A Rough Sets Based Generalization of Functional Dependencies

Laurentiu B. Cristofor

May 8, 2000

1 Introduction

Functional dependencies occupy an important place in the theory of relational databases. In this article we present a generalization of functional dependencies based on rough sets theory and we demonstrate some interesting properties these dependencies have.

2 Rough set based dependencies

In this section we show how we can generalize the notion of functional dependency based on rough sets concepts as presented in [4]. We will first present some formal definitions, then we will prove some interesting properties, and finally we will go through some examples.

2.1 Definitions

Let A be a set of attributes $\{A_1, A_2, \ldots, A_n\}$ and let T be a set of tuples such that $T \subseteq \mathrm{Dom}(A_1) \times \mathrm{Dom}(A_2) \times \ldots \times \mathrm{Dom}(A_n)$. We will use the notation $\mathrm{Dom}(A)$ for $\mathrm{Dom}(A_1) \times \mathrm{Dom}(A_2) \times \ldots \times \mathrm{Dom}(A_n)$.

We will use letters X, Y, Z to refer to sets of attributes.

Definition 1 *The equivalence relation RE(T,X), determined by a set of attributes $X \subseteq A$ on the set of tuples T, is defined by:*

$$(t_1, t_2) \in RE(T, X) \quad \Leftrightarrow \quad t_1[X] = t_2[X] \quad \forall\, t_1, t_2 \in T$$

The partition PART(T,X), determined by a set of attributes $X \subseteq A$ on the set of tuples T, is defined by: PART(T,X) = T/RE(T,X).

□

When there is no risk of confusion, we will use the notation $RE(X)$ instead of $RE(T, X)$ and $PART(X)$ instead of $PART(T, X)$. The letters U, V, W will designate blocks belonging to a partition.

Definition 2 *The positive space [4] determined by a set of attributes $X \subseteq A$ with respect to a set of attributes $Y \subseteq A$ on the set of tuples T, is defined by:*

$$POS(T, X, Y) = \bigcup \left\{ U \in PART(X) \mid U \subseteq V, \, for\, some\, V \in PART(Y) \right\}$$

$\square$

The notation $POS(X, Y)$ will be used as a shorthand for $POS(T, X, Y)$.

Definition 3 *We say that the set of attributes Y depends on degree $k \in [0, 1]$ on the set of attributes X and we write $X \xrightarrow{k} Y$ where k is defined as follows:*

$$k = \frac{card(POS(X, Y))}{card(T)}$$

$\square$

For $X \xrightarrow{0} Y$ we will simply write $X \not\to Y$ and for $X \xrightarrow{1} Y$ we will simply write $X \to Y$.

We can look at these dependencies like being a generalized form of the functional dependencies in a relational database since $X \xrightarrow{1} Y$ is the equivalent of a standard functional dependency.

Lemma 1 *$X \xrightarrow{1} Y$ is equivalent to the functional dependency $X \to Y$.*

$\square$

Proof: It results directly from the definition.

$\square$

2.2 Properties

In the following theorems and corollaries we will prove some of the rules respected by the above defined dependencies.

Theorem 1 *$POS(X, Y) \subseteq POS(XZ, YZ)$, for $X, Y, Z \subseteq A$.*

$\square$

Proof: Suppose $t \in POS(X, Y)$.

Then there exists $U \in PART(X)$ and $V \in PART(Y)$ such that $t \in U$ and $U \subseteq V$. This means that $t \in V$.

We therefore have: $\forall t_k \in U : t_k[X] = t[X] = X_1 \in \mathrm{Dom}(X)$ and: $\forall t_k \in U : t_k[Y] = t[Y] = Y_1 \in \mathrm{Dom}(Y)$.

Also $\forall t_l \in T \backslash U : t_l[X] \neq t[X]$.

Considering $PART(XZ)$, we will have: $U = \bigcup U_i$ where $U_i \in PART(XZ)$ and $t_m[X] = X_1$, $\forall t_m \in U_i$.

Then, $t \in U_j$ for some j and we will have: $\forall t_k \in U_j : t_k[XZ] = t[XZ] = X_1 Z_j$ where $Z_j \in \mathrm{Dom}(Z)$.

It follows that for all $t_k \in U_j$: $t_k[Z] = t[Z] = Z_j$ and $t_k[Y] = t[Y] = Y_1$, so $t_k[YZ] = t[YZ] = Y_1 Z_j$. So for all $t_k \in U_j$ we also have $t_k \in V_j$ where $V_j \in PART(YZ)$ and $t_l[YZ] = Y_1 Z_j$ for all $t_l \in V_j$.

We have $t \in U_j \subseteq V_j$, where $U_j \in PART(XZ)$ and $V_j \in PART(YZ)$ so $t \in POS(XZ, YZ)$. QED.

$\square$

From this result we easily derive the following:

Corollary 1 $X \overset{k}{\to} Y \Rightarrow XZ \overset{k'}{\to} YZ, \quad k' \geq k$

$\square$

Proof: From Theorem 1 we know that $POS(X,Y) \subseteq POS(XZ,YZ)$, for $X, Y, Z \subseteq A$. Therefore $\mathrm{card}(POS(X,Y)) \leq \mathrm{card}(POS(XZ,YZ))$, so $\frac{\mathrm{card}(POS(X,Y))}{\mathrm{card}(T)} \leq \frac{\mathrm{card}(POS(XZ,YZ))}{\mathrm{card}(T)}$, so $k \leq k'$.

$\square$

Theorem 2 *If $POS(X,Y) = T$ then $POS(Y,Z) \subseteq POS(X,Z)$, for $X, Y, Z \subseteq A$.*

$\square$

Proof: Suppose $t \in POS(Y,Z)$.

Then there exists $V \in PART(Y)$ and $W \in PART(Z)$ such that $t \in V$ and $V \subseteq W$.

$t \in T$ so $t \in POS(X,Y)$ so there exists $U \in PART(X)$ such that $t \in U$ and $U \subseteq V$.

So we have $t \in U$ and $U \subseteq V \subseteq W$ where $U \in PART(X)$ and $W \in PART(Z)$. So $t \in POS(X,Z)$. QED.

$\square$

From this result we easily derive the following:

Corollary 2 $X \to Y, Y \overset{k}{\to} Z \Rightarrow X \overset{k'}{\to} Z, \quad k' \geq k$

$\square$

Proposition 1 *If $POS(X,Y) = T$ then for any $U \in PART(X)$ there exists $V \in PART(Y)$ such that $U \subseteq V$.*

$\square$

Proof: Suppose $U \in PART(X)$. Let's take an element $t \in U$. Then, since $t \in T$, it follows that $t \in POS(X, Y)$ so there exists $V \in PART(Y)$ such that $U \subseteq V$. QED.

□

Theorem 3 *If $POS(Y, Z) = T$ then $POS(X, Y) \subseteq POS(X, Z)$, for $X, Y, Z \subseteq A$.*

□

Proof: Suppose $t \in POS(X, Y)$.

Then there exists $U \in PART(X)$ and $V \in PART(Y)$ such that $t \in U$ and $U \subseteq V$. Since $POS(Y, Z) = T$ and $V \in PART(Y)$, using Proposition 1 it follows that there exists $W \in PART(Z)$ such that $V \subseteq W$.

We have $t \in U$ and $U \subseteq V \subseteq W$ where $U \in PART(X)$ and $W \in PART(Z)$ so $t \in POS(X, Z)$. QED.

□

From this result we easily derive the following:

Corollary 3 $X \xrightarrow{k} Y, Y \rightarrow Z \Rightarrow X \xrightarrow{k'} Z, \quad k' \geq k$

□

From Corollary 3 and by using the first Armstrong rule ($YZ \rightarrow Y$), we also get:

Corollary 4 $X \xrightarrow{k} YZ \Rightarrow X \xrightarrow{k'} Y, \quad k' \geq k$

□

2.3 Examples

Example 1 Consider the following set of tuples T:

	A	B	C
1	0	0	1
2	0	1	1
3	0	2	1
4	1	2	0
5	2	0	0
6	2	0	2
7	1	1	2

We determine k for $A \xrightarrow{k} C$

We have $PART(C) = \{\{1,2,3\},\{4,5\},\{6,7\}\}$ and $PART(A) = \{\{1,2,3\},\{4,7\},\{5,6\}\}$, so $A \xrightarrow{\frac{3}{7}} C$.

We determine k for $AB \xrightarrow{k} BC$
We have $PART(BC) = \{\{1\},\{2\},\{3\},\{4\},\{5\},\{6\},\{7\}\}$ and $PART(AB) = \{\{1\},\{2\},\{3\},\{4\},\{5,6\},\{7\}\}$, so $AB \xrightarrow{\frac{5}{7}} BC$.

We determine k for $A \xrightarrow{k} B$
We have $PART(B) = \{\{1,5,6\},\{3,4\},\{2,7\}\}$ and $PART(A) = \{\{1,2,3\},\{4,7\},\{5,6\}\}$, so $A \xrightarrow{\frac{2}{7}} B$.

We determine k for $B \xrightarrow{k} C$
We have $PART(C) = \{\{1,2,3\},\{4,5\},\{6,7\}\}$ and $PART(B) = \{\{1,5,6\},\{3,4\},\{2,7\}\}$, so $B \xrightarrow{0} C$.

This example is an exemplification of Corollary 1 and is also illustrating an interesting case: we can have $A \xrightarrow{k} B$, $B \xrightarrow{0} C$ and $A \xrightarrow{k'} C$ with $k' > k > 0$.

Example 2 Consider the following set of tuples T:

	A	B	C
1	0	1	2
2	0	1	0
3	1	0	2
4	1	2	1
5	1	0	0
6	2	0	2
7	2	0	0
8	2	2	1

We determine k for $A \xrightarrow{k} B$
We have $PART(B) = \{\{1,2\},\{3,5,6,7\},\{4,8\}\}$ and $PART(A) = \{\{1,2\},\{3,4,5\},\{6,7,8\}\}$, so $A \xrightarrow{\frac{1}{4}} B$.

We determine k for $B \xrightarrow{k} C$
We have $PART(C) = \{\{1,3,6\},\{2,5,7\},\{4,8\}\}$ and $PART(B) = \{\{1,2\},\{3,5,6,7\},\{4,8\}\}$, so $B \xrightarrow{\frac{1}{4}} C$.

We determine k for $A \xrightarrow{k} C$
We have $PART(C) = \{\{1,3,6\},\{2,5,7\},\{4,8\}\}$ and $PART(A) = \{\{1,2\},\{3,4,5\},\{6,7,8\}\}$, so $A \xrightarrow{0} C$.

This example is illustrating another interesting case: we can have: $A \xrightarrow{k} B$, $B \xrightarrow{k} C$ and $A \xrightarrow{0} C$ with $k \neq 0$.

3 A generalization of the rough set based dependencies

In this section we will use another approach to generalize dependency rules based on measures called predictability factors. We first define the predictability factors and then we go again through the same set of examples as in the previous section.

3.1 Definitions

Definition 4 *Consider $X_i \in PART(X)$ and $Y_j \in PART(Y)$. Then we define:*

$$P(Y_j|X_i) = \frac{card(X_i \cap Y_j)}{card(X_i)}$$

$$P(Y_j|X) = \sum_i P(Y_j|X_i) \frac{card(X_i \cap Y_j)}{card(Y_j)}$$

$$P(Y|X) = \sum_j P(Y_j|X) \frac{card(Y_j)}{card(T)}$$

□

$P(Y|X)$ is a similar factor to k from Definition 3. We will also use the notation $k(X,Y)$ for $P(Y|X)$. We will use the term *predictability factors* to refer to the expressions P.

We also have the following relations:
$P(Y_j|X) = \sum_i \frac{\mathrm{card}(X_i \cap Y_j)}{\mathrm{card}(X_i)} \frac{\mathrm{card}(X_i \cap Y_j)}{\mathrm{card}(Y_j)} = \sum_i \frac{\mathrm{card}^2(X_i \cap Y_j)}{\mathrm{card}(X_i)\,\mathrm{card}(Y_j)}$
and
$P(Y|X) = \sum_j \left(\sum_i \frac{\mathrm{card}^2(X_i \cap Y_j)}{\mathrm{card}(X_i)\,\mathrm{card}(Y_j)} \right) \frac{\mathrm{card}(Y_j)}{\mathrm{card}(T)} =$
$\sum_j \sum_i \frac{\mathrm{card}^2(X_i \cap Y_j)}{\mathrm{card}(X_i)\,\mathrm{card}(Y_j)} \frac{\mathrm{card}(Y_j)}{\mathrm{card}(T)} = \frac{1}{\mathrm{card}(T)} \sum_j \sum_i \frac{\mathrm{card}^2(X_i \cap Y_j)}{\mathrm{card}(X_i)}$

Notes

The first predictability factor in the definition simply gives the probability that if attribute X has value X_i then attribute Y will have value Y_j. As such it takes values between 0 and 1. The other two predictability factors are scaled in such a way as to obtain values between 0 and 1. The intuition is that a predictability factor of 1 indicates a strong dependency.

3.2 Properties

Theorem 4 *The following properties hold for the new definition:*

1. $X \xrightarrow{k} Y \Rightarrow XZ \xrightarrow{k'} YZ, \quad k' \geq k$
2. $X \rightarrow Y, Y \xrightarrow{k} Z \Rightarrow X \xrightarrow{k'} Z, \quad k' \geq k$
3. $X \xrightarrow{k} Y, Y \rightarrow Z \Rightarrow X \xrightarrow{k'} Z, \quad k' \geq k$
4. $X \xrightarrow{k} YZ \Rightarrow X \xrightarrow{k'} Y, \quad k' \geq k$

□

Proof: The proof of these properties is lengthier and for it we refer you to [1] and [2]. In those papers we have generalized the notion of entropy and also that of functional dependencies. The above properties are obtained using the gini function as a generator function and using a slightly different definition of the impurity between the partitions defined by two sets of attributes, namely: $IMP_{\pi_Y}(\pi_X) = (\sum_{B_i \in \pi_X} IMP_{\pi_Y}(B_i))/|T|$. With the above in mind, $P(Y|X)$ is the equivalent of $1 - IMP_{\pi_Y}(\pi_X)$, so it acts as a measure of purity between the partitions defined by two sets of attributes.

□

3.3 Examples

Using the new definitions, we will go again over the examples from the previous section:

Example 1 Now we have:

$k(A, C) = 1/7(9/3 + 1/2 + 1/2 + 1/2 + 1/2) = 5/7$
$k(AB, BC) = 1/8(1 + 1 + 1 + 1 + 1 + 1/2 + 1/2) = 6/7$
$k(A, B) = 1/7(1/3 + 1/3 + 1/3 + 4/2 + 1/2 + 1/2) = 4/7$
$k(B, C) = 1/7(1/3 + 1/3 + 1/3 + 1/2 + 1/2 + 1/2 + 1/2) = 3/7$

Example 2 We have:

$k(A, B) = 1/8(4/2+4/3+1/3+4/3+1/3) = 1/8(4/2+10/3) = 1/8(32/6) = 2/3$
$k(B, C) = 1/8(1/2 + 1/2 + 4/4 + 4/4 + 4/2) = 1/8(3 + 2) = 5/8$
$k(A, C) = 1/8(1/2 + 1/2 + 1/3 + 1/3 + 1/3 + 1/3 + 1/3 + 1/3) = 3/8$

4 Conclusions

We have shown how functional dependencies can be generalized using elements of rough sets theory and we have proved some properties of these generalized dependency rules. It is interesting that similar rules have also been proved for a generalization of functional dependencies obtained using entropy ([3]) and also for generalizations of entropy ([1], [2]). The generalizations introduced in [1] and [2] do not seem however to cover the rough-sets dependencies; it doesn't seem possible to find a generator function that would allow us to obtain the rough-sets based dependencies. It is possible that the notions of impurity and generalized entropy can be further generalized such as to encompass the rough sets dependencies but further work is required in order to do that.

References

[1] Simovici D., Cristofor D., Cristofor L.:*Mining for Purity Dependencies in Databases*, UMass/Boston Dept. of Math. and Comp. Sci. Technical Report 00-2, 2000

[2] Simovici D., Cristofor D., Cristofor L.:*Generalized Entropy and Projection Clustering of Categorical Data*, UMass/Boston Dept. of Math. and Comp. Sci. Technical Report 00-3, 2000

[3] Simovici D., Jaroszewicz S.:*On Information-Theoretical Aspects of Relational Databases*, International Journal on Multiple-Valued Logic, 1999.

[4] Ziarko W.:*The Discovery, Analysis, and Representation of Data Dependencies in Databases*, G. Piatetsky-Shapiro, and W. J. Frawley (eds.), Knowledge Discovery in Databases, AAAI Press/MIT Press, 177–195.