both experiments. Both these outcomes establish that the chosen discretization approach is relevant to fuzzy tolerance relation given in Eq. 7.1. Computation time results show that on small size data sets the CT's of IMQRA_Seed and IMQRA_MW are not differentiable. But for large data sets such as Image Segmentation, Web data sets, IMQRA_Seed has obtained significant computational gains (85.1%, 62.1% respectively) over IMQRA_MW algorithm.

From the classifier accuracy results reported in Table 7.9 , seed based classifier models resulted in similar classification accuracies with other models in most of the data sets which further establish the significance of proposed approach. Between classifier models built using FDSR_Seed and FDSR accuracies over 10x10 fold experiments are compared using right tailed t-test at 0.05 significance level. The t-test results and the reported 95% confidence intervals established the following. In 3NN classifier models, FDSR models have significantly achieved better performance than FDSR_Seed models in WDBC data set and FDSR_Seed models have achieved significant better performance in Wine and Olitos data sets. In MLP Classifier models, FDSR model has performed better in Sonar Mines vs Rocks, Wine data sets. In all other instances there is no significant difference between accuracies obtained in FDSR_Seed and FDSR models. Particularly in CART classifier both models have similar results with all data sets. Overall one can conclude that FDSR_Seed based classifier models have obtained similar classification accuracies as FDSR based models.

Experiment done with Spambase data set illustrates the significance of proposed approach for finding FDSR_Seed for large hybrid decision systems. In particular obtaining gamma measure of 0.9954 for Seed Reduct in around 10 seconds compared to 7 minutes involved in computation with memory optimized IMQRA_MW algorithm exemplifies the utility of the proposed approach. The gamma measure of seed is very near to gamma measure of first 35 attributes of FDSR. This property has implication of resulting FDSR_Seed of similar size (in this case same size) with FDSR.

IMQRA_Seed is achieving more computational gains compared to IMQRA_MW with increasing size of data set as seen in the results obtained with Image Segmentation, Web, Spambase data sets. This scalability aspect of IMQRA_Seed makes it as a viable alternative to find FDSR for large hybrid decision systems.

7.4 Summary

In this chapter seed based approaches are developed with effective hybridization of reduct computation, essentially extended rough set tool(s) wrapped around

Figure 7.11: 95% Confidence Interval for 3NN Classification Experiment Results

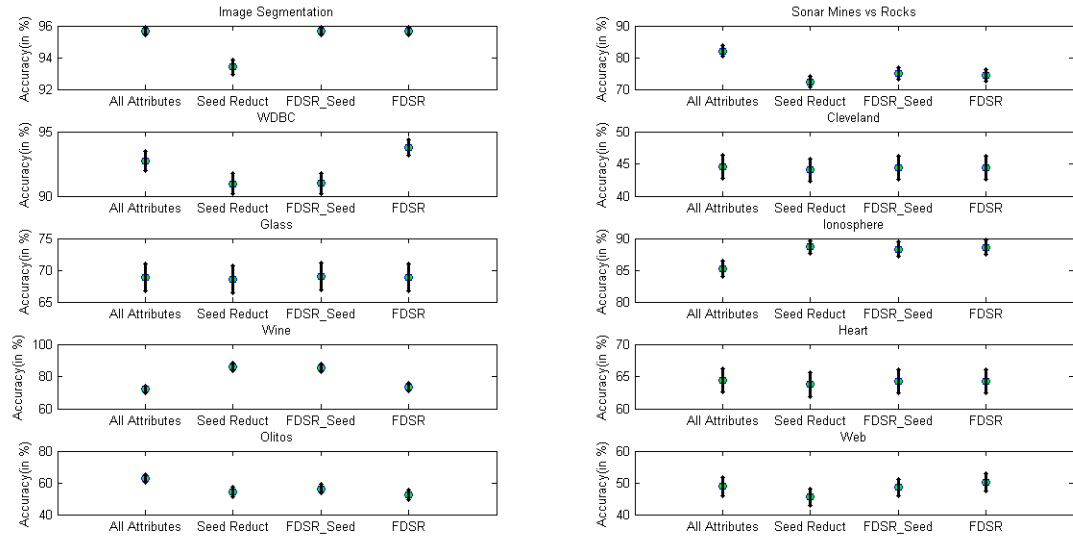


Figure 7.12: 95% Confidence Interval for MLP Classification Experiment Results

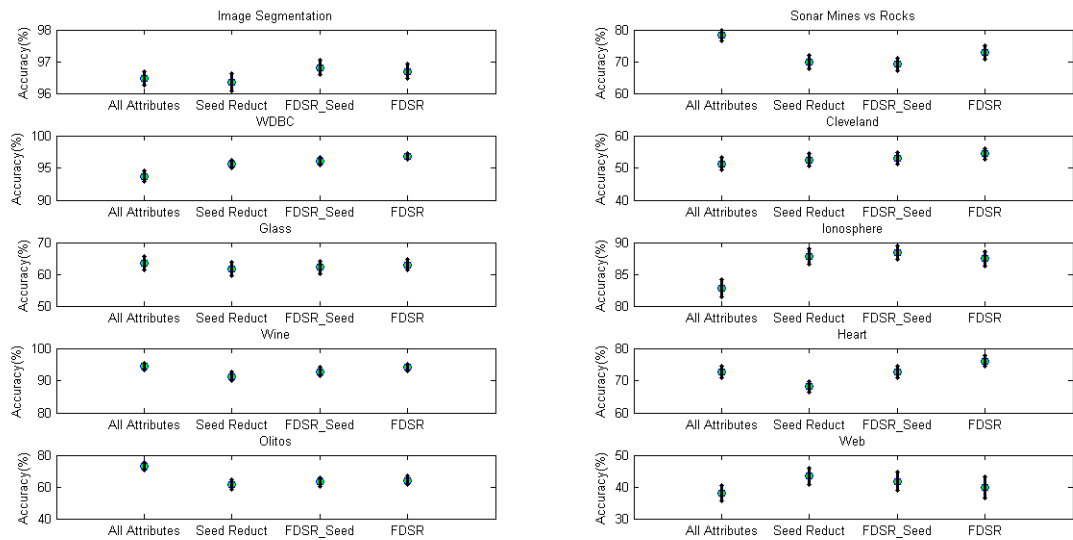
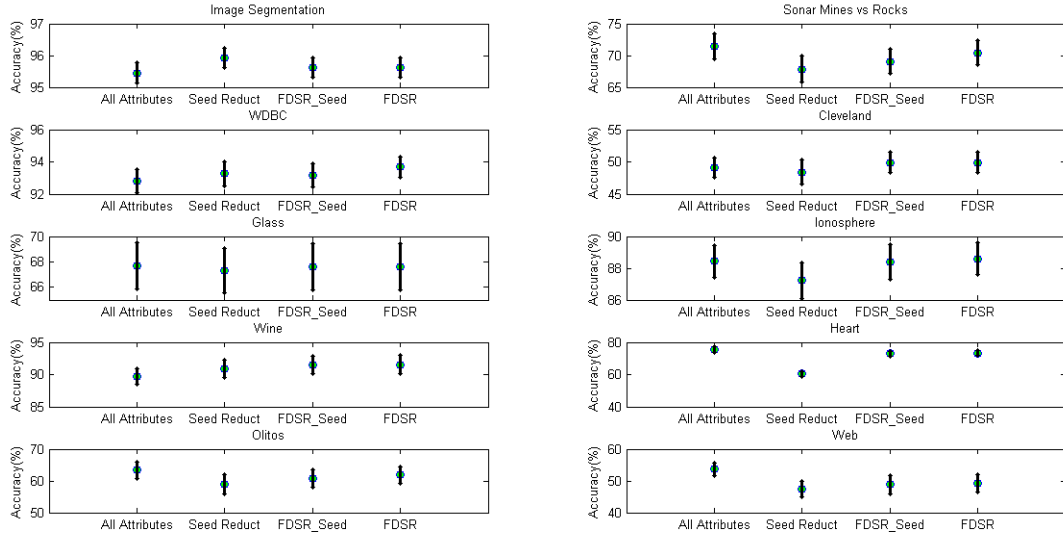


Figure 7.13: 95% Confidence Interval for CART Classification Experiment Results



classical rough set tool, for higher complex decision system with that of simpler decision systems to achieve significant computational gains without compromising on quality of reduct. Seed based approaches are developed for reduct computation in incomplete decision systems, hybrid decision systems.

IQRAIG_Incomplete_Seed, IMQRA_Seed algorithms are developed for incomplete decision systems and hybrid decision systems respectively as seed based approaches. In these approaches, incomplete or hybrid decision system is reduced to complete symbolic decision system and reduct obtained with reduced system is taken as the seed for evolving to reduct in respective decision systems. These approaches are found to be effective and provided an alternate approach for scalable reduct computation.

Chapter 8

Conclusion

This chapter contains the summary of contributions made in the thesis in Section 8.1 and future scope of work is presented in Section 8.2.

8.1 Summary of Contributions

The study developed effective and scalable reduct computation methods for decision systems for different kinds of data (1) Complete Symbolic, (2) Incomplete Symbolic and (3) Hybrid, considering complete table i.e., all objects, and sampled table i.e., sampled objects. The proposed methods are evaluated by performing comparative analysis with existing approaches by considering various database repositories. The algorithms developed in the study, exhibit computational efficiency without compromising accuracy compared to the existing methods. Results reported in Chapters 3 to 7 for proposed algorithms are quite encouraging comparative to what was reported for existing approaches in literature. It is expected that gain factor of ERCS is at least 50%.

The importance of dealing with trivial ambiguous situations, the suitability of non persistent sampling approaches for reduct computation, novel granular framework for incomplete decision systems, simplified computation model for Radzikowska-Kerry's fuzzy rough set model and the relevance of seed based hybrid approaches are the prominent contributions of this thesis whose relevance goes beyond the context of proposed algorithms.

8.2 Future Scope of Work

The secondary heuristics considered in work are VPRS, IG, MDL. Experimenting with the other heuristics proposed in literature and design of new heuristics with the objective of minimizing Excess Count in SFS algorithms can be explored.

The effective reduct computation strategies developed in the present study are based on SFS search strategy with apt measures to reduce the Excess Count to obtain a shorter length reduct. However, it is observed that it is not possible to avoid Excess Count. One can design one of the following strategies to obtain

redundancy free reduct.

1. By treating any SBS strategy on the reduct obtained by ERCS.
2. ERCS can be appropriately modified by apt hybridization of SBS method at each iteration to tune Excess Count to zero.

These propositions will suffer in scalability aspects and possibilities of efficiently integrating SBS with SFS approach will be explored in future.

The combinations made in Chapters 3 and 4 can be extended to incomplete and hybrid decision systems. The methods discussed are confined to combinations of IQRA_IG and non persistent sampling methods, as a natural consequence of the analysis carried out in Chapters 3 and 4 respectively. A study of all possible combinations can be conducted for getting insight into dynamics of feature selection process.

The proof of concept were demonstrated restricting to simple and stratified sampling (persistent and non persistent) for scalability aspects in the present study. One can study the influence of other sampling methods for enhancing the scalability issues.

Further designing adaptive, hybrid methodologies integrating parallel, distributed and sampling methods to meet the requirements of fields like bio informatics, video analytics, data science should be the important objective of future work.

References

- [1] Michal Baczynski and Balasubramaniam Jayaram. (s, n)- and r-implications: A state-of-the-art survey. *Fuzzy Sets and Systems*, 159(14):1836–1859, 2008.
- [2] Mohua Banerjee, Sushmita Mitra, and Sankar K. Pal. Rough fuzzy mlp: knowledge encoding and classification. *IEEE Transactions on Neural Networks*, 9(6):1203–1216, 1998.
- [3] Jan G. Bazan, Andrzej Skowron, and Piotr Synak. Dynamic reducts as a tool for extracting laws from decisions tables. In Zbigniew W. Ras and Maria Zemankova, editors, *ISMIS*, volume 869 of *Lecture Notes in Computer Science*, pages 346–355. Springer, 1994.
- [4] Jan G. Bazan, Marcin S. Szczuka, and Jakub Wroblewski. A new version of rough set exploration system. In James J. Alpigini, James F. Peters, Jacek Skowronek, and Ning Zhong, editors, *Rough Sets and Current Trends in Computing*, volume 2475 of *Lecture Notes in Computer Science*, pages 397–404. Springer, 2002.
- [5] Richard E. Bellman. *Adaptive control processes - A guided tour*. Princeton University Press, Princeton, New Jersey, U.S.A., 1961.
- [6] Malcolm Beynon. Reducts within the variable precision rough sets model: A further investigation. *European Journal of Operational Research*, 134(3):592–605, 2001.
- [7] Rajen B. Bhatt and M. Gopal. On the compact computational domain of fuzzy-rough sets. *Pattern Recognition Letters*, 26(11):1632–1640, 2005.
- [8] Anders Torvill Bjorvand. Rough enough - a system supporting the rough sets approach. In *SCAI*, pages 290–291, 1997.
- [9] Anders Torvill Bjorvand and Jan Komorowski. Practical applications of genetic algorithms for efficient reduct computation. In Achim Sydow, editor, *15th IMACS World Congress on Scientific Computation, Modelling and Applied Mathematics*, volume 4, pages 601–606. Wissenschaft & Technik Verlag, 1997.
- [10] Piotr Blajdo, Jerzy W. Grzymala-Busse, Zdzislaw S. Hippe, Maksymilian Knap, Teresa Mroczek, and Lukasz Piatek. A comparison of six approaches to discretization-a rough set perspective. In Guoyin Wang, Tian rui Li,

- Jerzy W. Grzymala-Busse, Duoqian Miao, Andrzej Skowron, and Yiyu Yao, editors, *RSKT*, volume 5009 of *Lecture Notes in Computer Science*, pages 31–38. Springer, 2008.
- [11] C. Blake and C. Merz. UCI repository of machine learning data sets. <http://www.ics.uci.edu/~mlearn/MLRepository.html>, 1999.
 - [12] Avrim L. Blum and Pat Langley. Selection of relevant features and examples in machine learning. *Artificial Intelligence*, 97(1-2):245–271, 1997.
 - [13] Avrim L. Blum and Ronald L. Rivest. Training a 3-node neural network is NP-complete. *Neural Networks*, 5:117–127, 1992.
 - [14] Moussa Boussouf and Mohamed Quafafou. Scalable feature selection using rough set theory. In *Rough Sets and Current Trends in Computing*, pages 131–138, 2000.
 - [15] Degang Chen, Qiang He, and Xizhao Wang. Frsvms: Fuzzy rough set based support vector machines. *Fuzzy Sets and Systems*, 161(4):596–607, 2010.
 - [16] J. Chen, Z.H. Jiang, and Y.S. Zhao. Algorithm for rule table reduction based on extension information entropy. *Computer Engineering and Applications*, 43(7):167–169, 2007.
 - [17] Alexios Chouchoulas, J. Halliwell, and Qiang Shen. On the implementation of rough set attribute reduction. In *Proceedings of the 2002 UK Workshop on Computational Intelligence*, pages 18–23, 2002.
 - [18] Alexios Chouchoulas and Qiang Shen. Rough set-aided keyword reduction for text categorization. *Applied Artificial Intelligence*, 15(9):843–873, 2001.
 - [19] Chris Cornelis, Martine De Cock, and Anna Maria Radzikowska. *Fuzzy Rough Sets: From Theory into Practice*, pages 533–552. John Wiley & Sons, Ltd, 2008.
 - [20] Chris Cornelis, Richard Jensen, German Hurtado Martin, and Dominik Slezak. Attribute selection with fuzzy decision reducts. *Inf. Sci.*, 180(2):209–224, 2010.
 - [21] Olitos Dataset. website at. <http://michem.disat.unimib.it/chm/download/datasets.html>.
 - [22] Supriya Kumar De and P. Radha Krishna. Clustering web transactions using rough approximation. *Fuzzy Sets and Systems*, 148(1):131–138, 2004.
 - [23] Shifei Ding and Hao Ding. Research and development of attribute reduction algorithm based on rough set. In *Control and Decision Conference (CCDC), 2010 Chinese*, pages 648–653, May 2010.

- [24] David L. Donoho. High-dimensional data analysis: The curses and blessings of dimensionality. aide-memoire of a lecture. In *AMS Conference on Math Challenges of the 21st Century*, 2000.
- [25] Didier Dubois and Henri Prade. Rough fuzzy sets and fuzzy rough sets. *International Journal of General Systems*, 17(2-3):191–209, 1990.
- [26] Didier Dubois and Henri Prade. Putting rough sets and fuzzy sets together. In Roman Slowinski, editor, *Intelligent Decision Support*, volume 11 of *Theory and Decision Library*, pages 203–232. Springer Netherlands, 1992.
- [27] Didier Dubois and Henri Prade. Similarity versus preference in fuzzy set-based logics. In Ewa Orłowska, editor, *Incomplete Information: Rough Set Analysis*, volume 13 of *Studies in Fuzziness and Soft Computing*, pages 441–461. Physica-Verlag HD, 1998.
- [28] Ivo Duntsch and Gunther Gediga. *Rough set data analysis: A road to non-invasive knowledge discovery*, volume 2 of *Primers Series*. Methodos Publishers (UK), Bangor, 1 edition, 2000.
- [29] Encyclopedia. t-norm. In Ling Liu and M. Tamer Ozsu, editors, *Database Systems*, page 3121. Springer US, 2009.
- [30] Usama M. Fayyad, Gregory Piatetsky-Shapiro, Padhraic Smyth, and Ramasamy Uthrusamy. *Advances in Knowledge Discovery and Data Mining*. The MIT Press, 1996.
- [31] Farideh Fazayeli, Lipo Wang, and Jacek Mandziuk. Feature selection based on rough set theory and expectation maximization clustering algorithm. In Chien-Chung Chan, Jerzy W. Grzymala-Busse, and Wojciech Ziarko, editors, *RSCTC*, volume 5306, pages 272–282, 2008.
- [32] I K Fodor. A survey of dimension reduction techniques. Technical report, Lawrence Livermore National Lab., CA (US), 2002.
- [33] Gottlob Frege. *Grundlagen der Arithmetik: Eine logisch mathematische Untersuchung Aber den Begriff der Zahl*. Wilhelm Koebner, Breslau, 1884.
- [34] Stephen R. Garner. Weka: The waikato environment for knowledge analysis. 1995.
- [35] Salvatore Greco, Benedetto Matarazzo, and Roman Slowinski. Fuzzy similarity relation as a basis for rough approximations. In Lech Polkowski and Andrzej Skowron, editors, *Rough Sets and Current Trends in Computing*, volume 1424 of *Lecture Notes in Computer Science*, pages 283–289. Springer, 1998.

- [36] Jerzy W. Grzymala-Busse and Ming Hu. A comparison of several approaches to missing attribute values in data mining. In *Revised Papers from the Second International Conference on Rough Sets and Current Trends in Computing*, RSCTC '00, pages 378–385, London, UK, UK, 2001. Springer-Verlag.
- [37] Isabelle Guyon and Andre Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.
- [38] H. Yu G.Y. Wang and D.C. Yang. Decision table reduction based on conditional information entropy. *Chinese Journal of Computers*, 25(7):759–766, 2002.
- [39] Jianchao Han, Xiaohua Hu, and TsaoYoung Lin. Feature subset selection based on relative dependency between attributes. In *Rough Sets and Current Trends in Computing*, volume 3066 of *Lecture Notes in Computer Science*, pages 176–185. Springer Berlin Heidelberg, 2004.
- [40] Yasser Fouad Hassan. Rough set genetic programming. *I. J. Comput. Appl.*, 17(3):161–171, 2010.
- [41] Christopher J. Henry and Sheela Ramanna. Parallel computation in finding near neighbourhoods. In Jingtao Yao, Sheela Ramanna, Guoyin Wang, and Zbigniew Suraj, editors, *RSKT*, volume 6954 of *Lecture Notes in Computer Science*, pages 523–532. Springer, 2011.
- [42] Nguyen Sinh Hoa. Some efficient algorithms for rough set methods. In *In Proceedings IPMU'96 Granada (Spain)*, pages 1541–1457, 1996.
- [43] F. Hu and G. Y. Wang. Analysis of the complexity of quick sort for two dimension table. *Chinese Journal of Computers*, 30(6):963–968, 2007.
- [44] Keyun Hu, Yuchang Lu, and Chunyi Shi. Feature ranking in rough sets. *AI Commun.*, 16(1):41–50, May 2003.
- [45] Qinghua Hu, Zongxia Xie, and Daren Yu. Hybrid attribute reduction based on a novel fuzzy-rough model and information granulation. *Pattern Recognition*, 40(12):3509–3521, 2007.
- [46] Qinghua Hu, Daren Yu, and Zongxia Xie. Information-preserving hybrid data reduction based on fuzzy-rough techniques. *Pattern Recognition Letters*, 27(5):414–423, 2006.
- [47] Qinghua Hu, Daren Yu, Zongxia Xie, and Jinfu Liu. Fuzzy probabilistic approximation spaces and their information measures. *IEEE T. Fuzzy Systems*, 14(2):191–201, 2006.

- [48] Xiaohua Hu and Nick Cercone. Learning in relational databases: A rough set approach. *Computational Intelligence*, 11:323–338, 1995.
- [49] Xiaohua Hu, Tsau Young Lin, and Jianchao Han. A new rough sets model based on database systems. *Fundam. Inform.*, 59(2-3):135–152, 2004.
- [50] Laurent Hyafil and Ronald L. Rivest. Constructing optimal binary decision trees is np-complete. *Inf. Process. Lett.*, 5(1):15–17, 1976.
- [51] Michael Ismail and Victor Ciesielski. An empirical investigation of the impact of discretization on common data distributions. In Ajith Abraham, Mario Koppen, and Katrin Franke, editors, *HIS*, volume 105 of *Frontiers in Artificial Intelligence and Applications*, pages 692–701. IOS Press, 2003.
- [52] Jacek Jelonek, Krzysztof Krawiec, and Roman Slowinski. Rough set reduction of attributes and their domains for neural networks. *Computational Intelligence*, 11:339–347, 1995.
- [53] Richard Jensen. Repository of datasets. <http://users.aber.ac.uk/rkj/datasets/index.php>.
- [54] Richard Jensen. Rough set based feature selection: A review. In D. Slezak A. Hassanien, Z. Suraj and P. Lingras, editors, *Rough Computing: Theories, Technologies and Applications*, pages 70–107. IGI Global, IGI, 2008.
- [55] Richard Jensen and Qiang Shen. Finding rough set reducts with ant colony optimization. In *Proceedings of the 2003 UK Workshop on Computational Intelligence*, pages 15–22, 2003.
- [56] Richard Jensen and Qiang Shen. Fuzzy-rough attribute reduction with application to web categorization. *Fuzzy Sets and Systems*, 141(3):469–485, 2004.
- [57] Richard Jensen and Qiang Shen. *Computational Intelligence and Feature Selection: Rough and Fuzzy Approaches*. Wiley-IEEE Press, 2008.
- [58] Richard Jensen and Qiang Shen. New approaches to fuzzy-rough feature selection. *IEEE T. Fuzzy Systems*, 17(4):824–838, 2009.
- [59] Xu Jiucheng and Sun Lin. Knowledge entropy and feature selection in incomplete decision systems. *Applied Mathematics & Information Sciences*, 7(2):829 – 837, 2013.
- [60] George H. John, Ron Kohavi, and Karl Pfleger. Irrelevant features and the subset selection problem. In William W. Cohen and Haym Hirsh, editors, *ICML*, pages 121–129. Morgan Kaufmann, 1994.

- [61] David S. Johnson. Approximation algorithms for combinatorial problems. In *STOC '73: Proceedings of the fifth annual ACM symposium on Theory of computing*, pages 38–49, New York, NY, USA, 1973. ACM.
- [62] Liangjun Ke, Zuren Feng, and Zhigang Ren. An efficient ant colony optimization approach to attribute reduction in rough set theory. *Pattern Recogn. Lett.*, 29(9):1351–1357, July 2008.
- [63] Ron Kohavi and Brian Frasca. Useful feature subsets and rough set reducts, 1994.
- [64] Jan Komorowski, Zdzislaw Pawlak, Lech Polkowski, and Andrzej Skowron. Rough sets: A tutorial, 1998.
- [65] M. Kretowski and J. Stepaniuk. Selection of objects and attributes, a tolerance rough set approach. In *Proceedings of Ninth International Symposium on Methodologies for Intelligent Systems*, pages 169–180, 1996.
- [66] Marzena Kryszkiewicz. Rough set approach to incomplete information systems. *Inf. Sci.*, 112(1-4):39–49, 1998.
- [67] P. Langley. Selection of relevant features in machine learning, 1994.
- [68] Yee Leung and Deyu Li. Maximal consistent block technique for rule acquisition in incomplete information systems. *Inf. Sci.*, 153:85–106, 2003.
- [69] Min Li, Changxing Shang, Shengzhong Feng, and Jianping Fan. Quick attribute reduction in inconsistent decision tables. *Inf. Sci.*, 254:155–180, 2014.
- [70] Jiye Liang and Zongben Xu. The algorithm on knowledge reduction in incomplete information systems. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(1):95–103, 2002.
- [71] T. Y. Lin and N. Cercone. *Rough Sets and Data Mining: Analysis of Imprecise Data*. Kluwer Academic Publishers, Norwell, MA, USA, 1996.
- [72] T.Y. Lin. Neighborhood systems and approximation in database and knowledge base systems. In *Proceedings of the 4th International Symposium on Methodologies for Intelligent Systems*, 1989.
- [73] Zhanao Xue Lin Sun, Jiucheng Xu and Lingjun Zhang. Rough entropy-based feature selection and its application. *Journal of Information and Computational Science*, 8(9):1525–1532, 2011.
- [74] Pawan Lingras. Comparison of neofuzzy and rough neural networks. *Inf. Sci.*, 110(3-4):207–215, 1998.

- [75] Pawan Lingras and Cedric Davies. Application of rough genetic algorithms. *Computational Intelligence*, 17(3):435–445, 2001.
- [76] Huan Liu and Rudy Setiono. Feature selection via discretization. *IEEE Trans. Knowl. Data Eng.*, 9(4):642–645, 1997.
- [77] Huan Liu and Rudy Setiono. Some issues on scalable feature selection. *Expert Systems and Applications*, 15:333–339, 1998.
- [78] W.-N. Liu, Jingtao Yao, and Yiyu Yao. Rough approximations under level fuzzy sets. In Shusaku Tsumoto, Roman Slowinski, Henryk Jan Komorowski, and Jerzy W. Grzymala-Busse, editors, *Rough Sets and Current Trends in Computing*, volume 3066 of *Lecture Notes in Computer Science*, pages 78–83. Springer, 2004.
- [79] Yue-Jin Lv and Nan-Xing Liu. Application of quantum genetic algorithm on finding minimal reduct. In *Granular Computing*, pages 728–733. IEEE, 2007.
- [80] S. Marcus. Tolerance rough sets, cech topologies, learning processes. *Bull. of Polish Academy of Sciences, Technical Sciences*, 42(3):471–487, 1994.
- [81] Mathworks. Matlab. <http://www.mathworks.in/products/matlab/>.
- [82] Zuqiang Meng and Zhongzhi Shi. A fast approach to attribute reduction in incomplete decision systems with tolerance relation-based rough sets. *Inf. Sci.*, 179(16):2774–2793, 2009.
- [83] Maciej Modrzejewski. Feature selection using rough sets theory. In Pavel Brazdil, editor, *ECML*, volume 667 of *Lecture Notes in Computer Science*, pages 213–226. Springer, 1993.
- [84] Michinori Nakata and Hiroshi Sakai. Rough sets handling missing values probabilistically interpreted. In *RSFDGrC (1)*, pages 325–334, 2005.
- [85] Michinori Nakata and Hiroshi Sakai. Applying rough sets to information tables containing missing values. In *ISMVL*, pages 286–291, 2009.
- [86] S. Nanda and S. Majumdar. Fuzzy rough sets. *Fuzzy Sets Syst.*, 45(2):157–160, January 1992.
- [87] D. V. Nguyen, K. Yamada, and M. Unehara. Knowledge reduction in incomplete decision tables using probabilistic similarity-based rough set model. In *ISIS*, pages 147–150, 2011.
- [88] Hung Son Nguyen. Discretization problem for rough sets methods. In Lech Polkowski and Andrzej Skowron, editors, *Rough Sets and Current Trends*

- in Computing*, volume 1424 of *Lecture Notes in Computer Science*, pages 545–552. Springer, 1998.
- [89] Hung Son Nguyen. On exploring soft discretization of continuous attributes. In Sankar K. Pal, Lech Polkowski, and Andrzej Skowron, editors, *Rough-Neural Computing*, Cognitive Technologies, pages 333–350. Springer Berlin Heidelberg, 2004.
 - [90] Hung Son Nguyen and Andrzej Skowron. Quantization of real value attributes - rough set and boolean reasoning approach. In *Proc. of the Second Joint Annual Conference on Information Sciences, Wrightsville Beach, North Carolina, Sept 28 - Oct 1*, pages 34–37, 1995.
 - [91] J. Nieminen. Rough tolerance equality and tolerance black boxes. *Fundamenta Informaticae*, 11(3):289–296, 1988.
 - [92] Michael Ningler, Gudrun Stockmanns, Gerhard Schneider, Oliver Dressler, and Eberhard F. Kochs. Rough set-based classification of eeg-signals to detect intraoperative awareness: Comparison of fuzzy and crisp discretization of real value attributes. In Shusaku Tsumoto, Roman Slowinski, Henryk Jan Komorowski, and Jerzy W. Grzymala-Busse, editors, *Rough Sets and Current Trends in Computing*, volume 3066 of *Lecture Notes in Computer Science*, pages 825–834. Springer, 2004.
 - [93] President’s Council of Advisors on Science and Technology (PCAST). Leadership under challenge: Information technology r & d in a competitive world, an assessment of the federal networking and information technology r & d program. Technical report, USA, August 2007.
 - [94] Aleksander Ohrn. *Discernibility and Rough Sets in Medicine: Tools and Applications*. PhD thesis, 1999.
 - [95] Sankar K. Pal, Biswarup Dasgupta, and Pabitra Mitra. Rough self organizing map. *Appl. Intell.*, 21(3):289–299, 2004.
 - [96] S. Paul and P. Maji. Fuzzy discretization for rough set based gene selection algorithm. In *Emerging Applications of Information Technology (EAIT), 2011 Second International Conference on*, pages 317–320, Feb 2011.
 - [97] Zdzislaw Pawlak. Rough sets. *International Journal of Parallel Programming*, 11(5):341–356, 1982.
 - [98] Zdzislaw Pawlak. Rough sets and decision tables. In Andrzej Skowron, editor, *Symposium on Computation Theory*, volume 208 of *Lecture Notes in Computer Science*, pages 187–196. Springer, 1984.

- [99] Zdzislaw Pawlak. *Rough Sets: Theoretical Aspects of Reasoning About Data*. Kluwer Academic Publishers, Norwell, MA, USA, 1992.
- [100] Zdzislaw Pawlak. Ai and intelligent industrial applications: the rough set perspective. *Cybernetics and Systems*, 31(3):227–252, 2000.
- [101] Zdzislaw Pawlak. A treatise on rough sets. 3700:1–17, 2005.
- [102] Zdzislaw Pawlak, Jerzy W. Grzymala-Busse, Roman Slowinski, and Wojciech Ziarko. Rough sets. *Commun. ACM*, 38(11):88–95, 1995.
- [103] Zdzislaw Pawlak and Andrzej Skowron. Rough sets and boolean reasoning. *Inf. Sci.*, 177(1):41–73, 2007.
- [104] James F. Peters and Sheela Ramanna. Feature selection: Near set approach. In Zbigniew W. Ras, Shusaku Tsumoto, and Djamel A. Zighed, editors, *MCD*, volume 4944 of *Lecture Notes in Computer Science*, pages 57–71, 2007.
- [105] James F. Peters and Piotr Wasilewski. Tolerance spaces: Origins, theoretical aspects and applications. *Inf. Sci.*, 195:211–225, 2012.
- [106] A. Pethalakshmi and K. Thangavel. Performance analysis of accelerated quickreduct algorithm. In *Proceedings of the International Conference on Computational Intelligence and Multimedia Applications (ICCIMA 2007) - Volume 02*, ICCIMA '07, pages 318–322, Washington, DC, USA, 2007. IEEE Computer Society.
- [107] Lech Polkowski and Piotr Artiemjew. On granular rough computing with missing values. In *RSEISP*, pages 271–279, 2007.
- [108] Lech Polkowski, Shusaku Tsumoto, and Tsau Young Lin. A rough set perspective on knowledge discovery in information systems: An essay on the topic of the book. In Lech Polkowski, Shusaku Tsumoto, and TsauY. Lin, editors, *Rough Set Methods and Applications*, volume 56 of *Studies in Fuzziness and Soft Computing*, pages 9–45. Physica-Verlag HD, 2000.
- [109] Yuhua Qian, Chao Li, and Jiye Liang. An efficient fuzzy-rough attribute reduction approach. In Jingtao Yao, Sheela Ramanna, Guoyin Wang, and Zbigniew Suraj, editors, *RSKT*, volume 6954 of *Lecture Notes in Computer Science*, pages 63–70. Springer, 2011.
- [110] Yuhua Qian and Jiye Liang. Combination entropy and combination granulation in rough set theory. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 16(2):179–193, 2008.

- [111] Yuhua Qian, Jiye Liang, Witold Pedrycz, and Chuangyin Dang. Positive approximation: An accelerator for attribute reduction in rough set theory. *Artif. Intell.*, 174(9-10):597–618, 2010.
- [112] Yuhua Qian, Jiye Liang, Witold Pedrycz, and Chuangyin Dang. An efficient accelerator for attribute reduction from incomplete data in rough set framework. *Pattern Recognition*, 44(8):1658–1670, 2011.
- [113] J. R. Quinlan. *C4.5 Programs for Machine Learning*. Morgan Kaufmann, California, 1993.
- [114] Anna Maria Radzikowska and Etienne E. Kerre. A comparative study of fuzzy rough sets. *Fuzzy Sets and Systems*, 126(2):137–155, 2002.
- [115] Adwait Ratnaparkhi. Learning to parse natural language with maximum entropy models. *Machine Learning*, 34(1-3):151–175, 1999.
- [116] J. Rissanen. Modeling by the shortest data description. *Automation*, 14:465–471, 1978.
- [117] Amitava Roy and Sankar K. Pal. Fuzzy discretization of feature space for a rough set classifier. *Pattern Recognition Letters*, 24(6):895–902, 2003.
- [118] Yvan Saeys, Inaki Inza, and Pedro Larranaga. A review of feature selection techniques in bioinformatics. *Bioinformatics*, 23(19):2507–2517, 2007.
- [119] Noelia Sanchez-Marono, Amparo Alonso-Betanzos, and Maria Tombilla-Sanroman. Filter methods for feature selection - a comparative study. In Hujun Yin, Peter Tino, Emilio Corchado, Will Byrne, and Xin Yao, editors, *Intelligent Data Engineering and Automated Learning - IDEAL 2007*, volume 4881 of *Lecture Notes in Computer Science*, pages 178–187. Springer Berlin Heidelberg, 2007.
- [120] B. Wu Z.Z. Shi S.H. Liu, Q.J. Sheng and F. Hu. Research on efficient algorithms for rough set methods. *Chinese Journal of Computers*, 26(5):524–529, 2003.
- [121] Claude Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 623–656, 1948.
- [122] Andrzej Skowron and Cecylia Rauszer. The discernibility matrices and functions in information systems. In Roman Slowinski, editor, *Intelligent Decision Support*, volume 11 of *Theory and Decision Library*, pages 331–362. Springer Netherlands, 1992.
- [123] Andrzej Skowron and Jaroslaw Stepaniuk. Tolerance approximation spaces. *Fundam. Inform.*, 27(2/3):245–253, 1996.

- [124] Dominik Slezak. Approximate reducts in decision tables. Technical report, Research report, Institute of Computer Science, Warsaw University of Technology, 1995.
- [125] Dominik Slezak. Approximate entropy reducts. *Fundam. Inform.*, 53(3-4):365–390, 2002.
- [126] Dominik Slezak and Piotr Wasilewski. Granular sets - foundations and case study of tolerance spaces. In Aijun An, Jerzy Stefanowski, Sheela Ramanna, Cory J. Butz, Witold Pedrycz, and Guoyin Wang, editors, *RSFDGrC*, volume 4482 of *Lecture Notes in Computer Science*, pages 435–442. Springer, 2007.
- [127] Dominik Slezak and Jakub Wroblewski. Order based genetic algorithms for the search of approximate entropy reducts. In Guoyin Wang, Qing Liu, Yiyu Yao, and Andrzej Skowron, editors, *RSFDGrC*, volume 2639 of *Lecture Notes in Computer Science*, pages 308–311. Springer, 2003.
- [128] R. Slowinski and D. Vanderpooten. Similarity relation as a basis for rough approximations. In Wang P., editor, *Advances in Machine Intelligence & Soft-computing*, pages 17–33. Bookwrights, Raleigh, 1997.
- [129] R. Slowinski and D. Vanderpooten. A generalized definition of rough approximations based on similarity. *IEEE Transactions on Data and Knowledge Engineering*, 12:331–336, 2000.
- [130] Roman Slowinski and Jerzy Stefanowski. Handling various types of uncertainty in the rough set approach. In Wojciech Ziarko, editor, *RSKD*, Workshops in Computing, pages 366–376. Springer, 1993.
- [131] Tomasz G. Smolinski and Astrid A. Prinz. Rough sets and neuroscience. In Andrzej Skowron and Zbigniew Suraj, editors, *Rough Sets and Intelligent Systems - Professor Zdzislaw Pawlak in Memoriam*, volume 43 of *Intelligent Systems Reference Library*, pages 493–514. Springer Berlin Heidelberg, 2013.
- [132] Jerzy Stefanowski and Alexis Tsoukiàs. On the extension of rough sets under incomplete information. In *RSFDGrC*, pages 73–81, 1999.
- [133] Tomasz Strakowski and Henryk Rybinski. A new approach to distributed algorithms for reduct calculation. *T. Rough Sets*, 9:365–378, 2008.
- [134] Chao-Ton Su and Jyh-Hwa Hsu. An extended chi2 algorithm for discretization of real value attributes. *IEEE Trans. Knowl. Data Eng.*, 17(3):437–441, 2005.

- [135] N. Suguna and K. Thanushkodi. A weighted bee colony optimisation hybrid with rough set reduct algorithm for feature selection in the medical domain. *IJGCRSIS*, 2(2):123–140, 2011.
- [136] Lin Sun, Jiucheng Xu, and Shuangqun Li. Knowledge reduction based on granular computing from decision information systems. In Jian Yu, Salvatore Greco, Pawan Lingras, Guoyin Wang, and Andrzej Skowron, editors, *RSKT*, volume 6401 of *Lecture Notes in Computer Science*, pages 46–53. Springer, 2010.
- [137] Lin Sun, Jiucheng Xu, Shuangqun Li, Xizheng Cao, and Yunpeng Gao. New approach for feature selection by using information entropy. *Journal of Information and Computational Science*, 8(12):2259–2268, 2011.
- [138] Robert Susmaga. Parallel computation of reducts. In Lech Polkowski and Andrzej Skowron, editors, *Rough Sets and Current Trends in Computing*, volume 1424 of *Lecture Notes in Computer Science*, pages 450–458. Springer Berlin Heidelberg, 1998.
- [139] Roman W. Swiniarski and Andrzej Skowron. Rough set methods in feature selection and recognition. *Pattern Recognition Letters*, 24(6):833–849, 2003.
- [140] K. Thangavel and A. Pethalakshmi. Dimensionality reduction based on rough set theory: A review. *Appl. Soft Comput.*, 9(1):1–12, 2009.
- [141] Pethalakshmi A. Thangavel, K. and P. Jaganathan. A comparative analysis of feature selection algorithms based on rough set theory. *International Journal of Soft Computing*, 1(4):288–294, 2006.
- [142] David Tian, Xiao-Jun Zeng, and John A. Keane. Core-generating approximate minimum entropy discretization for rough set feature selection in pattern classification. *Int. J. Approx. Reasoning*, 52(6):863–880, 2011.
- [143] J. Tick and C. Fodor. Fuzzy implications and inference processes. *Computers and Artificial Intelligence*, 24(6):591–602, 2005.
- [144] J. Umakant, K. Sudhakar, P. M. Mujumdar, and C. Raghavendra Rao. Ranking based uncertainty quantification for a multifidelity design approach. *Journal of Aircraft*, 44(2):410–419, 2007.
- [145] J. Umakant, K. Sudhakar, P. M. Mujumdar, and C. Raghavendra Rao. Uncertainty modeling for expensive functions: A rank transformation approach. *Engineering Letters*, 14(1):154–166, 2007.
- [146] Arizona State University. Feature selection repository. <http://featureselection.asu.edu/datasets.php>.

- [147] Cuiru Wang and Fangfang Ou. An attribute reduction algorithm based on conditional entropy and frequency of attributes. In *Intelligent Computation Technology and Automation (ICICTA), 2008 International Conference on*, volume 1, pages 752–756, Oct 2008.
- [148] Guoyin Wang. Extension of rough set under incomplete information systems. In *Fuzzy Systems, 2002. FUZZ-IEEE'02. Proceedings of the 2002 IEEE International Conference on*, volume 2, pages 1098–1103, 2002.
- [149] Guoyin Wang and Feng Hu. Quick knowledge reduction based on divide and conquer method in huge data sets. In Ashish Ghosh, Rajat K. De, and Sankar K. Pal, editors, *PReMI*, volume 4815 of *Lecture Notes in Computer Science*, pages 312–315. Springer, 2007.
- [150] Pai-Chou Wang. Highly scalable rough set reducts generation. *J. Inf. Sci. Eng.*, 23(4):1281–1298, 2007.
- [151] Xiangyang Wang, Jie Yang 0002, Xiaolong Teng, Weijun Xia, and Richard Jensen. Feature selection based on rough sets and particle swarm optimization. *Pattern Recognition Letters*, 28(4):459–471, 2007.
- [152] Xizhao Wang, Eric C. C. Tsang, Suyun Zhao, Degang Chen, and Daniel S. Yeung. Learning fuzzy rules from fuzzy samples based on rough set technique. *Inf. Sci.*, 177(20):4493–4514, 2007.
- [153] Xuren Wang, Famei He, and Lizhen Liu. Application of rough set theory to intrusion detection system. In *Granular Computing, 2007. GRC 2007. IEEE International Conference on*, pages 562–562, 2007.
- [154] Wei Wei, Jiye Liang, and Yuhua Qian. A comparative study of rough sets for hybrid data. *Inf. Sci.*, 190:1–16, 2012.
- [155] Wiki. on t-norm. <http://en.wikipedia.org/wiki/T-norm>.
- [156] Wikipedia. Hill climbing. http://en.wikipedia.org/wiki/Hill_climbing.
- [157] Ian H. Witten, Eibe Frank, and Mark A. Hall. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, Amsterdam, 3 edition, 2011.
- [158] Jakub Wroblewski. Finding minimal reducts using genetic algorithms. In *Proc. of the Second Annual Join Conference on Information Sciences*, pages 186–189, Wrightsville Beach, NC., 1995.
- [159] Jakub Wroblewski. Theoretical foundations of order-based genetic algorithms. *Fundam. Inform.*, 28(3-4):423–430, 1996.

- [160] Jiucheng Xu and Lin Sun. New reduction algorithm based on decision power of decision table. In Guoyin Wang, Tian rui Li, Jerzy W. Grzymala-Busse, Duoqian Miao, Andrzej Skowron, and Yiyu Yao, editors, *RSKT*, volume 5009 of *Lecture Notes in Computer Science*, pages 180–188. Springer, 2008.
- [161] Bing Xue, Mengjie Zhang, and Will N. Browne. Particle swarm optimisation for feature selection in classification: Novel initialisation and updating mechanisms. *Applied Soft Computing*, 18(0):261 – 276, 2014.
- [162] Yong Yang, Zhengrong Chen, Zhu Liang, and Guoyin Wang. Attribute reduction for massive data based on rough set theory and mapreduce. In Jian Yu, Salvatore Greco, Pawan Lingras, Guoyin Wang, and Andrzej Skowron, editors, *RSKT*, volume 6401 of *Lecture Notes in Computer Science*, pages 672–678. Springer, 2010.
- [163] Jingtao Yao, Yiyu Yao, and Wojciech Ziarko. Probabilistic rough sets: Approximations, decision-makings, and applications. *Int. J. Approx. Reasoning*, 49(2):253–254, 2008.
- [164] Yiyu Yao. Relational interpretation of neighborhood operators and rough set approximation operators. *Information Sciences*, 111(1-4):239–259, 1998.
- [165] Yiyu Yao. Decision-theoretic rough set models. In Jingtao Yao, Pawan Lingras, Wei-Zhi Wu, Marcin S. Szczuka, Nick Cercone, and Dominik Slezak, editors, *RSKT*, volume 4481 of *Lecture Notes in Computer Science*, pages 1–12. Springer, 2007.
- [166] Yiyu Yao and Rong Fu. The concept of reducts in pawlak three-step rough set analysis. *T. Rough Sets*, 16:53–72, 2013.
- [167] Yiyu Yao, Yan Zhao, and Jue Wang. On reduct construction algorithms. In Guoyin Wang, James F. Peters, Andrzej Skowron, and Yiyu Yao, editors, *RSKT*, volume 4062 of *Lecture Notes in Computer Science*, pages 297–304. Springer, 2006.
- [168] Liu Yong, Xiong Rong, and Rong Chu Jian. Quick attribute reduction with hash. *Chinese Journal of Computers*, 8:1493–1499, 2009.
- [169] Lofti A. Zadeh. Fuzzy sets. *Information and Control*, 8:338–353, 1965.
- [170] Qingguo Zhang, Xuefeng Zheng, and Zhangyan Xu. Efficient attribute reduction algorithm based on incomplete decision table. In *Intelligent Computation Technology and Automation, 2009. ICICTA '09. Second International Conference on*, volume 4, pages 192–195, 2009.

- [171] Baiting Zhao, Xijun Chen, and Qingshuang Zeng. Incomplete hybrid attributes reduction based on neighborhood granulation and approximation. In *Mechatronics and Automation, 2009. ICMA 2009. International Conference on*, pages 2066–2071, 2009.
- [172] Yan Zhao, Feng Luo, S. K. Michael Wong, and Yiyu Yao. A general definition of an attribute reduct. In Jingtao Yao, Pawan Lingras, Wei-Zhi Wu, Marcin S. Szczuka, Nick Cercone, and Dominik Slezak, editors, *RSKT*, volume 4481 of *Lecture Notes in Computer Science*, pages 101–108. Springer, 2007.
- [173] Yan Zhao, Yiyu Yao, and Feng Luo. Data analysis based on discernibility and indiscernibility. *Inf. Sci.*, 177(22):4959–4976, 2007.
- [174] Ning Zhong, Juzhen Dong, and Setsuo Ohsuga. Using rough sets with heuristics for feature selection. *J. Intell. Inf. Syst.*, 16(3):199–214, 2001.
- [175] Wojciech Ziarko. Variable precision rough set model. *J. Comput. Syst. Sci.*, 46(1):39–59, 1993.
- [176] Huasheng ZOU and Changsheng ZHANG. Efficient algorithm for knowledge reduction in incomplete information system. *Journal of Computational Information Systems*, 8(6):2531–2538, 2012.
- [177] B.R. Yang Z.Y. Xu, Z.P. Liu and W. Song. A quick attribute reduction algorithm with complexity of $\max O(|C||U|), O(|C|^2|U/C|)$. *Chinese Journal of Computers*, 29(3):391–399, 2006.