Elaboration of implicative graph according to measure M_{GK}

Bemarisika Parfait[1], Totohasina André[2]

**[1,2] Department of Mathematics and Computer Science, University of Antsiranana
Antsiranana, Madagascar**

Abstract
The implicative graph is one of the statistical tools very essential for the representation of the sequence of attribute groups. It allows the user to highlight association rules for these attributes. Several studies have already been developed, but we find that these works focus only on two measure: measure of Gras and Lerman's. While only using these two measures are not sufficient to guarantee the quality of graphs implicative. In this work, we focus on the development of the implicative graph of all association rules by using another quality measure M_{GK}. Such a construction is exponential in the size of database, and mainly due to the paths implicative step. It is therefore necessary to define an efficient algorithm to automate the construction. Following this study, we propose a new algorithm that generates an implicative graph using this new quality measure M_{GK}. We conducted experiments using one database to test the performance of our algorithm.
Keywords: *Algorithm, Implicative graph, Database, Association rules, Quality measure M_{GK}.*

1. Introduction

Introduced by the works of Gras [12] and IC Lerman [5] in the early of 1980's, the implicative statistical analysis is a fairly old discipline. The problem of successive sequences of attributes or the attribute group is frequently studied. The implicative graph is one of the statistical tools that is very essential for the representation of this sequence, particularly for taxonomical objective. It allows the user to highlight the association rules he wants to see. In this work, the relationship "arc" between the two attributes *i* and *j* will be shown in this graph implicative if the value of implicative measure is higher than a given minimum threshold. Biological sciences, Medical sciences, Computer networks [9] and Telecommunication networks are particularly the fields where the implicative graph could efficiently intervene. To our knowledge, the didactic of disciplines, the psychology and the sociology are, up to now, the main fields in which the implicative graph intervenes. As an example: Maria and Sylvia apply the implicative analysis in order to reveal the existence of hierarchy of the difficulties in finding the simple mathematical solutions [4]. Using analysis implicative, Serge and al. study the psychological behavior trait of association members for the white-collar workers, in order to build up the help system of decision-taking [13]. In [10], Ritschard and al. use implicative statistical analysis on the study of the professional dynamic in Geneva during the 19th century. In [7], Jean-Claude and J-C Réginier propose a conceptualization of implicative analysis methods to describe an activity of their students in computer data processing. In [6], Lerman and Pascal propose a formalization of a binary hierarchical oriented to provide a graphical representation of a family of association rules. In [15], the authors compare the three methods, such as the implicative statistical analysis, the factorial correspondence analysis and automatic classification of individuals. In [8], Lucia and Dusan describe the analysis of student's solution on the particular theme of conception in geometry, in order to detect the possible strategies that can be used by the students. We can note that these works are only appropriate under two implicative measures: measure of Gras [12] and the Lerman's [5], and they emphasize on the study of positive association rules. The negative association rules are thus still unknown in spite of their important interest for the user. In this work, we would like to focus on the construction of implicative graph of negative and positive rules. Such a construction is exponential in the size of database and mainly due to particularly of the paths implicative step. It is then important to define an efficient algorithm in order to get the automatic construction. In this paper, we have proposed a new algorithm generating an implicative graph under the implicative measure M_{GK} [14] (Guillaume Khenchaff's Measure). M_{GK} is varying into the interval $[-1, +1]$ [2], i.e. she is reflect the references situations such as incompatibility (M_{GK}=-1), independence (M_{GK}=0), and logical implication between the premise and the consequent of the association rules (M_{GK}=+1).

The rest of this paper is organized as follows. Some preliminaries are presented in section 2. Section 3 is devoted to the description of the method we propose. The algorithm proposed is described in section 4. Experimental evaluation will be presented in form of summary in section 5. In conclusion, we shall present our results of our contributions and research perspectives.

2. Preliminaries

In this section, we shall present all the basic concepts that will be useful later in this work. Therefore, we shall introduce the concepts of the association rules in a binary context, and directed graph.

2.1 Association rules

Definition 1. *A binary context is a triplet* $K=(T,I,R)$, *where T and I are respectively finite sets of transactions (or objects) and items (or attributes), and* $R \subseteq I \times T$ *is a binary relation between the transaction and the item. A couple* $(i,t) \in R$ *denotes the fact that the transaction* $t \in T$ *contains the item* $i \in I$.

It is assumed that the data K to explore are binary, its means we can describe each transaction by means of a finite set of items $I=\{i_1,...,i_m\}$, also called attributes. Each transaction T will be a subset of I. Furthermore, it combines each transaction identified TID (Transaction IDentify): $T=\{t_1,...,t_n\}$, that is to say $\forall (i,t) \in I \times T$, $t[i]$=1 if the item i is present in t and $t[i]$= 0 otherwise. The pattern X is a subset of items I $(X \subseteq I)$. Below, the table 1 shows an example of binary context with four items {A, B, C, D} and five transactions {1, 2, 3, 4, 5}. Let $X'=\{t \in T \mid \forall i \in X, iRt\}$ the set of all public entities to all elements X, this is the dual of the pattern X.

Table I. Example of binary context

TID	A	B	C	D
1	1	1	1	1
2	0	1	0	0
3	1	0	1	0
4	0	1	1	0
5	1	1	0	1

By considering the context of table I, for example $X=\{AB\}$, we have: $X'=\{1,5\}$ and its logical negation is presented by $\overline{X'}=\{2,3,4\}$. In the rest of this paper, to simplify the notation, let us denoted $P(Y \mid X)=P(Y' \mid X')$, where X' is the extension (or dual) of the pattern X.

Definition 2. *An association rule is a quasi implication of the form* $X \to Y$, *where X and Y are disjointed patterns* ($X,Y \subseteq I$ and $X \cap Y = \varnothing$) *respectively called the premise and the consequent of the rule.*

Example 1 Consider the association rule given by: *Computer* $\to$ *Printer*. This can be translated as customers who buy a computer also buy a printer. When considering an association rule, it is essential to associate at least one quality measure on which the user can rely on to judge its relevance. A quality measure is a function that evaluates an association rule. We present below a definition of the measure we used, measure M_{GK}.

Definition 6. *For all association rules* $X \to Y$, *the quality measure* M_{GK} *is defined as:*

$$M_{GK}(X,Y)=\begin{cases} \dfrac{p(Y \mid X)-p(Y)}{1-p(Y)}, if & p(Y \mid X) \geq p(Y) \\ \dfrac{p(Y \mid X)-p(Y)}{p(Y)}, if & p(Y \mid X) < p(Y) \end{cases} \quad (1)$$

Independently of Guillaume [14], Totohasina et al. introduced in [2] the same measure called ION (implicative measure, oriented and normalization) to allow the extraction negative and positive association rules. Wu et al. also introduced in [16] the same measurement known CPIR (Conditional Probability Increment Ratio) for the extraction of positive and negative context of data mining rules.

If $M_{GK}(X,Y)=1$, then the attraction between the premises X and therefore Y is high, that is X and Y are positively dependent. If $M_{GK}(X,Y)=0$, that is X and Y are independent, the rule $X \to Y$ is not interesting. If $M_{GK}(X,Y)=-1$, that is X and Y are stochastically incompatible, there is therefore a strong repulsion between X and Y, i.e. X and Y are negatively dependent. The main mathematical properties characterizing this quality measure are developed in [1-3]. Therefore, we urge interested reader to check their works.

2.2 Concepts of the directed graph

Definition 1. We call graph $G=(V,E)$, *the data set V whose elements are called the vertices and part E symmetrical* $\left((x,y) \in E \Leftrightarrow (y,x) \in E\right)$ *whose elements are called edges (or arcs).*

Definition 2. *A directed graph is a pair* $G=(V,E)$, *where V is a set of vertices and* $E \subset V \times V$ *is a set of edges. A directed arc* (x,y) *is denoted by* $x \to y$. *We say that* $x \to y$ *it starts at the top* x *and* y *comes from the top. In this case, each edge is oriented and cannot be driven in the direction of the arrow.*

The figure 1 is an exemplary graph of 7 vertices having order 7 *V={1, 2, 3, 4, 5, 6, 7}* and 9 edges: *E={(1,2),(1,3),(2,3), (2,4),(2,5),(2,6),(4,6),(6,7),(3,7)}*.

Definition 3. *Given a graph* $G=(V,E)$, $E \subset V \times V$. *If* $(x,y) \in E$, *then* y *is a successor of x, and if* $(y,x) \in E$, *then y is a predecessor. Two vertices connected by an edge are said to be adjacent or neighbors, i.e. x and y are adjacent if there is a predecessor and / or successor of x.*

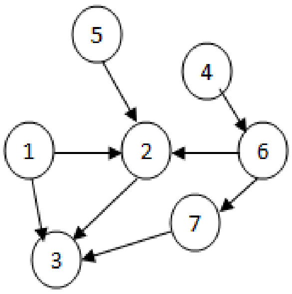

Fig. 1 Example of a graph of order 7 oriented.

In the example of figure1 , the top 2 recognizes the nodes 1, 5 and 6 as predecessors but admits the top 3 as successor, one notes respectively *pred(2) = {1, 5, 6}* and *succ(2) = {3}*. Then the peaks (1 and 2) or (2 and 3) are adjacent or immediate neighbors.

Definition 4. *Given a graph* $G=(V,E)$. *The set of successors x is noted* $\Gamma^{+}(x)$ *as* $\Gamma^{+}(x)=\{y\,|\,(x,y)\in E\}$. *The set of predecessors of x is denoted* $\Gamma^{-}(x)$ *as* $\Gamma^{+}(x)=\{y\,|\,(y,x)\in E\}$. *The set of neighbors of x is denoted* $\Gamma(x)=\Gamma^{+}(x)\cup\Gamma^{+}(x)$.

Also according to the example of figure 1, all of the successor peak 1 is given by $\Gamma^{+}(1)=\{2,3\}$, while all of the predecessor vertex is empty as $\Gamma^{-}(1)=\{\}$. Then the set of neighbors of the vertex 1 is $\Gamma(1)=\{2,3\}$.

Definition 5. *Given a graph* $G=(V,E)$. *Called interior grade or half-degree of a vertex x is the number of elements denoted* $d^{-}(x)=|\Gamma^{-}(x)|$.

In the example of figure 1, there is no peak that is within the top one, its degree is zero: $d^{-}(1)=0$, while the degree within the top 3 is at number 3 , so $d^{-}(3)=3$.

Definition 6. *Given a graph* $G=(V,E)$. *Called out-degree (external degree) or half-degree of a vertex x is the number of elements noted* $d^{+}(x)=|\Gamma^{+}(x)|$.

In figure 1, there are two vertices which are beyond the top one, so $d^{+}(1)=2$.

Definition 7. *Given a graph* $G=(V,E)$. *We call the degree of x (or valence) the sum of the in-degree and out-degree* $d(x)=d^{-}(x)+d^{+}(x)$.

In our example in figure 1, the degree of vertex 1 is none other than the out-coming degree because its incoming degree is zero, so $d(1)=2$. A vertex of degree zero incoming and outgoing non-zero degree is called source, while a vertex of degree entering non-zero and zero is called out-degree wells. In this example, vertices 1, 4 and 5 are source vertices, while the top 3 is well known.

Definition 8. *A graph is valued when, each edge or arc is associated with a number real. If this number is positive, it is called weight and weighted graph.*

In a graph, it is natural to want to move from peak to peak along the edges. Such a step is called a string or path.

Definition 9. A path of graph $G=(V,E)$ *is a sequence of vertices* $x_1,\ldots,x_k,\forall k\geq 1,\forall i:1\leq i\leq k-1,(x_i,x_{i+1})\in E$. *A path* $[x_1,\ldots,x_k]$ *is called elementary if* $x_1,\ldots,x_k,\forall k\geq 1$ *and* $\forall i, x_i\neq x_{i+1}$, *i.e. contains not twice the same vertex.*

The length of a path is defined by the number arcs composing the path. It equals the number of vertices minus one: $|V|-1$. A path consisting of only the top x_1 noted $[x_1]$ is of zero length. In the example of figure 1, the path [4, 6, 7, 3] is an elementary way, for each vertex of the path is visited once.

Definition 10. *A circuit of* $G=(V,E)$ *is one path* $[x_1,\ldots,x_k]$ *such that* $x_1=x_k$.

The example of figure 1 is a graph without circuit, because there are not an edge returns between the peaks.

3. Description of the method

In this section, we describe our approach to build a graph called implicative by using the measure of involvement M_{GK}. In most cases, we tried to explore large databases for this purpose, the construction of such implicative graph has become tedious and exhausting, mainly because of the problem of course implicative way. It is therefore necessary to define an efficient algorithm that can automate the construction. The approach we have adopted is as follows. Formalize the data measuring M_{GK} involvement in an adjacency matrix or Boolean matrix to switch to binary mode involvement between associations, such as:

$$M=\begin{cases}1, & \text{if}\quad M_{GK}(i,j)\geq \min M_{GK}\\ 0 & \text{otherwise}\end{cases} \tag{1}$$

where *i* and *j* two adjacent items, $M_{GK}(i,j)$ represents the value of the arc $i\rightarrow j$ according to the measure M_{GK}, and minM_{GK} is the minimum threshold set by the expert. In other words, we assign the value 1 if the measure M_{GK} verifies the minimum threshold, and 0 otherwise. Interest is finding the path of implicative path all vertices. After formalized data, we generated the set of vertices (going through the implicative path), adapting the situation to the technique of the search path theory of classical graphs, in which the transition cost function becomes, in our approach, the involvement measure M_{GK}. The adapted new technique is defined as follows. Given a set of items (or vertices) V and a set of arcs E, the generating implicative graph $G_{imp} = (V,E)$ such that:

$$G_{imp} = \{\forall j \in V \setminus \Gamma^{+}(i) : \delta(j) = \min\{\delta(j); \delta(i) + M_{GK}(i,j)\}\} \quad (2)$$

where j is the natural successor to i; $\Gamma^{+}(i)$ is the set of successors of i, and $\delta(j)$ is the minimum weight of at least j. As shown in *Algorithm* 1, the process is repeated until the implicative graph is generated in step $|V|-1$. In the beginning, the parameters of the oriented graph are obtained by *Lemma* 1 below.

Lemma 1. *If an item j is adjacent to the immediate source item* i_0*, then the minimum weight j coincides with the value of the implicative measure* M_{GK} *of* i_0 *to j as:*

$$\delta(j) = M_{GK}(i_0, j) \quad (4)$$

When all the nodes of the implicative graph are integrated in the set *V*, the lists represent the length of the path from the source item i_0 to all other vertices.

Definition 12. *An implicative graph* $G=(V,E)$ *is a directed graph, weighted and not reflexive, where V is the set of items of context, and E is the set of arcs in pairs of items. The arc x y represents the rule if x item is chosen, so y is probably chosen too.*

The weight carried on such arc (x, y) is defined by the value the implicative measure M_{GK}. The graph in figure 1 is an oriented graph and not reflexive and weighted.

Proposition 1. *An implicative graph* $G=(V,E)$ *is necessarily without circuit.*

Proof 1. By the absurd. Taking an implicative measure M_{GK} is (transitive and not reflexive) on a finite set *V*. If $c=[x_1,...,x_k]$ is a elementary circuit, we deduce by transitivity of M_{GK}, we have:

$M_{GK}(x_1,x_2), M_{GK}(x_2,x_3),..., M_{GK}(x_{k-1},x_k) = M_{GK}(x_1,x_k)$.

Where, $M_{GK}(x_1,x_1)$ since $x_1 = x_k$ Contradiction!! Because M_{GK} is not reflexive.

Proposition 2. *An implicative graph* $G=(V,E)$ *has at least one source and a sink.*

Proof 2. Consider a path c which is maximal in the following sense: $c=[x_1,...,x_k]$ and there is no apex *y* in *G* as $[y,x_1,...,x_k]$ or $[x_1,...,x_k,y]$ as paths of *G*. Such a path exists since *G* is without circuit. This means that x_1 is a source, and x_k is a well. Well lead to our model, we take an experiment using a small base in the below table 1.

Table 1. Example of association rule valid under M_{GK}

$d \to f(0.15)$	$b \to d(0.10)$
$b \to e(0.25)$	$d \to g(0.30)$
$d \to e(0.20)$	$c \to f(0.20)$
$e \to g(0.10)$	$a \to c(0.20)$
$f \to g(0.20)$	$c \to d(0.15)$
$a \to b(0.15)$	

As a result, we formalize this data as a Boolean matrix to find the items sources and the last item of the graph. Using the relation (1), the adjacency obtained matrix is in table 2 below.

Table 2. Adjacency matrix of the data M_{GK}

	a	b	c	d	e	f	g
a	0	1	1	0	0	0	0
b	0	0	0	1	1	0	0
c	0	0	0	1	0	1	0
d	0	0	0	0	1	1	1
e	0	0	0	0	0	0	1
f	0	0	0	0	0	0	1
g	0	0	0	0	0	0	0

As the column of the item *a* and a line item *g* are all zero (see table 2), then according to proposition 2, there is no *h* of *G* such as [*h, a, b, c, d, e, f, g*] or [*a, b, c, d, e, f, g, h*] as the paths *G*. In other words, neither item is predecessor of *a* and nor successor of *g*. This shows that a is *a* source and g is the last item. Therefore, the construction of that graph begins with item a. The calculation steps are carried out as follows. To do this, denote by *V* the set of vertices and permanently marked by *p(j)* the predecessor of item j.
In the 0 step, the set *V* naturally contains a single item because a is a source, therefore *V*={a}=a. Then, based on the adjacency matrix of table 2, we find that *b* and *c* are immediate adjacent to the source item, then by *Lemma 1*, we have: $\delta(b)=0.15$ and $\delta(c)=0.20$. We also see that the items *d*, *e*, *f* and *g* are not adjacent to *a* immediate, so they temporarily take a weight "∞" and a predecessor line "$-$". The result is summarized in table 3 below.

Table 3. Item parameters at step 0

step	V	$\delta(b)\,p(b)$	$\delta(c)\,p(c)$	$\delta(d)\,p(d)$	$\delta(e)\,p(e)$	$\delta(f)\,p(f)$	$\delta(g)\,p(g)$
0	a	0.15,a	0.20,a	∞,–	∞,–	∞,–	∞,–

In step1, we see that $\delta(b)=0.15$ which is the minimum weight of all the peaks, which will *b* integrate the list of *V* and we get $V=\{ab\}=ab$. Then from table 2 the adjacency matrix, we see that d and e are in immediate adjacent items *b*, we have:

$\delta(d)=\min\{\delta(d);\delta(b)+M_{GK}(b,d)\}=\min\{\infty;0.15+0.10\}=0.25$

and

$\delta(e)=\min\{\delta(e);\delta(b)+M_{GK}(b,e)\}=\min\{\infty;0.15+0.25\}=0.40$

Initially, the predecessors of these two items are not known, they will now argue *b*. While items *f* and *g* are not adjacent to *b* immediate, they keep their old settings ("∞" and *a* dash "$-$"). The updated parameters are summarized in table 4.

Table 4. Item parameters at step 1

step	V	$\delta(b)p(b)$	$\delta(c)p(c)$	$\delta(d)p(d)$	$\delta(e)p(e)$	$\delta(f)p(f)$	$\delta(g)p(g)$
0	a	0.15,a	0.20,a	∞,–	∞,–	∞,–	∞,–
1	ab		0.20,a	0.25,b	0.40,b	∞,–	∞,–

In step 2, from table 4 it is seen that: $\delta(c)=0.20$ which is the minimum weight of all vertices. We will then move c in the set *V*, and we have $V=\{abc\}$. From table 2 of the adjacency matrix, we see that *d* and *f* are immediate adjacent to *c*. For this purpose, was

$\delta(d)=\min\{\delta(d);\delta(c)+M_{GK}(c,d)\}=\min\{0.25;0.2+0.15\}=0.25$

We have the same value as before, this means that the two solutions (0.25, b) and (0.25, c) are possible, but we will choose b arbitrarily for further calculations. For f, we have:

$\delta(f)=\min\{\delta(f);\delta(c)+M_{GK}(c,f)\}=\min\{\infty;0.20+0.20\}=0.40$.

The immediate predecessor of *f* is *c*. The result is in table 5.

Table 5. Item parameters at step 2

step	V	$\delta(b)p(b)$	$\delta(c)p(c)$	$\delta(d)p(d)$	$\delta(e)p(e)$	$\delta(f)p(f)$	$\delta(g)p(g)$
0	a	0.15,a	0.20,a	∞,–	∞,–	∞,–	∞,–
1	ab		0.20,a	0.25,b	0.40,b	∞,–	∞,–
2	abc			0.25,bc	0.40,b	0.40,c	∞,–

In step 3, table 5 shows that the minimum weight of all the vertices is $\delta(d)=0.25$, which is going to include in the list of the set *V*, and $V=\{abcd\}$. From table 2, its immediate adjacent vertices are e, f and g, we have:

$\delta(e)=\min\{\delta(e);\delta(d)+M_{GK}(d,e)\}=\min\{0.40;0.25+0.20\}=0.40$

So we have the same value as before, the two solutions (0.40, b) and (0.40, d) are possible, but we will keep (0.40,b) in the following calculations. On the other hand,

$\delta(f)=\min\{\delta(f);\delta(d)+M_{GK}(d,f)\}=\min\{0.40;0.25+0.15\}=0.40$

We also have the same value as before, so the two solutions (0.40, c) and (0.40, d) are also possible, but we will keep (0.40, c). And $\delta(e)=\delta(f)=0.40$, so we arbitrarily choose between e and f, and we have chosen e. For g, we have:

$\delta(g)=\min\{\delta(g);\delta(d)+M_{GK}(d,g)\}=\min\{\infty;0.25+0.30\}=0.55$

The immediate predecessor of g is initially unknown, it will now argue d. The results are in table 6.

Table 6. Item parameters at step 3

step	V	$\delta(b)p(b)$	$\delta(c)p(c)$	$\delta(d)p(d)$	$\delta(e)p(e)$	$\delta(f)p(f)$	$\delta(g)p(g)$
0	a	0.15,a	0.20,a	∞,–	∞,–	∞,–	∞,–
1	ab		0.20,a	0.25,b	0.40,b	∞,–	∞,–
2	abc			0.25,bc	0.40,b	0.40,c	∞,–
3	abcd				0.40,bd	0.40,cd	0.55,d

In step 4, from table 6 was $\delta(e)=0.40$ which is the minimum distance, there will therefore integrate e in the set S and we get $V=\{abcde\}$. According to the adjacency matrix of table 2, only the adjacent item is immediate to e g. So,

$\delta(g)=\min\{\delta(g);\delta(e)+M_{GK}(e,g)\}=\min\{0.55;0.4+0.1\}=0.5\neq 0.55$

We have a change of values. So the immediate predecessor of *g* becomes *e*. The result of calculations is in table 7.

Table 7. Item parameters at step 4

step	V	$\delta(b)p(b)$	$\delta(c)p(c)$	$\delta(d)p(d)$	$\delta(e)p(e)$	$\delta(f)p(f)$	$\delta(g)p(g)$
0	a	0.15,a	0.20,a	∞,–	∞,–	∞,–	∞,–
1	ab		0.20,a	0.25,b	0.40,b	∞,–	∞,–
2	abc			0.25,bc	0.40,b	0.40,c	∞,–
3	abcd				0.40,bd	0.40,cd	0.55,d
4	abcde					0.40,c	0.50,e

In step 5, from table 7, the minimum distance of all vertices is $\delta(f)=0.40$, so we will include f in S and we get $V=\{abcdef\}$. Then g is the only immediate adjacent vertex f (see table 2). So,

$\delta(g)=\min\{\delta(g);\delta(f)+M_{GK}(f,g)\}=\min\{0.50;0.40+0.20\}=0.50$

We have the same value as before, the two solutions (0.50,e) and (0.50,f) are possible, but we kept arbitrarily (0.50,e). The result is summarized in table 8 below.

Table 8. Item parameters at step 5

step	V	$\delta(b)p(b)$	$\delta(c)p(c)$	$\delta(d)p(d)$	$\delta(e)p(e)$	$\delta(f)p(f)$	$\delta(g)p(g)$
0	a	0.15,a	0.20,a	∞,–	∞,–	∞,–	∞,–
1	ab		0.20,a	0.25,b	0.40,b	∞,–	∞,–
2	abc			0.25,bc	0.40,b	0.40,c	∞,–
3	abcd				0.40,bd	0.40,cd	0.55,d
4	abcde					0.40,c	0.50,e
5	abcdef						0.50,ef

In step 6, it is obvious that the minimum distance is $\delta(d)=0.50$, because it is the only remotely compare. We will integrate *g* in the list of the set *V*, and we have: $V=\{abcdefg\}$. This is the end of step calculations, because no top is to go.

Table 9. Item parameters at step 6

step	V	$\delta(b)p(b)$	$\delta(c)p(c)$	$\delta(d)p(d)$	$\delta(e)p(e)$	$\delta(f)p(f)$	$\delta(g)p(g)$
0	a	0.15,a	0.20,a	∞,–	∞,–	∞,–	∞,–
1	ab		0.20,a	0.25,b	0.40,b	∞,–	∞,–
2	abc			0.25,bc	0.40,b	0.40,c	∞,–
3	abcd				0.40,bd	0.40,cd	0.55,d
4	abcde					0.40,c	0.50,e
5	abcdef						0.50,ef
6	abcdefg						0.50,e

IJCSI International Journal of Computer Science Issues, Vol. 11, Issue 4, No 1, July 2014
ISSN (Print): 1694-0814 | ISSN (Online): 1694-0784
www.IJCSI.org

From table 9 (summary of all stages of calculations), the implicative graph our method is built in this way. Item from the source, draw arcs from *a* to *b*, and *c* to the respective distances 0.15 and 0.20. Then from item b, draw arcs from b to d and b to c respective distances 0.25 and 0.40. From item c, draw arcs c to d and f to c respective distances 0.25 and 0.40. From item d, draw arcs d to f, and d to e and d to g, respective distances 0.40, and 0.40 and 0.55. From item e, draw the arc e to g, 0.50 distances. Finally, draw arc f to g, 0.50 distances. The implicative graph matching is as follows (see figure 2).

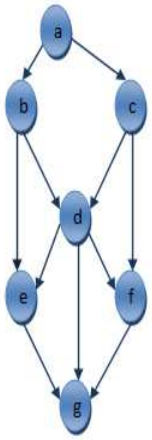

Fig. 2 : Implicative graph of our method

As we have indicated, the manual construction of a graph implicative is revealed to be difficult when we have a large database (large association rules) mainly due to the different steps of implicative paths. It is therefore necessary to define an effective algorithm in order to automate the construction. The following section presents an algorithm that we have proposed.

4. Presentation of the algorithm

In this section, we shall present a proposal of an algorithm generating a problem implicative graph according to implicative measure M_{GK} . The implementation of this problem is described in the pseudo code of the algorithm 1 and 2 below. The first algorithm is called graph construction implicative (CGI) which has an objective of elaborating an implicative graph of association rules. It receives the data of the M_{GK}, the minimum threshold $minM_{GK}$ and all items *V*. The algorithm generates the CGI implicative graph with a single source; however the second algorithm, called algorithm for constructing global implicative graph (GIG), generates all other existing sources items.

Algorithm 1 Construction of implicative graph (CGI)

input : Matrix M_{GK} ; $min_{M_{GK}}$; i_0 item source ; V set of vertices.
output: $G_{imp} = (V, E)$: Implicative graph.

1 **begin**
2 **for all** $item\ i \in V$ **do**
3 **for all** $item\ j \in V$ **do**
4 **if** $M_{GK}(i,j) \geq min_{M_{GK}}$ **then**
5 $M(i,j) \leftarrow 1$ **else** $M(i,j) \leftarrow 0$;
6 **end**
7 **end**
8 **end**
9 $G_{imp} \leftarrow \emptyset$; $SAI \leftarrow SomAdjIm(i_0)$;
10 **for all** $j \in SAI$ **do**
11 $\delta(j) \leftarrow M_{GK}(i_0, j)$;
12 **end**
13 **for all** $j \in (V \setminus SAI)$ **do**
14 $\delta(j) \leftarrow \infty$;
15 **end**
16 $i \leftarrow i_0$; $arret \leftarrow 0$;
17 **repeat**
18 $SAI \leftarrow SomAdjIm(i)$;
19 **for all** $j \in SAI$ **do**
20 $\delta(j) \leftarrow min\{\delta(j); \delta(i) + M_{GK}(i,j)\}$;
21 $G_{imp} \leftarrow G_{imp} \cup (i,j)$;
22 **end**
23 $min \leftarrow \delta(V(1))$;
24 **for** $k \leftarrow 2$ **to** $|V|$ **do**
25 $\delta 1 \leftarrow \delta(V(k))$;
26 **if** $\delta 1 < min$ **then**
27 $min \leftarrow \delta 1$; $i \leftarrow V(k)$;
28 **end**
29 **end**
30 $arret + +$;
31 **until** $(arret = |V| - 1)$;
32 **end**

Firstly, the algorithm formalizes the data M_{GK} to the adjacency matrix (lines 2 to 8). The algorithm CGI begins with finding the items sources because of *SomAdjIm* function (line 9). Beginning with the source item, the algorithm will generate the natural successor of item source and found (lines 10 to 15). As a result, the algorithm determines the parameters of these successor peaks to put them in order in ascending order (lines 10 to 12). It then performs the update of all other vertices that are not adjacent to the immediate source item (lines 13 to 15). Then comes the phase of construction of other new natural successor (lines 16 to 31), thanks to the *SomAdjIm* function (line 18). In this case, the algorithm starts by including a new item with the minimal weight, then for the next step, it generates all the new vertices adjacent immediate newly included in this item. This procedure is repeated until the implicative graph is extracted in step $|V|-1$ (lines 17 to 31). The algorithm will stop when all the peaks are placed (line 31). The use of the other sources is presented in Algorithm 2 below. The overall algorithm implicative graph (GIG) receives as input a set of items sources. It generates the implicative graphs from all sources (lines 1 to 7) through the *Gimp* function (line 4).

IJCSI International Journal of Computer Science Issues, Vol. 11, Issue 4, No 1, July 2014
ISSN (Print): 1694-0814 | ISSN (Online): 1694-0784
www.IJCSI.org


```
Algorithm 2 Generation global implicative graph(GIG)
input  : V set of items sources; V set of item; Matrix M_GK ; min_M_GK
output: GG = (V,E) : Global Implicative Graphe
begin
    GG ← ∅;
    for all source s ∈ V do
        G_imp(Matrix M_GK, min_M_GK, s, V);
        GG ← GG ∪ G_imp
    end
end
```

4. Experimentation

The objective of this section is to test the feasibility of our approach. We evaluate our algorithm in terms of generation of implicative graph. We have implemented our algorithm in R language and C++, inspired from the work of R. Couturier (CHIC) [11].

Our experiments were performed on a PC with 4 GB of RAM under Windows system. In this context, we have conducted a series of experiments on a synthetic database “animaux” [11]. This database includes 71 items and 41 variables describing the characteristics of animals. Our experiments were performed on a PC with 4 GB of RAM under Windows system. In this context, we have conducted a series of experiments of a synthetic database.
The main objective of this experiment is to observe the quality of implicative graph obtained when the minimal threshold varies $\min M_{GK}$. As a result, we varied the minimal $\min M_{GK}$ à 90%, 95% and 97%, the obtained results found in figures 3, 4 and 5. As we have mentioned the calculation of the implicative graph of our method provides a graph on which the variables that have a value greater than or equal to a certain threshold implication, are connected with an arrow. Figure 3 below shows an implicative graph with a 97% threshold. The figure 3 shows a graph of implicative six sources (“Grand”, “Peureux”, “Comique”, “Sale”, “Tetu”, and “Stupide”) and four wells (“Beau”, “Discret”, “Sournois”, and “Bete”).

With a threshold of 95%, we have 18 most frequent transitions. To interpret it, we rely on the concept of implicative measure M_{GK} and visualize in knots transitions statistically significant involvement. Specifically, the involvement is considered that of the rule which is defined by the premise leading to the branch node and entering the transition in question. Transitions are represented and most characteristic of the node in the sense that they are the ones whose relative frequency of against-examples within the group is the significantly lower than the proportion of other transitions in the node initial. In each node transitions meaningful involvement are ordered from bottom to top in order to decrease the significance.

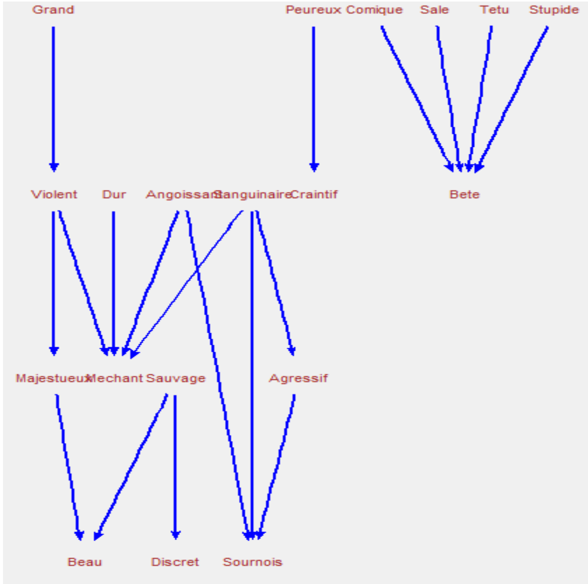

Fig. 3: Implicative graph tested our algorithm ($\min M_{GK} = 97\%$)

With this new tool, we can present the implicative graph on the same figure. Figure 4 presents of implicative graph of threshold 95% (green) and 97% (red). While figure 5 as 90% threshold (green) and 97% (red).

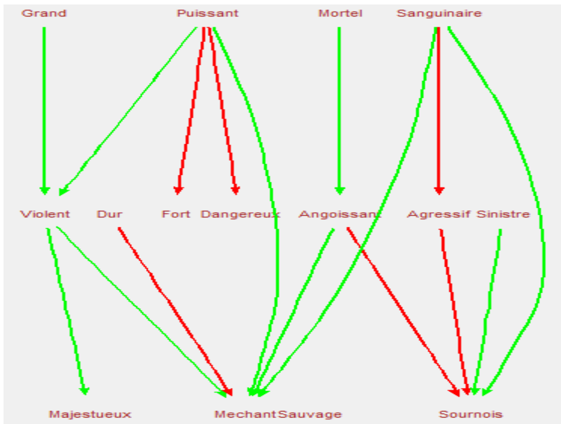

Fig. 4: Implicative graph tested our algorithm ($\min M_{GK} = 95\%, 97\%$)

As we have noted, the implicative graph allows the user to highlight the important features of the data. As indicated in the given examples, we can interpret the Figure 4 as follows. With 95%, a Grand admits animal characters and violent. It is majestic and wicked. At 95% of chance, an animal “Powerful” is a violent and wicked animal. It is 97% strong and hazardous animal. At 95%, the animal “Mortal” is scary and mean, but 97% sneaky. An animal "Bloody" is 95%, mean and nasty. At 97%, it is aggressive and sneaky. Character both sinister and cunning is made 95% chance. The figure 5 below shows the feasibility of our approach that provides the set of graphs to several paths implicative step in very reasonable time. As we have seen in Figure 5, we observe that when the threshold is reduced minG_{GK} (90%), we have implicative graphs larger due the increase in the size of rules of significant associations. We can say that our approach is operational even if the size is big bass.

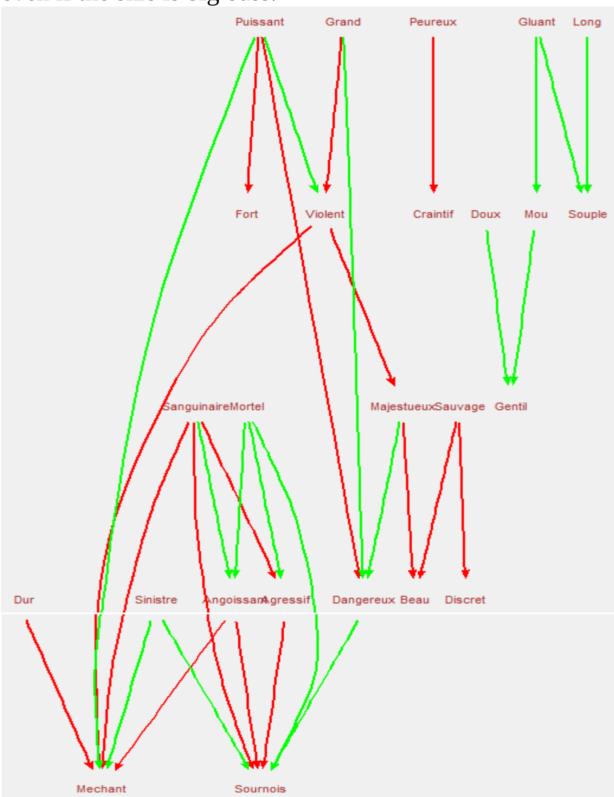

Fig. 5: Implicative graph tested our algorithm ($\min M_{GK} = 90\%, 97\%$)

4. Conclusions and future work

We have proposed a new method to construct the implicative graph in the sense of implicative measure M_{GK}. Also, we have presented a new algorithm that generates these implicative graphs. The results of our experiments show the effectiveness of the proposed strategy to generate graphs in several ways, in times in a very reasonable answer. In the near future, extending our algorithm to make the construction the implicative graph of implicative quantitative and qualitative association rules.

Acknowledgments

This work has been realized in the Laboratory of Mathematics and Computer Science from the "Normal School for Technical Education (ENSET)", University of Antsiranana (Madagascar), with Laboratory of Computer Science and Mathematics (LIM), University of the Reunion, France. We acknowledge the AUF (University Agency of the Francophone), for his financial support.

References

[1] A. Totohasina, "Contribution to the study of measures of quality of association rules: normalization and constraints in five cases and M_{GK}, properties, composite base and extension rules for applying statistical and physical sciences", University of Antsiranana, Madagascar, PhD thesis, 2008.

[2] A. Totohasina, H. Ralambondrainy and J. Diatta, "A unifying vision of quality measures Boolean association rules and an efficient algorithm for extracting association rules implicative", *University of Antsiranana, Madagascar, 2005.*

[3] D. R. Feno, "Measuring quality of association rules: normalization and characterization of bases", University of the Reunion, France. PhD thesis, 2007.

[4] G. O Maria Sylvia and Z. "Statistical involvement and learning", University of Roma, 2000.

[5] I.C Lerman, "Classification and ordinal data analysis", Dunod, 1981.

[6] I.C Lerman and Pascal, "Directed binary hierarchies and directed ultrametrics", University of Rennes 1, 2009.

[7] Jean-Claude and J-C. Réginier, "Conceptualizing the implicative statistical analysis", University Lyon 2, 2009.

[8] Lucia. R and V. Dusan. The evaluation of a geometric problem of the application of the statistical analysis implicative. *International Conference*, ASI, 2012.

[9] Pushpa C. N, Thriveni J, Venugopal K. R and L. M. Patnaik, "Modified Pattern Extraction Algorithm for Efficient Semantic Similarity Measures between Words" University Visvesvaraya College of Engineering, IJCSI, Vol. 10, September 2013 Bangalore, Karnataka, India

[10] Ritschard and S. Matthias and O. Michel, "Applications sociology", University of Geneva, 2009.

[11] R. Couturier, "Processing of statistical analysis and CHIC",University of Franche-Comté, 2000.

[12] R. Gras,"Contribution to the experimental study and analysis of certain cognitive acquisitions and certain learning objectives in mathematics", University of Rennes I, PhD thesis, 1979.

[13] Serge Baquedano, Vincent Philippé, Jacques Philippé, Régis Gras, and Olivier Guilhaume. The implicative analysis for the development of behavioral standards. Third International Meeting, ASI, 2005.

[14] S. Guillaume,"Processing large amounts of data. Measure and algorithms mining association rules and ordinal rules", University Nantes, France, PhD thesis, 2000.

[15] S. Matthias and G. Ritschard, "A implicative statistical analysis a text mining", University of Geneva.

[16] X. Wu, C. Zhang and S. Shang, "Mining booth Positive and Negative Association Rules", *ICML-2002 & ACM TOIS 2004*, University of Vermont.

First Author: Bemarisika Parfait, holder of Master 2 in Mathematics and Applications, University of Toulouse I, France (2010). Currently, PhD student in Mathematics and Computer Science, under supervision of Professor Totohasina André and Professor Jean Diatta.

Second Author: Totohasina André, Professor at the University of Antsiranana, Madagascar. Specialist in Mathematics and Applications, Probability and Statistics in particular. Director of Doctoral formation Mathematics and Computer Science and research didactic, said University.