On Granular Rough Computing with Missing Values

Lech Polkowski[1,2] and Piotr Artiemjew[2]

[1] Polish–Japanese Institute of Information Technology
Koszykowa 86, 02008 Warszawa, Poland
[2] Department of Mathematics and Computer Science
University of Warmia and Mazury, Olsztyn, Poland
polkow@pjwstk.edu.pl,artem@matman.uwm.edu.pl

Abstract. Granular Computing as a paradigm in Approximate Reasoning is concerned with granulation of available knowledge into granules that consists of entities similar in information content with respect to a chosen measure and with computing on such granules. Thus, operators acting on entities in a considered universe should factor through granular structures giving values similar to values of same operators in non–granular environment. Within rough set theory, proposed 25 years ago by Zdzisław Pawlak and developed thence by many authors, granulation is also a vital area of research. The first author developed a calculus with granules as well as a granulation technique based on similarity measures called rough inclusions along with a hypothesis that granules induced in data set universe of objects should lead to new objects representing them, and such granular counterparts should preserve information content in data. In this work, this hypothesis is tested with missing values in data and results confirm the hypothesis in this context.

Keywords: rough sets, decision systems, missing values, granules of knowledge, rough inclusions, granular decision systems.

1 Rough Computing

Rough sets are centered about the notion of *indiscernibility*[7]: entities with same description are regarded as identical. In practical terms, when knowledge is encoded in an *information system* (U, A) where U is a set of *entities* and A is a set of *attributes*, with each $a : U \to V_a$ a mapping on U into a value set, indiscernibility is given as an equivalence $ind(a) = \{(u, v) : u, v \in U, a(u) = a(v)\}$ for each $a \in A$, with extensions of the form $ind(B) = \bigcap_{a \in B} ind(a)$ for any $B \subseteq A$.

Rough computing is usually performed with *descriptors* of the form $(a = v)$, $v \in V_a$, interpreted as sets $[(a = v)] = \{u \in U : a(u) = v\}$; descriptors extend to *descriptor formulas* that form the smallest set containing all descriptors and closed on the action of propositional connectives $\vee, \wedge, \neg, \Rightarrow$; descriptor formulas are interpreted via identities $[\bigwedge_i (a_i = v_i)]$=$\bigcap_i [(a_i = v_i)]$, $[\bigvee_i (a_i = v_i)]$=$\bigcup_i [(a_i =$

M. Kryszkiewicz et al. (Eds.): RSEISP 2007, LNAI 4585, pp. 271–279, 2007.

$v_i)]$, $[\neg(a = v)]$=$U \setminus [(a = v)]$. *Decision systems* are information systems of the form $(U, A \cup \{d\})$, where d, the *decision*, is an attribute not in A; relations between the *conditional knowledge* (U, A) and the *world knowledge* (U, d) are expressed by means of *decision rules* of the form $\bigwedge_i (a_i = v_i) \Rightarrow (d = v)$; a set of decision rules is a *classifier*; its aim is to recognize decision classes of new entities on the basis of their conditional values.

2 Missing Values

An information/decision system is *incomplete* in case some values of conditional attributes from A are not known; some authors, e.g., Grzymala–Busse [2] make distinction between values that are *lost* (denoted ?), i.e., they were not recorded or were destroyed in spite of their importance for classification, and values that are *missing* (denoted $*$) as those values that are not essential for classification. Here, we regard all lacking values as missing without making any distinction among them denoting all of them with $*$. Analysis of systems with missing values requires a decision on how to treat missing values; Grzymala–Busse in his work [2], analyzes nine such methods known in the literature, among them, *1. most common attribute value*, *2. concept–restricted most common attribute value*, (...), *4. assigning all possible values to the missing location*, (...), *9. treating the unknown value as a new valid value.* Results of tests presented in [2] indicate that methods *4,9* perform very well among all nine methods. For this reason we adopt these methods in this work for the treatment of missing values and they are combined in our work with a modified method *1*: the missing value is defined as the most frequent value in the granule closest to the object with the missing value with respect to a chosen rough inclusion.

Analysis of decision systems with missing data in existing rough set literature relies on an appropriate treatment of indiscernibility: one has to reflect in this relation the fact that some values acquire a distinct character and must be treated separately; in case of missing or lost values, the relation of indiscernibility is usually replaced with a new relation called a *characteristic relation.* Examples of such characteristic functions are given in, e.g., Grzymala–Busse [3]: the function ρ is introduced, with $\rho(u, a) = v$ meaning that the attribute a takes on u the value v. Semantics of descriptors is changed, viz., the meaning $[(a = v)]$ has as elements all u such that $\rho(u, a) = v$, in case $\rho(u, a) =?$ the entity u is not included into $[(a = v)]$, and in case $\rho(u, a) = *$, the entity u is included into $[(a = v)]$ for all values $v \neq *, ?$. Then the characteristic relation is $R(B) = \{(u, v) : \forall .a \in B.\rho(u, a) =? \Rightarrow (\rho(u, a) = \rho(v, a) \vee \rho(u, a) = * \vee \rho(v, a) = *)\}$, where $B \subseteq A$. Classes of the relation $R(B)$ are then used in defining approximations to decision classes from which certain and possible rules are induced, see [3]. Specializations of the characteristic relation $R(B)$ were defined in Stefanowski–Tsoukias [18] (in case of only lost values) and in Kryszkiewicz [4](in case of only don't care missing values). An analysis of the problem of missing values along with algorithms *IApriori Certain* and *IAprioriPossible* for certain and possible rule generation was given in [5].

3 Granules of Knowledge and Granular Information/Decision Systems

Granulation of knowledge is a topic studied recently to much extent within rough set theory, see, e.g., [14],[15]. We describe briefly a method for inducing granules [10], [11] which consists in selecting a rough inclusion μ (see op.cit.), and $r \in [0, 1]$.

3.1 Rough Inclusions

Generally they are predicates of the form $\mu(u, v, r)$, where $u, v \in U$ satisfying conditions, 1. $\mu(u, u, 1)$;2. if $\mu(u, v, 1)$ then for each $w \in U$, from $\mu(w, u, r)$ it follows $\mu(w, v, r)$; 3. if $\mu(u, v, r)$ and $s < r$ then $\mu(u, v, s)$. For an analysis of various methods for inducing rough inclusions see, e.g., [10], [11]. In this work we will use exclusively the rough inclusion $\mu_L(u, v, r)$ satisfied if and only if $\frac{|IND(u,v)|}{|A|} \geq r$, where $IND(u, v) = \{a \in A : a(u) = a(v)\}$, induced by the Łukasiewicz implication (see, e.g., [10],[11]).

3.2 On Granule Formation

For a rough inclusion μ, $u \in U$, and $r \in [0, 1]$, the granule $g_\mu(u, r)$ is defined as the class $Cls\{v : \mu(v, u, r)\}$, where Cls is the class forming functor of mereology, see, e.g., [10],[11]; for the purpose of this work, one may assume that $g_\mu(u, r)$ is the list or the set of all v such that $\mu(v, u, r)$. In this work, granules are formed only by means of μ_L. In plain words, the granule $g_{\mu_L}(u, r)$ consists of all $v \in U$ with the property that $|IND(v, u)| \geq r \cdot |A|$, i.e., v, u have identical values of at least $r \cdot 100$ percent of attributes in A.

3.3 Granular Information/Decision Systems

The idea of a granular decision system was posed in [10]; for a given information system (U, A), a rough inclusion μ, and $r \in [0, 1]$, the new universe $U^G_{r,\mu}$ is given, whose elements are granules of the radius r about objects $u \in U$. We apply a strategy $\mathcal{G}$ to choose a covering $Cov^G_{r,\mu}$ of the universe U by granules from $U^G_{r,\mu}$.

We apply a strategy $\mathcal{S}$ in order to assign the value $a^*(g)$ of each attribute $a \in A$ to each granule $g \in Cov^G_{r,\mu}$: $a^*(g) = \mathcal{S}(\{a(u) : u \in g\})$. The granular counterpart to the information system (U, A) is a tuple $(U^G_{r,\mu}, \mathcal{G}, \mathcal{S}, \{a* : a \in A\})$; analogously, we define granular counterparts to decision systems by adding the factored decision $d*$. The heuristic principle that *objects, similar with respect to conditional attributes in the set A, should also reveal similar (i.e., close) decision values, and therefore, granular counterparts to decision systems should lead to classifiers satisfactorily close in quality to those induced from original decision systems*, was stated in [10], and borne out by simple hand examples. The hypothesis has been confirmed in [12] and in this work we apply this hypothesis to the problem of missing values.

4 An Approach to Missing Values in This Work

We will use the symbol $*$ commonly used for denoting the missing value; we will use two methods *4, 9* for treating $*$, i.e, either $*$ is a *don't care* symbol meaning that any value of the respective attribute can be substituted for $*$,thus $* = v$ for each value v of the attribute, or $*$ is a new value on its own, i.e., if $* = v$ then v can be only $*$.

Our procedure for treating missing values is based on the granular structure $(U^G_{r,\mu}, \mathcal{G}, \mathcal{S}, \{a* : a \in A\})$; the strategy $\mathcal{S}$ is the majority voting, i.e., for each attribute a, the value $a^*(g)$ is the most frequent of values in $\{a(u) : u \in g\}$, with ties broken randomly. The strategy $\mathcal{G}$ consists in random selection of granules for a covering.

For an object u with the value of $*$ at an attribute a, and a granule $g = g(v, r) \in U^G_{r,\mu}$, the question whether u is included in g is resolved according to the adopted strategy of treating $*$: in case $* = don't\ care$, the value of $*$ is regarded as identical with any value of a hence $|IND(u, v)|$ is automatically increased by 1, which increases the granule; in case $* = *$, the granule size is decreased. Assuming that $*$ is sparse in data, majority voting on g would produce values of a^* distinct from $*$ in most cases; nevertheless the value of $*$ may appear in new objects g^*, and then in the process of classification, such value is repaired by means of the granule closest to g^* with respect to the rough inclusion μ_L, in accordance with the chosen method for treating $*$.

In plain words, objects with missing values are in a sense absorbed by close to them granules and missing values are replaced with most frequent values in objects collected in the granule; in this way the method *4* or *9* in [3] is combined with the idea of the most frequent value *1*, in a novel way.

We have thus four possible strategies:

- Strategy A: in building granules $*$=*don't care*, in repairing values of $*$, $*$=*don't care*;
- Strategy B: in building granules $*$=*don't care*, in repairing values of $*$, $* = *$;
- Strategy C: in building granules $* = *$, in repairing values of $*$, $*$=*don't care*;
- Strategy D: in building granules $* = *$, in repairing values of $*$, $* = *$.

As data set used in experiments, Pima Indians diabetes data set [19] has been used. We first show results for this data set in granular and non–granular cases without missing values in Table 1, see [12] for a discussion of this method in more detail; then a randomly chosen collection of 10 percent of attribute values in the data set are replaced with $*$ values. Results of granular treatment in case of Strategies A,B,C,D are reported in Tables 2,3,4,5. As algorithm for rule induction, the exhaustive algorithm of the RSES system [16] has been selected, see, e.g., [1], [17], where the ideas implemented in the RSES package are discussed. 10–fold cross validation (CV–10) has been used to validate results of the experiment.

Table 1. 10-fold CV; Pima; exhaustive algorithm. r=radius,macc=mean accuracy,mcov=mean coverage,mrules=mean rule number, mtrn=mean size of granular training set

r	*macc*	*mcov*	*mrules*	*mtrn*
nil	0.6864	0.9987	7629.2	692.0
0.0	0.0	0.0	0.0	1.0
0.125	0.0618	0.0895	5.9	22.5
0.250	0.6627	0.9948	450.1	120.6
0.375	0.6536	0.9987	3593.6	358.7
0.500	0.6645	1.0	6517.7	579.4
0.625	0.6877	0.9987	7583.6	683.1
0.750	0.6864	0.9987	7629.2	692
0.875	0.6864	0.9987	7629.2	692.0

Table 2. Strategy A for missing values. 10-fold CV; Pima; exhaustive algorithm. r=radius, macc=mean accuracy, mcov=mean coverage, mrules=mean rule number, mtrn=mean size of granular training set

r	*macc*	*mcov*	*mrules*	*mtrn*
nil	0.6864	0.9987	7629.2	692.0
0.0	0.0	0.0	0.0	1.0
0.125	0.0	0.0	0.0	1.7
0.250	0.0	0.0	0.0	4.7
0.375	0.0	0.0	0.0	21.5
0.500	0.3179	0.4777	115.8	64.7
0.625	0.6692	0.9987	1654.7	220.2
0.750	0.6697	1.0	5519.3	527.0
0.875	0.6678	0.9987	7078.8	663.8

Table 3. Strategy B for missing values. 10-fold CV; Pima; exhaustive algorithm. r=radius,macc=mean accuracy, mcov=mean coverage, mrules=mean rule number, mtrn=mean size of granular training set

r	*macc*	*mcov*	*mrules*	*mtrn*
nil	0.6864	0.9987	7629.2	692.0
0.0	0.0	0.0	0.0	1.0
0.125	0.0	0.0	0.0	1.9
0.250	0.0	0.0	0.0	6.1
0.375	0.0	0.0	0.0	13.7
0.500	0.5772	0.8883	210.7	68.1
0.625	0.6467	0.9987	1785.8	229.4
0.750	0.6587	0.9987	5350.4	508.5
0.875	0.6547	0.9987	6982.7	663.4

Table 4. Strategy C for missing values. 10-fold CV; Pima; exhaustive algorithm. r=radius,macc=mean accuracy, mcov=mean coverage, mrules=mean rule number, mtrn=mean size of granular training set

r	*macc*	*mcov*	*mrules*	*mtrn*
nil	0.6864	0.9987	7629.2	692.0
0.0	0.0	0.0	0.0	1.0
0.125	0.0	0.0	0.0	21.2
0.250	0.6297	0.9948	388.9	116.9
0.375	0.6556	0.9974	3328.5	356.5
0.500	0.6433	1.0	6396.7	587.2
0.625	0.6621	1.0	7213.2	681.9
0.750	0.6640	0.9987	7306.3	691.9
0.875	0.6615	0.9987	7232.1	692.0

Table 5. Strategy D for missing values. 10-fold CV; Pima; exhaustive algorithm. r=radius, macc=mean accuracy, mcov=mean coverage, mrules=mean rule number, mtrn=mean size of granular training set

r	*macc*	*mcov*	*mrules*	*mtrn*
nil	0.6864	0.9987	7629.2	692.0
0.0	0.0	0.0	0.0	1.0
0.125	0.1471	0.1750	12.0	17.3
0.250	0.6572	0.9974	382.1	114.9
0.375	0.6491	0.9974	3400.3	355.0
0.500	0.6370	0.9974	6300.2	588.7
0.625	0.6747	0.9987	7181.2	682.3
0.750	0.6724	1.0	7231.3	691.9
0.875	0.6618	1.0	7253.6	692.0

5 Case of Real Data with Missing Values

We include results of tests with Breast cancer data set [19] that contains missing values. We show in Tables 6, 7, 8, 9 results for intermediate values of radii of granulation for strategies A,B,C,D and exhaustive algorithm of RSES [16]. For comparison, results on error in classification by the endowed system LERS from [2] for approaches similar to our strategies A and D (methods 4 and 9, resp., in Tables 2 and 3 in [2]) in which $*$ is either always $*$ (method 9) or $*$ is always *don't care* (method 4) are recalled in Tables 6 and 9. We have applied here the 1-train–and–9 test, i.e., the data set is split randomly into 10 equal parts and training set is one part whereas the rules are tested on each of remaining 9 parts separately and results are averaged.

5.1 Conclusions on Test Results

In case of perturbed Pima Indians diabetes data set, Strategy A attains accuracy value better than 97 percent and coverage value greater or equal to values in

Table 6. Breast cancer data set with missing values. Strategy A: r=granule radius, mtrn= mean granular training sample size, macc=mean accuracy, mcov=mean covering, gb=LERS method 4,[2]

r	*mtrn*	*macc*	*mcov*	*gb*
0.555556	9	0.7640	1.0	0.7148
0.666667	14	0.7637	1.0	
0.777778	17	0.7129	1.0	
0.888889	25	0.7484	1.0	

Table 7. Breast cancer data set with missing values. Strategy B: r=granule radius, mtrn= mean granular training sample size, macc=mean accuracy, mcov=mean covering

r	*mtrn*	*macc*	*mcov*
0.555556	7	0.0	0.0
0.666667	13	0.7290	1.0
0.777778	16	0.7366	1.0
0.888889	25	0.7520	1.0

Table 8. Breast cancer data set with missing values. Strategy C: r=granule radius, mtrn= mean granular training sample size, macc=mean accuracy, mcov=mean covering

r	*mtrn*	*macc*	*mcov*
0.555556	8	0.7132	1.0
0.666667	14	0.6247	1.0
0.777778	17	0.7328	1.0
0.888889	25	0.7484	1.0

Table 9. Breast cancer data set with missing values. Strategy D: r=granule radius, mtrn= mean granular training sample size, macc=mean accuracy, mcov=mean covering, gb=LERS method 9,[2]

r	*mtrn*	*macc*	*mcov*	*gb*
0.555556	9	0.7057	1.0	0.6748
0.666667	16	0.7640	1.0	
0.777778	17	0.6824	1.0	
0.888889	25	0.7520	1.0	

non–perturbed case from the radius of .625 on. With Strategy B, accuracy is within 94 percent and coverage not smaller than values in non–perturbed case from the radius of .625 on. Strategy C yields accuracy within 96.3 percent of accuracy in non–perturbed case from the radius of .625, and within 95 percent from the radius of .250; coverage is within 99.79 percent from the radius of .250. Strategy D gives results slightly better than C with the same radii. Results for C and D are better than results for A or B.

Table 10. Average number of $*$ values in granular systems. 10-fold CV; Pima; exhaustive algorithm. r=radius,mA=mean value for A, mB=mean value for B , mC=mean value for C, mD=mean value for D

r	mA	mB	mC	mD
0.375	0.0	0.0	135	132
0.500	0.0	0.0	412	412
0.625	3	4	538	539
0.750	167	167	554	554
0.875	435	435	554	554

We conclude that essential for results of classification is the strategy of treating the missing value of $*$ as $* = *$ in both strategies C and D; the repairing strategy has almost no effect: C and D differ with respect to this strategy but results for accuracy and coverage in cases C and D differ very slightly.

Let us notice that strategies C and D cope with a larger number of $*$ values to be repaired. Table 10 shows this.

In experiments with Breast cancer data set with missing values, best results are obtained with "pure" strategies A and D; strategy A gives accuracy of .7637 at $r = .(6)$ and strategy D gives accuracy of .7640 at $r = .(6)$, "mixed" strategies give best results at higher value of radius of .(7): .7474 in case of C and .7520 in case of B.

6 Conclusions

The method proposed in this work for treatment of missing values that combines either of two approaches, viz., $*$*don't care* or $* = *$ with the idea of absorbing objects with missing values into granules consisting of objects close to them to a degree specified by radii of granules, followed by the idea of replacing the missing value with the most frequent value over the granule, has proved very effective in the classification problem of data with missing values.

In the stage of repairing the missing value, strategies C and D proved most effective. Essential for results of classification is the strategy of treating the missing value of $*$ as $* = *$ in building granules as witnessed by cases of strategies C and D; strategies A and B give comparable results between them, implying that when the strategy $*=$*don't care* is used in building granules, then the choice of a repairing strategy has no practical impact.

Further research will be focused on more refined ways of granule selection, development of a granular algorithm for rule induction, and analysis of large real data with missing values.

References

1. Bazan, J.G.: A comparison of dynamic and non–dynamic rough set methods for extracting laws from decision tables. In: Polkowski, L., Skowron, A. (eds.) Rough Sets in Knowledge Discovery 1, pp. 321–365. Physica Verlag, Heidelberg (1998)

2. Grzymala–Busse, J.W., Hu, M.: A comparison of several approaches to missing attribute values in data mining. In: RSCTC 2000. LNCS (LNAI), vol. 2005, pp. 378–385. Springer, Berlin (2000)
3. Grzymala–Busse, J.W.: Data with missing attribute values: Generalization of rule indiscernibility relation and rule induction. In: Transactions on Rough Sets I, pp. 78–95. Springer, Berlin (2004)
4. Kryszkiewicz, M.: Rules in incomplete information systems. Information Sciences 113, 271–292 (1999)
5. Kryszkiewicz, M., Rybiński, H.: Data mining in incomplete information systems from rough set perspective. In: Polkowski, L., Tsumoto, S., Lin, T.Y. (eds.) Rough Set Methods and Applications, pp. 568–580. Physica Verlag, Heidelberg (2000)
6. Leśniewski, S.: On the foundations of set theory vol. 2, pp. 7–52 (1982)
7. Pawlak, Z.: Rough Sets: Theoretical Aspects of Reasoning about Data. Kluwer, Dordrecht (1991)
8. Polkowski, L.: Rough Sets. Mathematical Foundations. Physica Verlag, Heidelberg (2002)
9. Polkowski, L.: oward rough set foundations. Mereological approach (a plenary lecture). In: Tsumoto, S., Słowiński, R., Komorowski, J., Grzymała-Busse, J.W. (eds.) RSCTC 2004. LNCS (LNAI), vol. 3066, pp. 8–25. Springer, Heidelberg (2004)
10. Polkowski, L.: Formal granular calculi based on rough inclusions (a feature talk) In: [14], pp. 57–62
11. Polkowski, L.: A model of granular computing with applications (a feature talk), In: [15], pp. 9–16
12. Polkowski, L., Artiemjew, P.: On granular rough computing: Factoring classifiers through granulated decision systems. In: these Proceedings
13. Polkowski, L., Skowron, A.: Rough mereology: a new paradigm for approximate reasoning. International Journal of Approximate Reasoning 15(4), 333–365 (1997)
14. Proceedings of IEEE 2005 Conference on Granular Computing. In: GrC05, Beijing, China, July 2005, IEEE Press, New York (2005)
15. Proceedings of IEEE 2006 Conference on Granular Computing. In: GrC06, Atlanta, USA, May 2006, IEEE Press, New York (2006)
16. Skowron, A., et al.: RSES: A system for data analysis, available at `http://logic.mimuw.edu.plrses/`
17. Nguyen, S.H.: Regularity analysis and its applications in Data Mining. In: Polkowski, L., Tsumoto, S., Lin, T.Y. (eds.) Rough Set Methods and Applications, pp. 289–378. Physica Verlag, Heidelberg (2000)
18. Stefanowski, J., Tsoukias, A.: Incomplete information tables and rough classification. Computational Intelligence 17, 545–566 (2001)
19. `http://www.ics.uci.edu.mlearn/databases/`