# Discovering dependencies among mined association rules with population-based metaheuristics

Iztok Fister Jr.
University of Maribor
Maribor, Slovenia
iztok.fister1@um.si

Akemi Galvez
University of Cantabria
Santander, Spain
galveza@unican.es

Eneko Osaba
TECNALIA
Derio, Spain
eneko.osaba@tecnalia.com

Javier Del Ser
University of the Basque Country
(UPV/EHU), TECNALIA
Bilbao, Spain
javier.delser@tecnalia.com

Andres Iglesias
University of Cantabria
Santander, Spain
iglesias@unican.es

Iztok Fister
University of Maribor
Maribor, Slovenia
iztok.fister@um.si

## ABSTRACT

Stochastic population-based nature-inspired metaheuristics have been proven as a robust tool for mining association rules. These algorithms are very scalable, as well as very fast compared with some deterministic ones that search for solutions exhaustively. Typically, algorithms for association rule mining identify a lot of rules depending, on the transaction database and number of attributes. Therefore, evaluating these rules is very complex. On the other hand, establishing the relationships between discovered association rules can be considered as a very hard problem that cannot easily be solved manually. In this paper, we propose a new algorithm based on stochastic population-based nature-inspired metaheuristics for discovering dependencies among association rules.

## CCS CONCEPTS

• **Theory of computation** → **Evolutionary algorithms**; • **Information systems** → *Expert systems*;

## KEYWORDS

association rule mining, graphs, complex networks, population-based metaheuristics

**ACM Reference Format:**
Iztok Fister Jr., Akemi Galvez, Eneko Osaba, Javier Del Ser, Andres Iglesias, and Iztok Fister. 2019. Discovering dependencies among mined association rules with population-based metaheuristics. In *Genetic and Evolutionary Computation Conference Companion (GECCO '19 Companion), July 13–17, 2019, Prague, Czech Republic.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3319619.3326833

## 1 INTRODUCTION

In the current information age, data accompany us in almost every aspect of our lives. The data revolution has already changed many traditional human habits. For example, when interacting with websites on the Internet, our interactions are stored, and our behavior is analyzed further by website providers in order to get the best advertisement serving strategies. On the other hand, we usually operate with different loyalty cards that store our buying habits for further analysis.

By the same token, sport is also considered as an area where, nowadays, a lot of data are produced during the process of sport training. Modern sport training strategies enable athletes to wear sport trackers during sport activity. These have emerged in the form of sport watches or mobile devices and allow to measure the performance of an athlete during sports activities. Analyzing these activities can later be used for:

- monitoring the improvements of athletes,
- planning the sport training sessions based on existing activities,
- adapting sport training sessions,
- planning the proper nutrition, etc.

Association Rule Mining (ARM) [1] is a popular data mining method for identification of dependence rules between features in a transaction database. There are many traditional association rule mining algorithms, but the most used are Apriori proposed by Agrawal [2], Eclat introduced by Zaki et al. [18], and FP-Growth developed by Han et al. [8]. The main characteristic of these algorithms is that they search for the association rule exhaustively, i.e., by enumerating all possibilities. Consequently, the traditional algorithms are very time consuming, especially, when the problem to be solved is complex.

Recently, the scientific community has developed a lot of methods for mining association rules that are based on stochastic population based nature-inspired metaheuristics. Actually, the popular nature-inspired algorithms, like Differential Evolution (DE) [15], Particle Swarm Optimization (PSO) [10], and Genetic Algorithm (GA) [9] have been applied for tackling this problem. Usually, such methods produce a lot of association rules. Although we can filter these rules according to the value of fitness function, some interesting rules that may not have a high fitness value can easily be

---

*GECCO '19 Companion, July 13–17, 2019, Prague, Czech Republic*

ACM ISBN 978-1-4503-6748-6/19/07...$15.00
https://doi.org/10.1145/3319619.3326833

overlooked. Therefore, one of the biggest challenges referring to the association rule mining is the proper interpretation of results.

On the other hand, the mined rules with lower support and confidence values express similar characteristics, and, therefore, do not contribute much to comprehending some characteristics of an athlete in sports training. As a result, two solutions of the problem could be used:

- to increase the values of the minimum support and the minimum confidence in order to eliminate the solutions with lower support and confidence, or
- to analyze dependencies among the mined association rules, and to post-process the mined characteristics of the athlete automatically.

In this paper, we are focused on the second solution. In line with this, we would like to obtain answers to the following questions:

- Can we identify the features arising as antecedents in the majority of rules?
- Can we identify the features having an impact on the consequences in the majority of rules?

Interestingly, these issues led us to the domain of complex network. They appear in many aspects of our life, such as, for example, biological networks [13], molecular networks [14], social networks [5], Internet [17], etc. The main characteristics of these networks are that they have complex and irregular connectivity patterns. Their organization of vertices and edges are challenging to understand and characterize [11].

In our study, the mined association rules are presented as complex networks, where graphs based on the results of stochastic nature-inspired population-based algorithms depict association rules by connecting the features with edges. Graphs may be directed or undirected [4]. In directed graph, edges are referred as arcs and represent ordered pairs of vertices. Usually, these edges are associated with weights designating the power of relationship between vertices. The higher the weights, the powerful the connection. On the other hand, edges in undirected graphs are represented as unordered pairs of vertices, where direction is unimportant. Here, we are focused on the unordered graphs.

These graphs are appropriate for community detection, where the group of tightly knit vertices construct a so-called cohesive subgroup. Consequently, the first issue can be realized by searching for those antecedents that are connected with the higher number of the other antecedents. The second issue demands searching for vertices with the higher number of consequences. In our study, the former vertices are also called as central, while the latter as subsequent. However, the major advantage of using the complex networks is a wide range of visualization tools that can help the user to analyze the complex relationships in an easy way.

The paper of Fister et al. [6] presents the starting point for the study, where the so-called BatMiner was presented for mining the characteristics of athletes in sports training. The previous work finished with an analysis of the huge set of mined association rules, and a conclusion that this amount of rules is hard to analyze manually. Actually, the study integrates the BatMiner in a new method that is capable of joining the results of the algorithm with their interpretation. The results of the proposed method were encouraging, and showed that the complex networks are the potential direction for analyzing the results of association rule mining in the future.

The structure of this paper is as follows: The ARM problem is explained In Section 2. Section 3 is devoted to describing the proposed method for discovering dependencies between mined association rules with the BA algorithm. The results of the method are illustrated in Section 4. Section 5 concludes the paper and outlines the possible directions for future work.

## 2 ASSOCIATION RULE MINING

This section briefly presents formal definition of ARM. Let us suppose, a set of objects $O = \{o_1, \ldots, o_n\}$ and transaction database $DB$ are given, where each transaction $T$ is a subset of objects; in other words, $T \subseteq O$. Then, an association rule can be defined as an implication:

$$X \Rightarrow Y, \tag{1}$$

where $X \subset O$, $Y \subset O$, in $X \cap Y = \emptyset$. The following two measures are defined for evaluating the quality of an association rule [3]:

$$conf(X \Rightarrow Y) = \frac{n(X \cup Y)}{n(X)}, \tag{2}$$

$$supp(X \Rightarrow Y) = \frac{n(X \cup Y)}{N}, \tag{3}$$

where $conf(X \Rightarrow Y) \geq C_{min}$ denotes confidence and $supp(X \Rightarrow Y) \geq S_{min}$ support of association rule $X \Rightarrow Y$. Thus, $N$ in Eq. (3) represents the number of transactions in the transaction database $DB$ and $n$ is the number of repetitions of a particular rule $X \Rightarrow Y$ within $DB$. Here, $C_{min}$ denotes minimum confidence and $S_{min}$ minimum support. This means that only those association rules with confidence and support higher than $C_{min}$ and $S_{min}$ are taken into consideration, respectively. Typically, an association rule is called strong, if it satisfies both posted objectives, i.e., the minimum support and the minimum confidence.

## 3 THE PROPOSED METHOD

The proposed method consists of the following steps:

- association rules generation,
- discovering dependencies,
- analysis,
- evaluation.

In the first step, the association rules are generated using some stochastic population-based nature-inspired algorithm [7]. Discovered dependencies between mined association rules are dedicated to highlight those characteristics of the rules that are characteristic for an athlete's behavior. The features in the mined rules are then visualized using complex networks. Finally, the obtained results need to be evaluated by the real sports trainer. All mentioned steps are outlined in detail in the next subsections.

### 3.1 Association rules generation

The association rule mining presents the first step of the proposed method. In this step, association rules are mined from the transaction database of the realized sports training sessions [12]. The bat algorithm for association rule mining (BatMiner), proposed by Fister et al. in [6], was used for the task. This algorithm discovers a lot

**Table 1: Features and their attributes.**

| Feature | Attribute domain |
|---|---|
| DISTANCE | {NULL, SHORT, MEDIUM, LONG} |
| DURATION | {NULL, SHORT, MEDIUM, LONG} |
| HEART_RATE | {NULL, LOW, MEDIUM, HIGH} |
| CALORIES | {NULL, SMALL, MEDIUM, HIGH} |
| WEATHER | {NULL, SUNNY, CLOUDY, RAINY, SNOWY} |
| TYPE | {NULL, EASY, INTERVALS, POWER, ENDURANCE} |
| NUTRITION | {NULL, POOR, MODERATE, GOOD} |
| FOOD | {NULL, PROTEINS, CARBOHYDRATES, FAT, FRUITS} |
| BEVERAGES | {NULL, WATER, JUICE, ISO, COKE} |
| REST | {NULL, AFTER_TRAINING, NO} |
| NIGHT_REST | {NULL, BAD, MEDIUM, GOOD} |
| INJURIES | {NULL, NO, LOW, MEDIUM, HIGH} |
| CRAMPS | {NULL, NO, LOW, HIGH} |
| HEALTH_PROBLEMS | {NULL, NO, LITTLE, YES} |

of association rules having their support and confidence measures above the prescribed minimum values of $S_{min}$ and $C_{min}$.

The features and their corresponding attributes, used in this study, are illustrated in Table 1. Indeed, the attributes of these features are obtained either by digitizing measured values obtained from sports activities into proper intervals (like DISTANCE, DURATION, HEART_RATE, etc.) or by interviewing the athletes before training sessions (like NUTRITION, FOOD, REST, etc.). Interestingly, each feature can also be assigned the value of NULL, which means that the feature is not present in the association rule. On the other hand, the feature is concatenated with its corresponding attribute by a character $'\_'$ to form the feature-attribute pair that is presented as an item of the association rule. For more information about this subject, the interested reader is invited to look at the aforementioned study [6].

Typically, the large set of discovered association rules were obtained after applying the BatMiner. Obviously, this set needs to be analyzed using an algorithm for discovering mutual dependencies, in order to highlight the hidden relationships on the one hand, and to expose the more important ones, on the other. The proposed algorithm is illustrated in detail in the remainder of the section.

## 3.2 Discovering dependencies

Discovering dependencies is represented as an optimization problem that is defined formally as follows. Let us assume, a set of mined association rules $R = \{r_1, \ldots, r_N\}$ is given, where each rule $r_i$ is in form of $X \Rightarrow Y$, and $N$ denotes the number of rules. Then, the subset $S \subset R$ of association rules $S = \{r_{\pi_1}, \ldots, r_{\pi_D}\}$ is selected randomly from the $R$ that is estimated with regards to two measures:

$$\begin{aligned} LHS &= \frac{m(X)}{D}, \text{ and} \\ RHS &= \frac{m(Y)}{D}, \end{aligned} \tag{4}$$

where $m(X)$ denotes the number of times that $X$ occurs as an antecedent or Left-Hand-Site (LHS) in the rules, $m(Y)$ is the number of times that $Y$ occurs as a consequence or Right-Hand-Side (RHS) in the rules, and $D$ determines the size of the subset $S$. Finally, the quality of the subset is determined as follows:

$$f(S) = \frac{LHS(S) + RHS(S)}{2}, \tag{5}$$

subject to

$$\begin{aligned} \forall X : |X| \geq M_X, \text{ and} \\ \forall Y : |Y| \geq M_Y, \end{aligned} \tag{6}$$

where $M_X$ denotes the feasible number of antecedents, and $M_Y$ the feasible number of consequences.

The motivation behind the inequalities in Eq. (6) is to discover relations between a specific number of items that occur as antecedents and the number of items occurring as consequences. Indeed, relations with only one object in consequence could be especially interesting for users, because some special behavioral characteristics of an observed athlete could thus be exposed.

Although a lot of stochastic nature-inspired population based algorithms could be taken for solving this problem, the Bat Algorithm (BA) [16] was used due to its simplicity. In order to prepare the algorithm properly, the following components of the original BA must be modified:

- representation of individuals,
- fitness function evaluation.

In the remainder of the paper, modifications of the mentioned components are illustrated in detail.

Individuals in the proposed BA algorithm for discovering dependencies within mined association rules are represented as real-valued vectors:

$$\mathbf{x}_i = \{x_{i,0}, \ldots, x_{i,D}\}, \quad \text{for } i = 1, \ldots, Np, \tag{7}$$

with elements $x_{i,j} \in [0, 1]$ presenting an index of the corresponding association rule, where $Np$ is a population size. As a matter of fact, the genotype (i.e., elements $x_{i,j}$) is mapped to the phenotype of an individual (i.e., association rule $r_k$) in compliance with the following equation:

$$k = \left\lfloor \frac{x_{i,j}}{N} \cdot N \right\rfloor, \tag{8}$$

where $k$ determines the corresponding $r_k \in R$.

The fitness function evaluation is performed according to Eq. (5). Let us emphasize that the task of the optimization is to maximize the value of the fitness function.

## 3.3 Analysis

The results obtained by BA for discovering dependencis among mined association rules are presented as a complex network [11]. The network is constituted from the best $M$ association rules according to fitness value by modeling the obtained results in the form of a complex interconnection of nodes by edges. The nodes present items (i.e., feature-attribute pairs) arising in association rules, while edges are connections between these. There are two kinds of edges that join vertices, arising as:

- antecedents, and
- consequences.

The first kind of joining is denoted by solid black lines, while the second by blue colored dotted lines in the complex network graphs.

In order to identify characteristics of the complex networks, the following two metrics are used in our study: an Average Neighborhood Degree (*AND*), and a Network Density (*ND*). The *AND* metric is defined as:

$$AND = \frac{1}{|V|} \sum_{v \in V} deg(v), \tag{9}$$

where $deg(v)$ denotes the degree of vertex $v$, and $|V|$ is the maximum number of vertices. On the other hand, the *ND* metric is expressed as:

$$ND = \frac{2 \cdot |E|}{|V| \cdot (|V| - 1)}, \tag{10}$$

where $|E|$ denotes the actual number of connections and the expression $|V| \cdot (|V| - 1)/2$ refers to the maximum number of connections in complex networks.

## 3.4 Evaluation

In this step, a community detection in complex networks is searched for. The evaluation of the relationship between the items in complex networks showed that there are two forms of vertex distribution in general:

- trees, and
- cliques.

Typically, trees identify a situation, where antecedents/consequences of two or more association rules are joined in a chain, while cliques represent a cohesive group of feature-attribute pairs that are connected to each other tightly. Typically, these pairs arise as antecedents/consequences in more than one rule, and, therefore, represent the tightly knit characteristics of the athlete in training. The critical vertices are searched for among these cohesive group members, while the decisive ones also need to be searched for among the other antecedents as well.

In order to get the estimation as reliable as possible, the obtained results need to be estimated from the practical aspect. Therefore, the real trainer, who was asked for help, analyzed the results of the proposed method, and assessed the value of the proposed method in the real-world.

# 4 EXPERIMENTS AND RESULTS

The goal of our experimental work was to show that the dependencies between mined association rules can be discovered using the stochastic population-based nature-inspired algorithms. Thus, hidden relationships between feature-attribute pairs of association rules can be established, where the most important antecedents could be identified on the one hand, and the most important consequences on the other. In line with this, three experiments were conducted, where:

- the number of antecedents were set to $M_X \geq 2$ and the number of consequences to $M_X \geq 1$,
- the number of antecedents were set to $M_X \geq 2$ and the number of consequences to $M_X \geq 2$,
- the number of antecedents were set to $M_X \geq 3$ and the number of consequences to $M_X \geq 3$.

As can be seen from this setup, it is expected that the complexity of the mined association rules $M$ are increased from test to test. On the other hand, the number $M$ might be reduced. Although the number of association rules $M$, forming the complex networks, is crucial, analysis of its influence on the results was left as a direction for future work.

In our study, the BA algorithm for discovering dependencies between mined association rules used the parameter settings as illustrated in Table 2.

**Table 2: The parameter settings of BA.**

| Parameter | Symbolic name | Value |
|---|---:|---:|
| Population size | *NP* | 30 |
| Number of generations | *MAX_GEN* | 1000 |
| Loudness | *A* | 0.5 |
| Pulse emission rate | *r* | 0.5 |

In the remainder of the section, the experimental data are presented. Then, the results of the experiments are illustrated. Finally, the obtained results are analyzed and commented on by the real sports trainer at the end of the section.

## 4.1 Experimental data

BatMiner mined the association rules from a transaction database consisting of 80 transactions. Each transaction is an essence of one sport training session realized by a professional cyclist, and highlights attributes of 14 features that determine the characteristics of the athlete before and during the training session.

Interestingly, BatMiner found 4,191 different association rules with minimum confidence and minimum support higher than $C_{min} \geq 0.1$ and $S_{min} \geq 0.1$, respectively. These rules present the starting point for the proposed stochastic population-based algorithm for discovering dependencies.

## 4.2 Results

The experiments were conducted as follows. At first, the BA algorithm selects the association rules from an archive generated by BatMiner [6] according to constraints given in Eq. (6). The best $M$ association rules according to fitness function are then selected for the second phase, where the complex networks of feature-attribute pairs are built. In this preliminary study, these networks were then analyzed manually with the help of the characteristic metrics *AND* and *ND*.

Table 3 illustrates ten the best association rules, from which the complex network in Fig. 1 was built. In this test, constraints were set as $M_X \geq 2 \wedge M_X \geq 2$.

As can be seen from Fig. 1, the connected graph is obtained from the features-attribute pairs. This means that no isolated vertices exist in the graph. In this case, the complex network consist of $|V| = 21$ vertices, and $|E| = 27$ edges. As a result, the average vertex density is expressed as $AND = 1.2857$, while the network density is $ND = 0.1286$. Since the last metric refers to a relatively sparsely connected graph, the antecedent nodes, denoted as black solid lines, constitute only trees (chains) of the feature-attribute pairs.

Table 4 illustrates ten the strongest association rules discovered in the database that are needed for building the complex networks. Let us mention that the association rules are more complex compared with their counterparts in Table 3 that is a consequence of the stronger constraints set in Eq. (6).

As can be seen from Fig. 2, the complex networks of ten the best mined association rules also form the connected graph of vertices

**Table 3: The strongest association rules discovered in database by $M_X \geq 2 \wedge M_X \geq 2$.**

DURATION_SHORT ∧ CALORIES_SMALL => TYPE_EASY ∧ FOOD_FRUITS
DISTANCE_SHORT ∧ HR_LOW ∧ CALORIES_SMALL => TYPE_EASY ∧ INJURIES_NO ∧ HEALTH_PROBLEMS_NO
NUTRITION_GOOD ∧ FOOD_CARBOHYDRATES ∧ NIGHT_REST_GOOD ∧ CRAMPS_NO => DURATION_MEDIUM ∧ HR_LOW ∧ CALORIES_MEDIUM
DISTANCE_SHORT ∧ HR_LOW => TYPE_EASY ∧ BEVERAGES_WATER ∧ HEALTH_PROBLEMS_NO
DISTANCE_SHORT ∧ DURATION_SHORT => CALORIES_SMALL ∧ NUTRITION_POOR ∧ INJURIES_NO
DISTANCE_SHORT ∧ DURATION_SHORT => TYPE_EASY ∧ REST_NO ∧ INJURIES_NO
DURATION_SHORT ∧ HR_HIGH => TYPE_INTERVALS ∧ HEALTH_PROBLEMS_NO
HR_LOW ∧ WEATHER_CLOUDY => REST_AFTER_TRAINING ∧ INJURIES_NO
DISTANCE_SHORT ∧ CALORIES_SMALL => BEVERAGES_WATER ∧ INJURIES_NO ∧ CRAMPS_NO
CALORIES_SMALL ∧ TYPE_EASY ∧ INJURIES_NO => DISTANCE_SHORT ∧ DURATION_SHORT ∧ HR_LOW

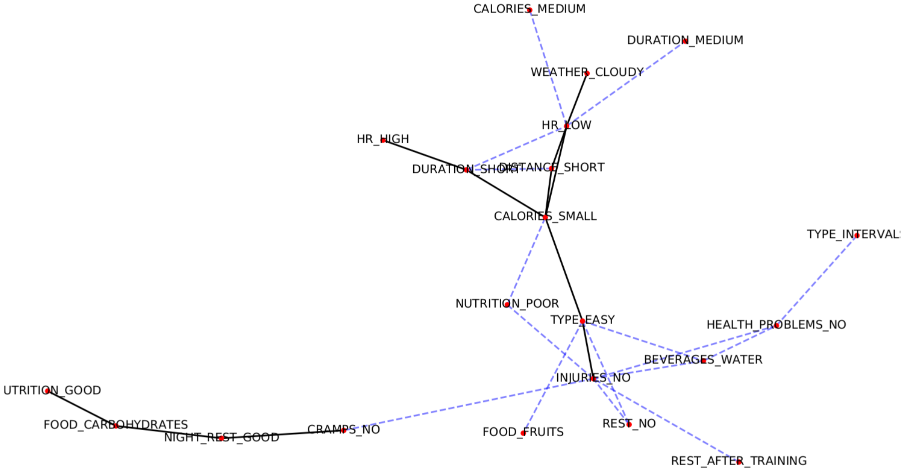

**Figure 1: Visualization of mined association rules by $M_X \geq 2 \wedge M_X \geq 2$.**

**Table 4: The strongest association rules discovered in the database by $M_X \geq 3 \wedge M_X \geq 3$.**

DISTANCE_LONG ∧ DURATION_LONG ∧ HR_MEDIUM ∧ CALORIES_HIGH ∧ TYPE_ENDURANCE => FOOD_PROTEINS ∧ BEVERAGES_ISO∧
*REST_NO ∧ NIGHT_REST_GOOD ∧ INJURIES_LOW ∧ CRAMPS_NO*
DISTANCE_LONG ∧ HR_LOW ∧ CALORIES_HIGH ∧ WEATHER_SUNNY => FOOD_FRUITS ∧ BEVERAGES_WATER ∧ HEALTH_PROBLEMS_NO
TYPE_EASY ∧ INJURIES_NO ∧ CRAMPS_NO => DURATION_SHORT ∧ CALORIES_SMALL ∧ WEATHER_SNOWY
NUTRITION_GOOD ∧ FOOD_CARBOHYDRATES ∧ NIGHT_REST_GOOD ∧ CRAMPS_NO => DURATION_MEDIUM ∧ HR_LOW ∧ CALORIES_MEDIUM
FOOD_FRUITS ∧ BEVERAGES_WATER ∧ NIGHT_REST_BAD ∧ HEALTH_PROBLEMS_NO => DISTANCE_LONG ∧ HR_LOW ∧ CALORIES_HIGH∧
*WEATHER_SUNNY ∧ NUTRITION_GOOD*
DISTANCE_SHORT ∧ DURATION_SHORT ∧ CALORIES_SMALL => WEATHER_SUNNY ∧ TYPE_EASY ∧ NUTRITION_POOR ∧ FOOD_PROTEINS∧
*NIGHT_REST_BAD ∧ INJURIES_NO*
DISTANCE_SHORT ∧ DURATION_SHORT ∧ HR_LOW => CALORIES_SMALL ∧ WEATHER_SUNNY ∧ TYPE_EASY ∧ FOOD_PROTEINS∧
*NIGHT_REST_BAD ∧ INJURIES_NO*
DISTANCE_SHORT ∧ HR_LOW ∧ CALORIES_SMALL => TYPE_EASY ∧ REST_AFTER_TRAINING ∧ INJURIES_NO
CALORIES_SMALL ∧ TYPE_EASY ∧ NUTRITION_POOR ∧ CRAMPS_NO => DISTANCE_SHORT ∧ DURATION_SHORT ∧ HR_LOW
DISTANCE_SHORT ∧ DURATION_SHORT ∧ HR_LOW => CALORIES_SMALL ∧ TYPE_EASY ∧ CRAMPS_NO

representing feature-attributes pairs. Now, the complex networks consist of $|V| = 29$ vertices, and $|E| = 42$ edges. This means that the average node density is set to $AND = 1.4483$, and the network density to $ND = 0.1034$. However, the antecedent vertices in this graph, excelled by the higher average vertex density, also constitute cliques of the feature-attribute pairs, and, thus, they indicate the tightly connected relationships between these pairs.

## 4.3 Analyzing complex networks

Two issues are put at the beginning of the section, i.e., to identify those feature-attribute pairs that emerge in the majority of the association rules as either: (1) antecedents or (2) consequences. As mentioned before, both issues are transformed into searching for central and subsequent vertices in complex networks.

The results of analyzing complex network in Fig. 1 are presented in Table 5, where the feature-attribute pairs can be seen emerging in the majority of the corresponding association rules. In the table, the columns 'Ante' and 'Cons' denote the number of rules where these pairs have arisen as antecedents and consequences, respectively, while the column 'All' is the number of all rules in which the pair

**Table 5: Results of analyzing the complex network in Fig. 1.**

| Feature-attribute | Ante | Cons | All | Type |
|---|---|---|---|---|
| DISTANCE_SHORT | 5 | 1 | 6 | clique 1 |
| CALORIES_SMALL | 4 | 1 | 5 | clique 1,center |
| HR_LOW | 3 | 2 | 5 | clique 1 |
| INJURIES_NO | 1 | 5 | 6 | subsequent |

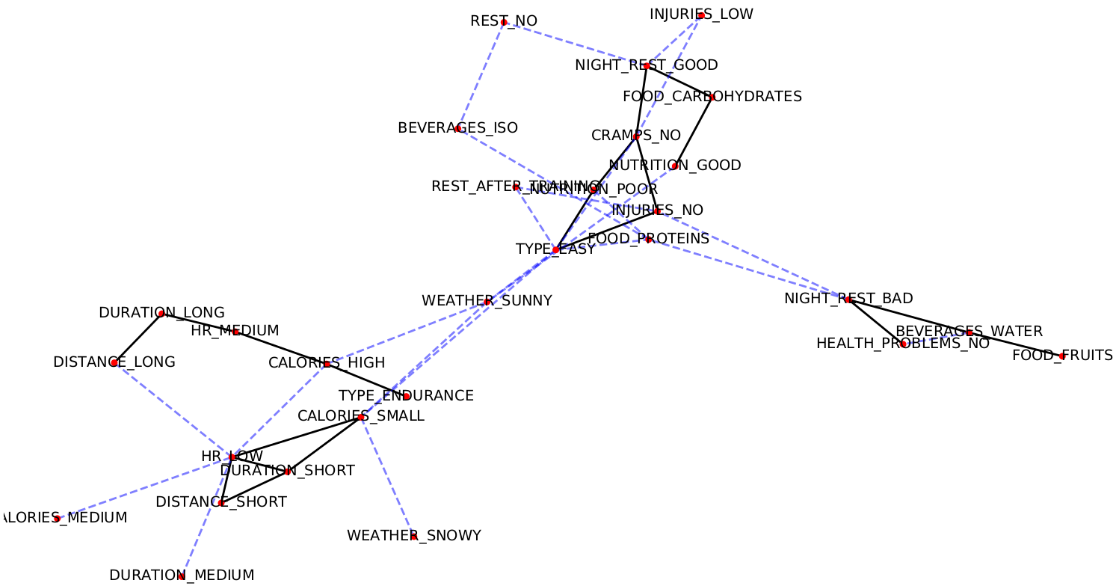

**Figure 2: Visualization of mined association rules by $M_X \geq 3 \wedge M_X \geq 3$.**

has emerged. The column 'Type' denotes the type of corresponding vertex in the complex network. For instance, the central vertex represents the CALORIES_SMALL feature-attribute pair that forms a clique 1:

{CALORIES_SMALL, DISTANCE_SHORT, HR_LOW}.

Additionally, the central vertex is connected with DURATION_SHORT and TYPE_EASY pairs. Although this vertex is not emerged in the maximum number of association rules, it is joint with the maximum number of antecedent vertices.

A subsequent vertex is the INJURIES_NO pair that is connected even with eight consequences, as follows:

{BEVERAGES_WATER, CALORIES_SMALL, CRAMPS_NO, HEALTH_PROBLEMS_NO,
NUTRITION_POOR, REST_AFTER_TRAINING, REST_NO, TYPE_EASY}.

The complex network in Fig. 2 is more interesting, because here the corresponding graph is more dense. The results of analyzing this complex network are illustrated in Table 6 from which it can be seen that even two cliques are detected as follows:

**Table 6: Results of analyzing the complex network in Fig. 2.**

,

| Feature-attribute | Ante | Cons | All | Type |
|---|---|---|---|---|
| HR_LOW | 4 | 3 | 7 | Clique 1+2,center |
| CALORIES_SMALL | 3 | 3 | 6 | Clique 2 |
| DISTANCE_SHORT | 4 | 1 | 5 | Clique 1 |
| DURATION_SHORT | 3 | 2 | 5 | Clique 1+2 |
| TYPE_EASY | 2 | 4 | 6 | Subsequent |

{DISTANCE_SHORT, HR_LOW, DURATION_SHORT}, and
{HR_LOW, DURATION_SHORT, CALORIES_SMALL}.

Indeed, vertices HR_LOW and DURATION_SHORT create the common membership of the cliques. In this graph, the HR_LOW pair is indicated as a central vertex due to its higher degree of 7.

On the other hand, the subsequent vertex TYPE_EASY is incident by eight edges as follows:

{CALORIES_SMALL, CRAMPS_NO, REST_AFTER_TRAINING, FOOD_PROTEINS,
INJURIES_NO, NIGHT_REST_BAD, NUTRITION_POOR, WEATHER_SUNNY}.

## 4.4 Comments of the real sport trainer

In order to get the proper interpretation of analyzing the complex networks, the real sport trainer was asked for help. His comments are as follows.

The complex network depicted in Fig. 1 consists of two components joined by the consequence edge incident to vertices CRAMPS_NO and INJURIES_NO. This means that the generated association rules base on the antecedents from two different components: The first is formed from the chain of the feature-attributes pairs, like NUTRIENT_GOOD, FOOD_CARBO, NIGHT_REST_GOOD, and CRAMPS_NO, while the second component is more complex, due to indicating the highly knitted vertices forming clique 1 (cohesive subgroup of vertices).

More interesting is analyzing the subsequent vertex, from which it can be concluded that, if the athlete is not injured (INJURIES_NO), he/she does not have any health problems (HEALTH_PROBLEMS_NO) and cramps (CRAMPS_NO), needs only poor nutrition (NUTRITION_POOR) of small energy value (CALORIES_SMALL), can consume only water beverage (BEVERAGE_WATER) during the sport activity, while the rest is unimportant (REST_AFTER_TRAINING_NO, REST_NO), when conducting the easy training sessions (TYPE_EASY). This conclusion complied strongly with the sport theory.

Moreover, the complex network depicted in Fig. 2 is divided evenly into four components that are joined between each other with more edges arisen as consequences in the association rules. The most interesting is the component consisting of tightly knitted vertices forming two cliques. From these components it can be deduced that the low heart rate (HR_LOW) of the observed athlete is characteristic for training sessions of medium duration (duration_medium), long distance (distance_long), where the energy consumption is medium or high (calories_medium, calories_high).

Analyzing the subsequent vertex justifies the fact that, if the athlete is not injured injuries_no, the observed athlete is healthy (health_problems_no, cramps_no), consumes only a small energy (calories_small) taken in with poor nutrition (calories_small), and resting is unimportant (rest_after_training_no, rest_no), and likes sunny weather conditions (weather_sunny) in the training session of easy type (type_easy). However, this assertion is also in compliance with the sport theory.

In summary, analyzing of the second complex network is more interesting for the practice, because it is more general and, consequently, can discover the deeper relationships between feature-attributes pairs.

## 5 CONCLUSION

This paper presented a new method for discovering dependencies among mined association rules. This is an extension of the existing BatMiner algorithm for association rule mining, where we are focused on discovering dependencies among mined association rules with the BA algorithm. Actually, the BatMiner is capable of generating a lot of association rules that are hard to be interpreted manually. Therefore, the new method is proposed that integrates the BatMiner with the BA for discovering dependencies between mined association rules, analyzing, and evaluation. These rules are analyzed in the sense of the complex networks, and interpreted (evaluated) by the real sport trainer.

The proposed method was applied on a transaction database that presents the training data of a professional cyclist. The results of experiments showed that the proposed method can find dependencies efficiently among different association rules. Thus, the mined association rules are analyzed as complex networks, where the feature-attribute pairs are presented as vertices, and edges denote connections between these. The results of analyzing showed new insights into complex data, that were confirmed by the real sport trainer.

Obviously, this is not the end of the development of the proposed method. In the future, we would like to test other fitness functions along with different population-based metaheuristics. Additionally, we would also apply this method on the mined association rules from other datasets.

## ACKNOWLEDGMENT

I. Fister Jr. acknowledge the financial support from the Slovenian Research Agency (Research Core Founding No. P2-0057). I. Fister acknowledge the financial support from the Slovenian Research Agency (Research Core Founding No. P2-0041). E. Osaba and J. Del Ser would like to thank the Basque Government for its funding support through the EMAITEK program. A. Iglesias and A. Galvez would like to thank the the financial support from the projects TIN2017-89275-R (AEI/FEDER, UE) and PDE-GIR (H2020, MSCA program, ref. 778035).

## REFERENCES

[1] Rakesh Agrawal, Tomasz Imieliński, and Arun Swami. 1993. Mining Association Rules Between Sets of Items in Large Databases. In *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data (SIGMOD '93)*. ACM, New York, NY, USA, 207–216. https://doi.org/10.1145/170035.170072

[2] Rakesh Agrawal, Heikki Mannila, Ramakrishnan Srikant, Hannu Toivonen, A Inkeri Verkamo, et al. 1996. Fast Discovery of Association Rules. *Advances in knowledge discovery and data mining* 12, 1 (1996), 307–328.

[3] Rakesh Agrawal and Ramakrishnan Srikant. 1994. Fast Algorithms for Mining Association Rules in Large Databases. In *Proceedings of the 20th International Conference on Very Large Data Bases (VLDB '94)*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 487–499. http://dl.acm.org/citation.cfm?id=645920.672836

[4] Adrian Bondy and U.S.R. Murty. 2008. *Graph Theory*. Springer Verlag, London, UK.

[5] Stephen P. Borgatti, Martin G. Everett, and Jeffrey C. Johnson. 2018. *Analyzing Social Networks* (2ed. ed.). SAGE Publications Ltd, Thousand Oaks, CA, US.

[6] Iztok Fister, Iztok Fister Jr, and Dušan Fister. 2019. BatMiner for Identifying the Characteristics of Athletes in Training. In *Computational Intelligence in Sports*. Springer, 201–221.

[7] Iztok Fister Jr., Xin-She Yang, Iztok Fister, Janez Brest, and Dušan Fister. 2013. A Brief Review of Nature-Inspired Algorithms for Optimization. *Elektrotehniški vestnik* 80, 3 (2013), 116–122.

[8] Jiawei Han, Jian Pei, and Yiwen Yin. 2000. Mining frequent patterns without candidate generation. In *ACM Sigmod Record*, Vol. 29. ACM, 1–12.

[9] John H. Holland. 1984. *Genetic Algorithms and Adaptation*. Springer US, Boston, MA, 317–333. https://doi.org/10.1007/978-1-4684-8941-5_21

[10] J. Kennedy and R. Eberhart. 1995. Particle swarm optimization. In *Proceedings of ICNN'95 - International Conference on Neural Networks*, Vol. 4. 1942–1948 vol.4. https://doi.org/10.1109/ICNN.1995.488968

[11] B. S. Manoj, Abhishek Chakraborty, and Rahul Singh. 2019. *Complex Networks: A Networking and Signal Processing Perspective*. Prentice Hall, Upper Saddle River, NJ.

[12] Uroš Mlakar, Milan Zorman, Iztok Fister Jr., and Iztok Fister. 2017. Modified binary cuckoo search for association rule mining. *Journal of Intelligent and Fuzzy Systems* 32 (2017), 4319–4330.

[13] Ushasi Roy, Rajdeep Kaur Grewal, and Soumen Roy. 2015. *Complex Networks and Systems Biology*. Springer Netherlands, Dordrecht, 129–150. https://doi.org/10.1007/978-94-017-9514-2_7

[14] Gregor N. Simm and Markus Reiher. 2017. Context-Driven Exploration of Complex Chemical Reaction Networks. *Journal of Chemical Theory and Computation* 13, 12 (2017), 1549–9618.

[15] Rainer Storn and Kenneth Price. 1997. Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. *Journal of Global Optimization* 11, 4 (01 Dec 1997), 341–359. https://doi.org/10.1023/A:1008202821328

[16] Xin-She Yang. 2010. *A New Metaheuristic Bat-Inspired Algorithm*. Springer Berlin Heidelberg, Berlin, Heidelberg, 65–74. https://doi.org/10.1007/978-3-642-12538-6_6

[17] S. Yong, W. Xiangming, Z. Zhenmin, and L. Yuan. 2012. Using complex network theory in the Internet engineering. In *2012 7th International Conference on Computer Science Education (ICCSE)*. 390–394. https://doi.org/10.1109/ICCSE.2012.6295099

[18] Mohammed J Zaki. 2000. Scalable algorithms for association mining. *Knowledge and Data Engineering, IEEE Transactions on* 12, 3 (2000), 372–390.