Quantitative Association Rule Mining using Multi-objective Particle Swarm Optimization

Jayashree Piri, Raghunath Dey

Abstract--Association rule mining is a technique of discovering interesting correlation among items present in a dataset. To discover such interesting association rules, more than one objective need to be optimized rather than exploiting a single objective. This motivated to pose the association rule mining algorithm as a multi objective problem and use particle swarm optimization based multi objective metaheuristics to solve this problem as they tend to explore the global search space effectively in less time. This paper considers confidence, comprehensibility, interestingness as three objective for mining association rule and use a pareto based Particle swarm optimization to extract useful and interesting rules from quantitative database. The results of these algorithms are evaluated on various quality measures and are found to be suitable.

Keywords- Association rule mining, Multi objective Particle swarm optimization, repairing operator, complete overlap, partial overlap, Join operator, quality measures.

----------<>----------

1 INTRODUCTION

Data mining is an important tool for discovering knowledge from a huge database. So it is known as an integral part of knowledge discovery process in database (KDD). ***Association rule mining*** (ARM) is a data mining task which aims to find correlation between the items in a large database. These correlation can be visualized in the form of IF-THEN rule i.e., if A then C, where A is the antecedent (set conditions) and C is the consequent (set of outcomes). The antecedent and consequent side can contain more than one item. However there should not be any common items between the two sides.

Association rule mining has received a great attention in research and has evolved from seminal to state of the art over last decade. However, there is a significant challenge in applying these algorithms to the real world applications. Association rule generation is considered as an NP-hard problem as it needs to search a search space of 2n, where n is the number of items. Because of the exponential growth of search space and database dependent thresholds, many researchers have proposed the use of evolutionary algorithms. In 2004, Ghosh et al.[1] showed that association rule can be mined by considering more than one objective .

But they restricted to the mining of association rule from market basket data. In 2002, Coello et al. [2] developed a multi objective version of particle swarm optimization algorithm (MOPSO) and since then it has been applied to various domain. It is believed that MOPSO tends to better over other meta heuristics algorithms because of its two simple operations. So to explore the benefits of MOPSO, A MOPSO based association rule mining technique has been developed for mining quantitative databases.

- *Jayashree Piri is currently working as Asst. Professor in the Dept. of Information Technology at Balasore College of Engineering and Technology.Email: jayashreepiri@gmail.com*
- *Raghunath Dey is currently working as Asst. Professor in the Dept. of Computer Science and Engineering at Balasore College of Engineering and Technology raghunathdey@gmail.com*

The paper is organized as follows. In section 2 an overview of the existing association rule mining techniques are described. Section 3 covers details of proposed work including the repairing and join operators. Experimental setups along with results and comparisons are put in section 4. At last section 5 includes the conclusions and future works.

2 RELATED WORKS

2.1 Classical Approaches for Association Rule Mining

The Association Rule Mining problem was first proposed in 1993 by Agrawal et al. [3] Since then it has evolved a lot and have become well explored research area. As stated by Agrawal et al.[3] association rule mining technique can be divided into two subproblems (1) generation of frequent pattern (2) generation of rules from these frequent patterns. Most of the existing algorithms present in the literature focus on the first part. The first among this is the AIS [3] algorithm. In this algorithm only one item can be present in the consequent part. So this algorithm generates association rule in the form of $X \cap Y \rightarrow Z$. However Agrawal and Srikant made a great improvement over the AIS algorithm and proposed an efficient approach called Apriori algorithm[4]. Apriori used a pruning technique which avoided the combination of non-frequent items, while guaranteeing completeness. However there are two bottlenecks of Apriori algorithm. (1) Its complex candidate generation technique require more time and space. (2) It requires multiple scans over the datasets. FP-growth [5] (frequent pattern growth) uses an extended prefix-tree (FP-tree) structure to store the database in a compressed form. The algorithms for association rule mining discussed are for market basket data. For quantitative association rule mining, the data are first mapped to boolean domain and then binary association rule mining algorithm are applied[6]. So discretization of numeric attribute is an extra overhead for quantitative association rule mining problem.

2.2 Evolutionary Approaches for Association Rule Mining.

There are several population based search algorithms exist in the literature, which can explore and exploits the search space to generate the association rules in polynomial time. In DMARG[7] and DDMARG[8], each allele in the chromosome represents a given item rather than a rule. These techniques may generate huge number of rules, some of which may be uninteresting and might have occurred by chance. To combine the two phase approach of DMARG and DDMARG, ARMGA [9] encoded the rules as a chromosome. In this approach a length k is set for the rules, and then a chromosome of length k+1 is used. Position 1 to k specifies the index of the items and position 0 contains the cut point between the antecedent and consequent. But they have restricted the rule length to k. The standard GA for boolean association rule mining cannot be directly applied for numeric association rule mining. There are several alternatives for representing continuous attributes inside a population. One of the first attempts is the GENAR[10] , which deals with only those rules which involve all the database attributes except the last one which acts as consequent. So it is necessary to prepare the data to indicate the tool which attribute form the part of antecedent and which one is the consequent. But there may be other association rule exists in the database which involves variable no of attributes in either side. GAR [11](Genetic Association Rules) is an extension of GENAR algorithm which uses a representation scheme where each attribute is codified through a set of three consecutive genes, the first one is the index of the variable, and then the interval is represented by the lower and upper bound. Furthermore a variable length representation for each item set is used in this case. Kwasnicka and Switalski have proposed an extension EGAR [12](Extended GAR) which uses a michigan encoding technique where each individual encodes a single frequent item set, and consists of two chromosomes. One chromosome contains the continuous attribute and the second one contains the discrete attributes. But it also works in two phase as in GAR. Salleb et al.[13] proposed a QuantMiner , based on a genetic algorithm that dynamically discovers good intervals in association rules by optimizing both support and confidence. But this also requires user defined parameters like minimum support, minimum confidence, and a user defined rule template or the format of the quantitative association rules.

So far the study, presented association rule generation using single objective. However generating association rules in a single phase requires more than one objective to be considered. This inspired researchers to pose the association rule mining problem as a multi objective optimization problem. The pioneering work in this context is done by Ghosh et al.[1]. They considered three objectives and presented a pareto based association rules. Wakabi-Waiswa et al.[14] extracted association rules by using the three objectives used by Ghosh et al.[1] along with J-measure and perplexity but they transformed the multi objective problem to a single objective problem to obtain the association rules. Though Ghosh et al.[1] proposed pareto based association rule mining but they restricted themselves to market basket data. For treating quantitative association rule mining as a multi objective problem, the pioneering work was carried out by Alatas et al. [15] they proposed a multi objective differential evolution algorithm for numeric association rule mining (MODENAR) to solve the quantitative association rule mining. But in MODENAR, they described that amplitude of the intervals in each of the attributes which conforms interesting rule must be smaller. That means between two individuals that cover the same no of records and have the same no of attributes the one whose intervals are smaller gives the best information. Suppose there are two rules of same support and both the rules have same no of attributes but they are completely different rules:

R1: A1[5.0, 10.5] → A3[22.0, 38.9]

R2 :A3[5.7, 7.5] → A4[23.0, 24.6]

In this case if the 2nd rule is selected (since having smaller amplitude of intervals), then a valid rule like R1 may miss out. However for more information on evolutionary based association rule mining the interested reader may refer to [16].

3 PRPOSED METHOD

3.1 MOPSO Based Quantitative ARM

A MOPSO based quantitative association rule mining technique is used here in order to extract association rules from numeric data. A brief description of this work is given below:

3.1.1 Particle Representation

The particles are represented by using Michigan approach of encoding, where each particle of the swarm represents a rule. Each particle have a position vector and velocity vector. Each component of the particle's position vector has 3 parts. The 1st part is always a random number between 0 to 1. If this no is ≤ 0.33, then the corresponding attribute is in the antecedent, if it is ≥0.66, then it present in the consequent, however any other value denotes absence of the attribute in that rule. The 2nd and 3rd part represent the lower and upper bound of the attribute interval respectively. The lower and upper bound must be within the range of the attribute's minimum and maximum allowable value. For example suppose there are 3 attributes A1(2.5-9.5), A2(1.00-20.00) and A3(22.5-25.5), then the pictorial representation of the rule

A3[22.6, 23.5]→ A2[3.2, 16.0] is shown in figure 1.

A1			A2			A3		
0.5	3.4	6.9	0.789	3.2	16.0	0.21	22.6	23.5

Figure 1: Particle representation in quantitative ARM.

3.1.2 Fitness Evaluation

The fitness of the particle is calculated by using confidence, comprehensibility and interestingness. These are given by the following equations adopted from(Ghosh and Nath) 2004 [1] .

$$\text{Confidence } (A \rightarrow C) = P(C|A) \quad (1)$$

$$\text{Comprehensibility} = \log(1+|C|)/\log(1+|A \cup C|) \quad (2)$$

$$\text{Interestingness} = [SUP(A \cup C)/SUP(A)] * [SUP(A \cup C)/SUP(C)] * [1- (SUP(A \cup C)/ |D|)] \quad (3)$$

Where |C | and |A U C| are the no of attributes in the consequent and in the whole rule respectively. |D| is the total no of records in the database. Here a constraint is specified that, the confidence of a rule should be >=50%. So instead of simple domination check, a constraint domination check [17] is carried out to find the non dominated particles.

3.1.3 Guide Selection

The task of a global guide *gbest* in particle swarm optimization is to explore the search space to move towards the global optimum. As multi objective optimization problem has multiple tradeoff solutions, there are more than one solutions present in the repository. For each particle a guide is selected from the repository. Since all the solutions in the repository are optimal solutions, other criteria must be used to select a single guide for each particle. So a crowding distance based approach is used to select a guide from the repository [18]. In this approach the particles are sorted in descending order of their crowding distance(CD). Then for each particle a guide is selected from top 10% of the repository. To exploit the search space personal guide *pbest* is used in particle swarm optimization. Initially the personal best position of each particle is initialized as the particle itself. After velocity and position update, if the current position is dominated by the position in memory i.e., the previous *pbest*, then the position in memory kept; otherwise the current position replaces the one in memory; if neither of them is dominated by the other, then randomly choose any one.

3.1.4 Velocity and position Updation

The velocity of each particle in the swarm is calculated by using the following equation.

$$v_{id}=w_i * v_{id} + c1 * \text{rand}() * (p_{id} - x_{id}) + c2 * \text{rand}() * (p_{gd} - x_{id}) \quad (4)$$

Where d is dimension or number of attributes, c1 and c2 are the cognitive and social learning factor respectively, w is the inertia weight. x_{id} and v_{id} are the position and velocity of the particle i at the dimension d respectively. p_{id} is the position of the personal best and p_{gd} is the position of the global best.

The positions of each particle are updated using the following equation.

$$x_{id}=x_{id} + v_{id} \quad (5)$$

Where x_{id} and v_{id} are the position and velocity of particle i at dimension d respectively.

3.1.5 Repository Maintenance

Initially the repository contains all the constraint non dominated solutions of the initial swarm. While inserting into the repository a ***complete rule overlap*** check is done. That means if the attributes in antecedent and consequent part of rule to be inserted (R1) is same as any of existing rule in the repository (R2) and one is completely overlapped with another, and their supports are equal then the rule with smaller interval should be kept. For example:

R1: A1[5.0, 10.5]→ A3[22.0, 38.9]

R2 :A1[5.7, 7.5]→ A3[23.0, 33.6]

Let their supports are equal then the 2nd rule must be kept in the repository. Here the equation which is used to calculate the amplitude of the interval is adopted from [15] i.e.,

$$\text{amplitude of interval} = \frac{1}{m} * \sum_{i=1}^{m} \frac{u_i - l_i}{max(A_i) - min(A_i)} \quad (6)$$

Where m is the no of attribute present in that rule. After each move of the swarm the repository gets updated. A particle can reside in the repository if it is not dominated by the particles already present in the repository. If any existing repository particles become dominated by this insertion then it should be removed. Since the size of the external repository is fixed by the user there may be a chance of overflow. If overflow happens then again particle are arranged in descending order of their CD. Select any particle from the bottom 10% of the repository which comprises the most crowded particle in the archive for the purpose of replacement.

3.1.6 Repairing Operator

After position update of each particle if some particles go outside their boundary or lower bound takes greater value than that of upper bound, at that time repairing mechanism is used so as to explore only the feasible region. If lower bound takes bigger value than the upper bound within their boundaries then simply their values are exchanged. If the variables or attributes in the new solution are outside their boundaries that are if the lower bound takes a value which is less than its allowable minimum value or upper bound takes a value which is bigger than its allowable maximum value, then repairing operator is applied as follows:

$$x'_{j,i} = \begin{cases} x_j^{min} + \left(\frac{|x_j^{min} - x'_{j,i}|}{2} \right) & \text{if } x'_{j,i} < x_j^{min} \\ x_j^{max} - \left(\frac{|x'_{j,i} - x_j^{max}|}{2} \right) & \text{if } x'_{j,i} > x_j^{max}, \end{cases}$$

Where x_j^{min} and x_j^{max} are the minimum and maximum allowable value for j_{th} attribute respectively. $x'_{j,i}$ is the calculated position for the i_{th} particle at j_{th} dimension.

3.1.7 Join Operator

Here filtering is used as a post processing method. When the specified no of generations completed, check ***partial overlap*** between rules. A rule is said to be partially overlapped with another rule if they both have same attribute in antecedent and consequent and for each attribute, one rule's lower bound or upper bound lies in between the lower bound and upper bound of another rule. For example:

R1: A1[5.0, 10.5] → A3[22.0, 38.9]

R2 :A1[5.7, 12.5]→ A3[20.0, 33.6]$

If such overlap found between any two rules then combine or join the two rules into one as follows:

Rnew: A1[5.0, 12.5] →A3[20.0, 38.9]

Algorithm 1 : Proposed Algorithm for MOPSO based ARM

1) for i= 1 to N (population size) do
    a. Initialize P[i] randomly.(P is the population)
    b. Initialize V[i]=0.
    c. Initialize the personal best of each particle
    d. PBEST[i]=P[i].
2) end for.
3) Initialize the iteration counter COUNT=0
4) for i= 1 to N do
    a. Decode p[i] to get the antecedent, consequent and the whole rule.
    b. Evaluate P[i] i.e., calculate confidence, comprehensibility and interestingness of the decoded rule.
5) end for.
6) Store the non dominated solutions found in P into external archive A.
7) Repeat
    a. Compute the crowding distance(CD) values of each non dominated solution in the archive A and sort them in descending order.
    b. for i= 1 to N do
        i. Randomly select the gbest guide for P[i] from top 10% of the sorted archive A.
        ii. Compute the new velocity and position of P[i].
        iii. If P[i] goes beyond the boundaries, then apply Repairing operator.
        iv. Evaluate P[i].
    c. end for.
    d. Update the repository.
    e. Update the personal best position of each particle in P.
    f. Increment iteration counter COUNT.
8) Until maximum no of iterations is reached.
9) Apply post processing methods.
10) Calculate the ***quality measures*** listed above for the discovered rules.

4 EXPERIMENTAL EVALUATIONS

4.1 Dataset Description

The proposed technique was implemented on different datasets, both synthetic and real world with satisfactory results. The descriptions of the used datasets which are collected from UCI repository are given below:

Iris Plants Dataset:
R.A. Fisher is the creator of this dataset. Number of Instances are 150. Number of Attributes is 4 numeric, 1 predictive attributes. There are no missing Attribute Values present.

Glass Identification Dataset
B. German has created this dataset. 214 Number of Instances are present. Number of Attributes are 10 (including an Id\#) plus the class attribute. All attributes are continuously valued. There are no missing Attribute Values present.

Wine Quality Dataset
This dataset is Created by Paulo Cortez (Univ. Minho), Antonio Cerdeira, Fernando Almeida, Telmo Matos and Jose Reis. Numbers of Instances are red wine - 1599; white wine - 4898. Number of Attributes are 11 + output attribute. There are no missing Attribute Values are present.

Abalone Dataset
The Original owners of database are Marine Resources Division, Marine Research Laboratories - Taroona, Department of Primary Industry and Fisheries, Tasmania. This dataset is for predicting the age of abalone from physical measurements. 4177 Number of Instances are present and

Number of Attributes are 8. There are no missing attribute values.

4.2 Quality Measures

For quantitative association rule mining problem, some ***quality measures*** [19] are used in order to get the quality of extracted rules. These are listed below:

Coverage: Coverage measures how often a rule A→ B is applicable in a database.

$$\text{Coverage} = P(A) \quad (8)$$

Recall: Recall is the fraction of relevant instances that are retrieved.

$$\text{Recall} = P(A \mid B) \quad (9)$$

Accuracy: Accuracy means rate of correctness over a dataset.

$$\text{Accuracy} = P(AC) + P(\sim A \sim C) \quad (10)$$

Leverage: It measures the difference of A and B appearing together in the dataset and what would be expected if A and B were statistically dependent.

$$\text{Leverage} = P(B \mid A) - P(A)P(B) \quad (11)$$

Added value: Added value measures whether the proportion of transactions containing A is greater than the proportion of transactions containing B among all transactions.

$$\text{Added value} = P(B \mid A) - P(B) \quad (12)$$

Jaccard: Jaccard coefficient accesses the distance between antecedent and consequent as the fraction of cases covered by both with respect to the fraction of cases covered by one of them. Higher value indicate that A and B tend to cover the same cases.

$$\text{Jaccard} = \frac{P(AC)}{P(A)+p(C)-P(AC)} \quad (13)$$

Certainty factor: It is interpreted as a measure of variation of the probability that B is in a transaction when we consider only those transaction where A is.

$$\text{Cirtainty factor} = \frac{P(C|A)-P(C)}{1-P(C)} \quad (14)$$

Where A and B denotes the antecedent and consequent part of a rule respectively.

4.3 Parameter Setting

The result of the proposed algorithm has been compared with that of MOGA(Multi-objective Genetic Algorithm). So the parameters values for the experiment has given in table 1

Table 1: Parameter setting

Methods	population Size	Iterations	Crossover Rate	Mutation rate	w	c1	c2
MOGA	100	100	0.8	0.02	-	-	-
MOPSO	100	100	-	-	0.5	1.5	2.0

4.4 Results and Analysis

The average results of 10 runs, obtained for quantitative association rule mining are shown in Table 2, Table 3 and Table 4. The proposed algorithms are evaluated using the quality measures described in section 4.2. It has been observed that the MOPSO based association rule mining gives better results than MOGA based association rule mining. However in case of some datasets, MOPSO results are comparable to the results of MOGA based association rule mining. In quantitative association rule mining, all the three objective function value obtained from the MOPSO based approach is better value than that of MOGA based approach. The value of the measures like coverage, recall, jaccard, certainty factor are more better in case of MOPSO based quantitative association rule mining. The concept of using guide in MOPSO has helped in exploring the search space efficiently. Due to this, more no of rules (or correlation among items) are extracted from the databases than MOGA, in a less time. Using crowding distance as a guide selection criteria and repository maintenance technique, helped in obtaining a diversed set of rules. Rules obtained from quantitative association rule mining are neither completely overlapped nor partially overlapped.

Table 2: Objective functions values for quantitative association rule mining.

Data sets Used	Confidence		Comprehensibility		Interestingness	
	MOPSO	MOGA	MOPSO	MOGA	MOPSO	MOGA
Abalone	0.96	0.95	0.80	0.80	0.38	0.36
Wine quality	0.91	0.84	0.82	0.70	0.22	0.15
Iris	0.95	0.91	0.75	0.75	0.43	0.40
Glass	0.93	0.87	0.82	0.67	0.33	0.17

Table 3: Quality measures for quantitative association rule mining

Measures	Method	Abalone	Wine Quality	Iris	Glass
Coverage	MOPSO	0.36	0.33	0.39	0.30
	MOGA	0.35	0.27	0.33	0.32
Recall	MOPSO	0.62	0.42	0.73	0.54
	MOGA	0.58	0.33	0.67	0.38
Leverage	MOPSO	0.74	0.66	0.74	0.71
	MOGA	0.74	0.63	0.73	0.61
Added value	MOPSO	0.39	0.21	0.44	0.46
	MOGA	0.36	0.30	0.44	0.21
Jaccard	MOPSO	0.59	0.38	0.69	0.51
	MOGA	0.56	0.28	0.60	0.34
Accuracy	MOPSO	0.76	0.55	0.83	0.78
	MOGA	0.72	0.62	0.81	0.58
Certainty factor	MOPSO	0.93	0.58	0.88	0.81
	MOGA	0.92	0.50	0.82	0.49

Table 4: average time span and average no. of rules for quantitative ARM

Data sets Used	Average time (sec)		Average no. of rules	
	MOPSO	MOGA	MOPSO	MOGA
Abalone	5.27	5.45	21	15
Wine quality	7.22	4.67	21	11
Iris	0.25	2.21	7	5
Glass	0.40	0.35	8	5

5 Conclusions and Future Work

This paper presents a evolutionary based multi objective association rule mining technique for quantitative data. The performance of the algorithms is evaluated using various measures. The experimental analysis showed that, MOPSO based association rule mining provides a better result than MOGA based association rule mining. As a future work an evolutionary based multi objective algorithm can be designed that can work on mixed (Boolean and quantitative) datasets. Also evolutionary based multi objective optimization techniques can be used to design a fuzzy based association rule mining.

ACKNOWLDGEMENT

We are sincerely and heartily grateful to Center for Soft Computing Research (CSCR) of the Indian Statistical Institute (ISI), Kolkata, for their support and guidance throughout the dissertation writing. We are truly indebted and thankful to Machine Intelligence Unit (MIU) of ISI, Kolkata for providing me such a peaceful environment throughout our work.

REFERENCES

[1] Ashish Ghosh and Bhabesh Nath. Multi-objective rule mining using genetic algorithms. Information Science, 163(1-3):123–133, 2004.

[2] C. A. Coello Coello and M. S. Lechuga. MOPSO: A proposal for multiple ob-jective particle swarm optimization. In Proceedings of the Evolutionary Compu-tation on 2002. CEC '02. Proceedings of the 2002 Congress - Volume 02, CEC '02, pages 1051–1056, Washington, DC, USA, 2002. IEEE Computer Society.

[3] Rakesh Agrawal, Tomasz Imieli´nski, and Arun Swami. Mining association rules between sets of items in large databases. SIGMOD Rec., 22(2):207–216, 1993.

[4] Rakesh Agrawal and Ramakrishnan Srikant. Fast algorithms for mining associa-tion rules in large databases. In Proceedings of the 20th International Conference on Very Large Data Bases, VLDB '94, pages 487–499, San Francisco, CA, USA, 1994. Morgan Kaufmann Publishers.

[5] Jiawei Han, Jian Pei, Yiwen Yin, and Runying Mao. Mining frequent patterns without candidate generation: A frequent-pattern tree approach. Data Mining and Knowledge Discovery, 8(1):53–87, 2004.

[6] Ramakrishnan Srikant and Rakesh Agrawal. Mining quantitative association rules in large relational tables. SIGMOD, 25(2): 1-12, 1996.

[7] P. Deepa Shenoy, K. G. Srinivasa, K. R. Venugopal, and L. M. Patnaik. Evo-lutionary approach for mining association rules on dynamic databases. In Pro-ceedings of the 7th Pacific-Asia conference on Advances in knowledge discovery and data mining, PAKDD'03, pages 325–336, Berlin, Heidelberg, 2003. Springer-Verlag.

[8] P. Deepa Shenoy, K. G. Srinivasa, K. R. Venugopal, and Lalit M. Patnaik. Dynamic association rule mining using genetic algorithms. Intell. Data Anal., 9(5):439–453, 2005.

[9] J. Mata, J.L. Alvarez, and J.C. Riquelme. Mining numeric association rules with genetic algorithms. In 5th International Conference on Artificial Neural Networks and Genetic Algorithms,ICANNGA, pages 264–267, 2001.

[10] Xiaowei Yan, Chengqi Zhang, and Shichao Zhang. Genetic algorithm-based strat-egy for identifying association rules without specifying actual minimum support. Expert System Application, 36(2):3066–3076, 2009.

[11] Jacinto Mata, Jose-Luis Alvarez, and Jose-Cristobal Riquelme. Discovering nu-meric association rules via evolutionary algorithm. In Proceedings of the 6th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining, PAKDD 02, pages 40–51. Springer-Verlag, 2002.

[12] H.Kwasnicka and K.Switalski. Discovery of association rules from medical data-classical and evolutionary approaches. pages 204–217, 2006.

[13] Ansaf Salleb-Aouissi, Christel Vrain, and Cyril Nortet. Quantminer: A genetic algorithm for mining quantitative association rules. In Proceedings of the 20th international joint conference on Artifical intelligence, IJCAI'07, pages 1035– 1040, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc.

[14] Wakabi Waiswa, Peter P, and Venansius Baryamureeba. Extraction of interesting association rules using genetic algorithms. International Journal of Computing and ICT Research, 2(1):26–33, 2008.

[15] Bilal Alatas, Erhan Akin, and Ali Karci. MODENAR:multi-objective differential evolution algorithm

for mining numeric association rules. Applied Soft Computing, 8(1):646–656, 2008.

[16]Mara J. del Jesus, Jos A. Gmez, Pedro Gonzlez, and Jos M. Puerta. On the discovery of association rules by means of evolutionary algorithms. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 1(5):397–415, 2011.

[17] K.Dev. Multi-Objective Optimization using Evolutionary Algorithms. Wiley, New York, 2001.

[18] Carlo R. Raquel and Prospero C. Nval, Jr. An effective use of crowding distance in multiobjective particle swarm optimization. In Proceedings of the 2005 conference on Genetic and evolutionary computation, GECCO '05, pages 257-264, New York, NY, USA, 2005. ACM.

[19] Fabrice Guillet and Howard ER Hamilton. Quality Measures in Data Mining. 2007.