Decomposition in Data Mining: A Medical Case Study

Andrew Kusiak
Intelligent Systems Laboratory
4312 Seamans Center
The University of Iowa
Iowa City, Iowa 52242 – 1527

andrew-kusiak@uiowa.edu
http://www.icaen.uiowa.edu/~ankusiak

ABSTRACT

Decomposition is a tool for managing complexity in data mining and enhancing the quality of knowledge extracted form large databases. A typology of decomposition approaches applicable to data mining is presented. One of the decomposition approaches, the structured rule-feature matrix, is used as the backbone of a system for informed decision-making. Such a system can be implemented as a decision table, a decision map, or a decision atlas. The ideas presented in the paper are illustrated with examples and a medical case study.

Keywords*:* Data mining, decision making, decomposition, rule structuring, disease diagnosis, lung cancer.

1. INTRODUCTION

Data mining is concerned with discovery of patterns, associations, rules, and other forms of knowledge in data sets. This knowledge is automatically extracted from data rather than being formulated by a user as it is done in traditional modeling approaches, e.g., statistical or optimization modeling. As a new discipline, data mining draws from other areas such as statistics, machine learning, database retrieval, pattern recognition, and high performance computing.

In many applications, data is automatically generated and therefore the number of objects to be mined can be large. The time needed to extract knowledge from such large data sets is an issue, as it may easily run in days, weeks, and beyond. One way to reduce computational complexity of knowledge discovery with data mining algorithms and decision making based on the acquired knowledge is to reduce the volume of data to be processed at a time, which can be accomplished by decomposition. In this paper, numerous decomposition approaches are defined and applied for effective knowledge discovery and decision making. Besides simplifying computation, decomposition facilities dynamic extraction of knowledge that can be used for real-time decision making.

1.1 Decomposition in the Literature

Decomposition has been discussed in the data mining literature, however, largely in the context of distributed learning. This research emphasizes the use of decomposition to enhance decision making rather than learning. One of the most comprehensive sources of papers on distributed learning is the volume edited by Zaki and Ho (2000). Several contributions to this book discussed ways of leveraging parallel and distributed techniques in knowledge discovery, such as data cleaning and preprocessing, transformation, and learning. Grossman *et al.* (1999) outlined fundamental challenges for mining large-sale databases, with one of them being the need to develop distributed data mining algorithms. Guo and Sutiwaraphun (1988)

B.V. Dasarathy (Ed.), *Proceedings of the SPIE Conference on Data Mining and Knowledge Discovery: Theory, Tools, and Technology III*, Vol. 4384, SPIE, Orlando, FL, April 2001, pp. 267-277.

described a meta-learning concept named Knowledge Probing to distributed data mining. In Knowledge Probing, supervised learning is organized into two stages. At the first stage, a set of base classifiers is learned in parallel from a distributed data set. At the second stage, the relationship between an attribute vector and the class predictions from all of the base classifiers is determined. Zaki *et al.* (1999) discussed a project called SPIDER that uses shared-memory multiprocessors systems (SMPs) to accomplish parallel data mining on distributed data sets.
Cluster analysis provides the basic theory and algorithms for decomposition (Anderberg 1973). Some of the most efficient clustering algorithms are presented in Kusiak (2000).

1.2 Machine Learning

Bazan (1998) categorized the existing learning algorithms as follows:

- Decision tree (Friedman *et al.* 1996, Quinlan 1986)
- Decision rule (Clark and Boswell 1989)
- Inductive logic programming (Michalski *et al.* 1986), and
- Rough set algorithms (Grzymala-Busse 1997, Pawlak 1982)

2. TYPOLOGY OF DECOMPOSITION IN DATA MINING

There are two basic approaches to data mining:

- Direct mining of data sets
- Mining of transformed data sets

The first approach is most often applied for mining data sets that can be processed in an acceptable time by the existing data mining algorithms. Transforming data sets before mining is intended either for large data sets or data sets with special properties, e.g., hybrid data discussed in the next section. One of the most useful forms of data transformation is decomposition, which may take place in space and time. The area of decomposition in time is extensive has received rather broad coverage in the literature and is beyond the scope of this paper.

The following two forms of decomposition in space are considered in this paper (Kusiak 2000a):

- Feature set decomposition
- Object set decomposition

The feature set decomposition is further classified into:

- Content-based decomposition; The feature set is decomposed into mutually exclusive or partially overlapping subsets with the same decision D used for each subset. The feature origin, availability, and any other criterion could drive the content of each feature set.
- Intermediate-decision decomposition; In some applications feature values are generated over time. In addition the downstream features may dependent on the upstream features.
- Feature type decomposition; Some of the existing rule extraction algorithms are intended for specific types of features, e.g., discrete value features.
- Feature relevance decomposition; Features may show various degree of relevance to the outcome, measured with statistical metrics (e.g., correlation) and context relationship, which is more tacit and difficult to measure (e.g., the impact of outside temperature on computer energy consumption).

The object set decomposition is further classified as:

- Object content decomposition; Objects are grouped according to time interval, origin, applicability, and so on.
- Decision value decomposition; The set of objects is split into subsets according to the decision value.
- Feature value decomposition; The objects (and possibly features) are partitioned into subsets based on the value of selected features.

3. HYBRID MODELS

One of the most meaningful applications of decomposition in data mining is hybrid modeling. A hybrid model is a collection of models of different types, for example, models developed on the first principles and models constructed from the knowledge extracted by machine learning algorithms. Hybrid models are often built because of different degree of understanding of the modeled process, availability of data, and other application specific limitations.

The need for hybrid models has been motivated by numerous engineering and medical applications. A typical process, e.g., a semiconductor manufacturing process or a disease management process, involves stages that are well understood due to available models and stages that are only loosely known. This lack of process knowledge is likely behind unwanted events in industrial processes (e.g., products below expected quality level) and in medicine (e.g., premature patient's death).

This paper meets the need for hybrid models involving both well-defined models and knowledge models derived from the data collected while observing an actual process. During the process operations, both desirable and adverse events occur, and therefore, the data collected is transformed into knowledge that can be used to minimize or even prevent adverse events from happening.

A domain of interest captured by the hybrid decision-making model in Figure 1 is decomposed into three stages. The model in Figure 1 allows for bi-directional reasoning, which is important for in-depth understanding of the modeled process.

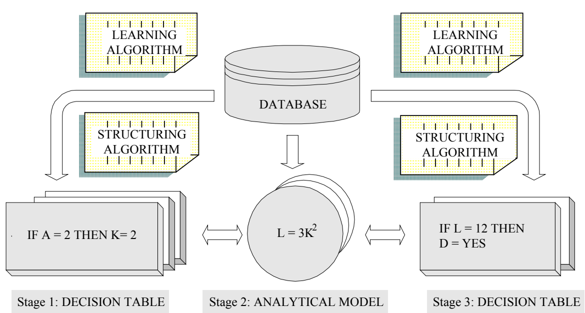

Figure 1. Hybrid decision making process.

The rule-structuring algorithm (Kusiak 2000a) organizes the knowledge extracted by different learning algorithms into decision tables that will be integrated with analytical models as well as models of other types as illustrated in Figure 1. The structured knowledge is 'packaged' in the form of decision tables that can be combined in other constructs such as decision maps, and those in turn in decision atlases. These constructs will increase transparency and effectiveness of the decision-making process.

A decision table provides decision basis (e.g., decision rules, rule support, rule coverage, etc.) and justification for the decision (e.g., historical cases supported by the decision rule) as symbolically illustrated in Figure 2.

The entries of the decision table in Figure 2 are attribute values generated by a learning algorithm. Each entry may represent a singular numerical or symbolic value, a bounded range of values, an unbounded value range (inequality), and so on.

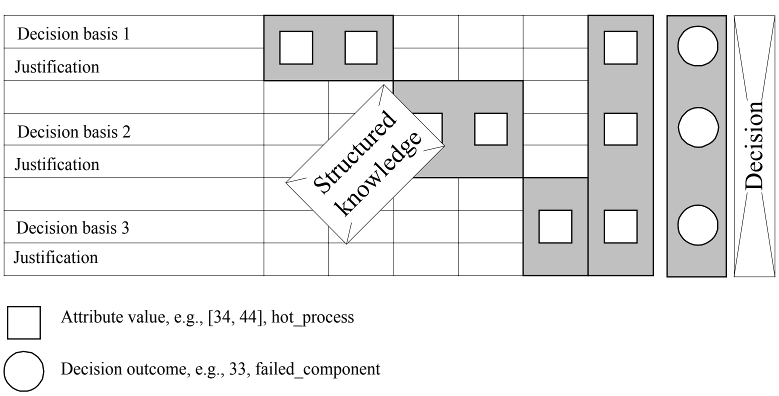

Figure 2. Example of a decision table.

Each decision basis contains feature values, while justification characterizes the decision basis, for example, it may contain the decision confidence and risks associated with a decision.

The simplest way of making decisions with a decision table is to match the values of an object with unknown outcome to an appropriate row of the decision table. Other ways of making robust decisions with orthogonal algorithms are discussed in Kusiak *et al.* (2000).

Example 1 illustrates a typical hybrid process.

***Example* 1**

Consider a five stage process in which the models at stages 1, 3, and 5 are not known, while stages 2 and 4 are modeled with known functions F_1 and F_2 (see the functions below and Figure 3). During a three-month

B.V. Dasarathy (Ed.), *Proceedings of the SPIE Conference on Data Mining and Knowledge Discovery: Theory, Tools, and Technology III*, Vol. 4384, SPIE, Orlando, FL, April 2001, pp. 267-277.

period, for process stages 1, 3, and 5 three data sets containing numerous observations (objects) on selected features f_i , i = 1, ..., 9 and the decision D have been collected. A learning algorithm has extracted three sets 1, 2, and 3 of decision rules shown next.

Rule set 1
IF $f_1 = 2$ AND f_2 = Low THEN $f_3 = 4$
IF f_2 in [2.1, 4] THEN $f_3 = 5$
Function F_1
$f_4 = 3.1 + (f_3 - 3.1)^3$
Rule set 2
IF $f_4 < 8.5$ AND f_5 = High THEN $f_6 = 8.4$
IF f_5 = Low THEN $f_6 = 12.4$
Function F_2
$f_7 = \ln (f_6 + 2.9)^{1/2}$
Rule set 3
IF $f_7 < 1.3$ THEN D = Good
IF $f_8 >= 3.3$ AND f_9 = Positive THEN D = Bad

The content of each of the three rule sets may change frequently while the two functions remain the same. In fact, the two functions F_1 and F_2 could interface with alternative rules generated from the corresponding data sets.

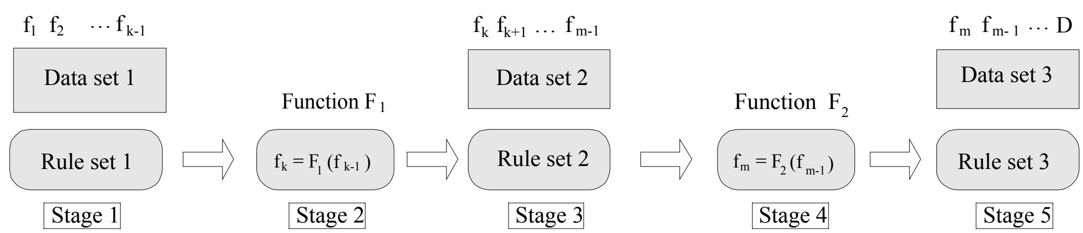

Figure 3. Example process with three rule sets linked by two functions.

When the number of rules is large, their collective interactions are difficult to understand. For better usability of decision rules they will be organized by the rule-structuring algorithm (Kusiak 2000a) and represented as decision tables, decision maps, or decision atlases.

The decomposition approach discussed in this paper offers the following advantages:
- Ease of model construction and understanding; The data set is partitioned into independent subsets (data sets 1, 3, and 5 in Figure 3) and therefore the rule induction is simplified;
- Support of the evolutionary computation concept; The basic the premise of the discussed approach is that the models are separable and they change with different frequency. Evolutionary computation algorithms can be involved in individual models as well as controlling the evolution of the overall model;
- Increased model structural stability; Only one sub-model (a rule set or a function in Figure 3) at a time is usually modified;
- Ease of data acquisition and model maintenance; As the scope of data at each stage is limited it is easier to acquire the data and maintain the component models;
- Reuse of known models and dependencies; Rather than building a new model from scratch, the model is built around existing component models, e.g., functions and neural network models;
- Representation of alternative solutions with positive and negative rules; The role of negative rules in decision making is discussed in Tsumoto (2000).

3.1 Knowledge Structuring
One of the main reasons for extracting knowledge from data sets is to use it for decision making, which has not received sufficient attention in the literature. Most of decision-making algorithms are rather simplistic and usually based on partial or full matching schemes. Many users have difficulty accepting decision rules that are non-intuitive and algorithms making

decisions based on non-transparent matching. This paper addresses this important gap in the presentation of knowledge for effective decision making.
The knowledge extracted by a learning algorithm is usually in the form of decision rules that make predictions at some level of accuracy, typically far from perfect. Decision rules might be numerous, the relationships discovered may be flawed, their meaning might be difficult to understand, and so on. In other words, users have certain expectations for the knowledge discovered that are outside of the scope of learning algorithms. The rule-structuring models and algorithms to be developed in this research will meet these expectations. They will be used both for supporting the user view as well as autonomous decision-making.

The rule-structuring concept is illustrated in Example 2.
Example 2

f_1	f_2	f_3	f_4	f_5	D	Rule	Algorithm
{B, C, D}	a				Low	R1	A1
			>9		Medium	R2	A2
			<2	(2, 5]	Low	R7	A3
		(2, 6]		=<8	High	R3	A2
{E, F}	b				Low	R4	A3
{C, F}		<4			Medium	R8	A1
			>9		Medium	R5	A1
		>=2		[1, 3]	High	R6	A3

Figure 4. Rule-feature matrix.

f_3	f_5	f_2	f_1	f_4	D	Rule	Algorithm
(2, 6]	=<8				High	R3	A2
>=2	[1, 3]				High	R6	A3
		a	{B, C, D}		Low	R1	A1
		b	{E, F}		Low	R4	A3
				>9	Medium	R2	A2
				>9	Medium	R5	A1

Figure 5. Structured rule-feature matrix.

Three different learning algorithms A1 – A3 were used to extract eight decision rules R1 – R8 from a data set. These rules R1 – R8 are represented as the rule-feature matrix in Figure 4. To simplify our considerations the information pertinent to each rule such as support, classification quality, and so on has not been included.

Though the rule set in Figure 4 is small, its analysis is not simple. Transforming the matrix in Figure 4 into the structured matrix in Figure 5 significantly improves interpretation of the rule set. The rule-structuring algorithm (Kusiak 2000a) generated the matrix in Figure 5 from the one in Figure 4 by removing two rules R7 and R8 due to their dissimilarity with the rules R1 through R6 and changing the sequence of the remaining rows and columns.

The content of the matrix in Figure 5 is structured and it allows drawing numerous conclusions, for example:
- ❑ The decisions D = High, Medium, and Low are totally separated by features;
- ❑ The rules R3 and R6 generated by algorithms A2 and A3 are equivalent;
- ❑ The decision D = Low can be reached in two alternative ways, using the feature values f_1 = {B, C, D} and f_2 = a, or f_1 = {E, F} and f_2 = b.

Example 2 illustrates only a few of numerous users' requirements that can be incorporated in the rule-structuring algorithm, such as:
- ❑ Matrix structure; Different structures of the rule-feature matrix may be considered, e.g., block-diagonal (see Figure 5), block-diagonal matrix with overlapping features, block-diagonal matrix with overlapping rules, triangular (for dependency analysis among rules), L-shape matrix, T-shape, etc.
- ❑ Differentiation of decisions on features; Each decision value is associated with an independent subset of features;
- ❑ Differentiation of decisions on feature values; Any two decision values are discernable on unique subset of feature values;
- ❑ Inclusion of user selected features; A user may have her/his preferences in terms of the features to be included in the selected rules, exclusion of some features, presence of the minimum set of features, and so on;
- ❑ Contrasting positive rules against negative ones is valuable in decision-making in some applications.

3.2 Rule-Structuring Model

The rule-structuring problem can be represented an *m*-partite graph with each node representing a set of rules called a candidate rule set. The rules contained in each candidate rule set share some common property, e.g., same decision value, common feature set, which depends on the objective of the rule-structuring problem. The candidate rule sets could be defined for the rules extracted from a specific data set, e.g., data set 1 in Figure 3, or across different data sets, e.g., data set 1 and 3 in

Figure 3. Candidate rules sets may include rules extracted by different learning algorithms. An arc of the *m*-partite graph represents relationships between the corresponding nodes, e.g., a distance.

Transforming the unstructured matrix in Figure 4 into the structured matrix in Figure 5 calls for development of a rule-structuring model that could be solved with a standard or a specialized algorithm. Both modeling and solution alternatives will be explored in this research. The rule-structuring model is illustrated in Example 3.

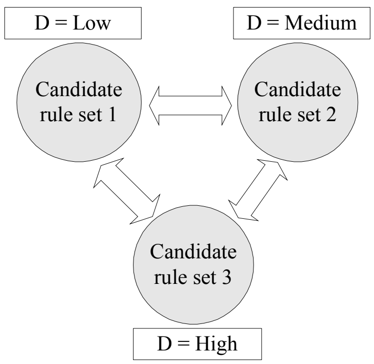

Figure 6. Graph representation of the rule-structuring problem for the rules in Figure 4.

***Example* 3**
Represent the eight rules in Figure 4 with the 3-partite graph in Figure 6.

Each of the three candidate rule sets in Figure 6 contains rules with the same outcomes. For example, the candidate rule set 1 includes the rules R1, R4, and R7 from Figure 4. The arrows in Figure 4 symbolize relationships (e.g. distances) among the rules belonging to different candidate rule sets. In fact these relationships can be of different types thus leading to a hypergraph (Berge 1973 and Shi 1992).

Based on the *m*-partite graph (or hypergraph) representation of the rule-structuring problem different models and algorithms can be developed.

The rule-structuring problem could be loosely formulated as follows:
Minimize the total distance between any two rules belonging to different nodes of the *m*-partite graph subject to constraints, for example, limiting the number of rules selected from one node, limiting the number of clusters.
The rule-structuring algorithm presented in Kusiak (2000a) will be illustrated with a medical data set.

3.3 Decision Tables, Maps, and Atlases

A decision table is a collection of knowledge needed to make decisions in a particular domain. It generalizes the structured matrix introduced in Figure 5 by:

- ❑ Including decision rules generated by different learning algorithms;
- ❑ Transformed rules that may combine different decision rules, and so on;
- ❑ Content organization determined by the rule-structuring algorithm.

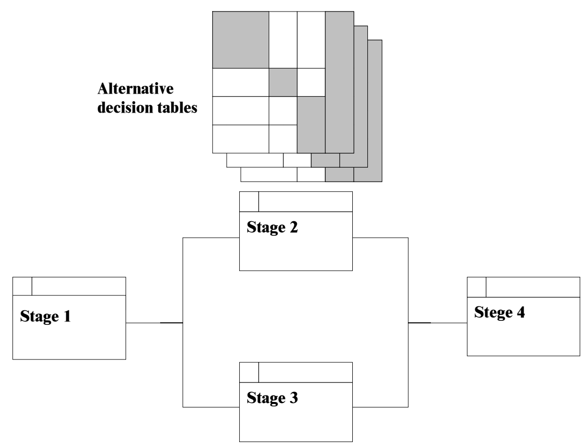

Figure 7. Simple decision map.

The actual decision can be made using one or more decision tables at any stage of the decision map (see Figure 7). Note that some decision tables may reduce to functions, neural network models, etc.

The collection of decision tables distributed over all decision stages constitutes a decision map (Figure 7). Maps in turn can be combined in an atlas and multiple atlases make up a library, etc. There are two primary reasons for alternative decision tables. One is that a decision may follow one or more paths (see Figure 17) contained in one of those tables. The second is due to increased user's transparency in the results generated from independent tables. The notion of independence has a profound impact on decision accuracy and user's confidence in the algorithmically generated result.

4. SOLITARY PULMONARY NODULE DATA SET

The data set studied involves predicting diagnosis of patients with lung abnormalities (possibly cancerous), known in the medical literature as Solitary Pulmonary Nodules (SPNs). The diagnosis is perceived to depend on many features such as the

patient's age, smoking history, overall prevalence of malignancy within the population, SPN diameter, border character, presence of calcification, and results of CT densitometry (Lillington 1993). It is difficult for a human decision-maker to determine which of these data points, if any, are critical for the correct diagnosis, treatment, and patient's survival. Physicians have difficulty in differentiating benign from malignant cases based on SPN clinical features (Khourini *et al.* 1987). In the end, considering the uncertainty and the probability of malignancy, biopsies are often performed on SPNs in spite of the clinical information gathered. Approximately 50%-60% of SPNs are benign and could have been monitored clinically (Gupta *et al.* 1996, Hubner 1996). This process adds direct costs and risks to our healthcare system, and indirect costs in terms of patient time and productivity losses. Therefore, there is a real need to improve the diagnostic accuracy, treatment selection, and outcome prediction of patients with SPNs.
Several procedures including chest radiography, chest-computed tomography (CT), and bronchoscopy have been further developed and are used in an attempt to differentiate benign from malignant nodules.

A small subset of the SPN database for 50 patients (objects), 18 features and the decision D has been selected for analysis (see the Appendix). The list of all features for this data set is provided in (Kusiak *et al.* 2000).

The decision rules extracted from the SPN data set are presented the following format:

`IF (Condition) THEN (Outcome);` [Rule support, Relative rule strength, Discrimination level] [Objects supporting the outcome] (for the definition of each term see Stefanowski 1998).

The two rule sets 1 and 2 generated from the data set in the Appendix are shown in Figures 8 and 10. The first rules set is represented as the rule-feature matrix in Figure 9, while the rule-feature matrix for the second rule set is shown in Figure 11.

```
Rule R1-1. IF (A3 < 5.3) AND (A12 < 24) AND (A17 >= 25) THEN (D = 0); [5, 50.00%, 100.00%][4,
  19, 26, 40, 43]
Rule R2-1. IF (A2 < 2.75) AND (A7 < 2) AND (A17 in [37, 57)) THEN (D = 0); [3, 30.00%,
  100.00%][3, 33, 47]
Rule R3-1. IF (A2 < 2.75) AND (A6 >= 3) AND (A8 < 1) AND (A12 >= 31) THEN (D = 0); [2,
  20.00%, 100.00%][9, 46]
Rule R4-1. (A3 >= 5.3) AND (A12 in [23, 40)) THEN (D = 1); [25, 62.50%, 100.00%][2, 6, 7, 8,
  10, 11, 14, 15, 17, 18, 20, 22, 23, 25, 27, 28, 30, 34, 35, 36, 38, 41, 42, 44, 49]
Rule R5-1. (IF A2 < 1.1) AND (A12 in [24, 31)) THEN (D = 1); [5, 12.50%, 100.00%][1, 12, 24,
  29, 39]
Rule R6-1. IF (A2 in [1.1, 2.1)) AND (A9 < 1) AND (A17 < 82) THEN (D = 1); [6, 15.00%,
  100.00%][5, 14, 18, 21, 31, 32]
Rule R7-1. IF (A2 >= 2.75) THEN (D = 1); [15, 37.50%, 100.00%][6, 7, 11, 15, 17, 20, 27, 28,
  34, 35, 36, 37, 38, 44, 48]
Rule R8-1. IF (A17 < 25) THEN (D = 1); [10, 25.00%, 100.00%][7, 13, 16, 18, 22, 25, 27, 32,
  35, 45]
Rule R9-9. IF (A2 < 0.65) THEN (D = 1); [1, 2.50%, 100.00%][50]
```

Figure 8. Rule set 1.

Rule	A2	A3	A6	A7	A8	A9	A17	Decision	Support
R1-1		<5.3					>=25	0	5
R2-1	<2.75			<2			[37, 57]	0	3
R3-1	<2.75		>=3		<1			0	2
R4		>=5.3						1	25
R5	<=1.1							1	5
R6	[1.1,2.1]					<1		1	6
R7	>=2.75							1	15
R8							<25	1	10
R9-1	<.65							1	1

Figure 9. Unstructured rule-feature matrix for the rule set 1 of Figure 8.

B.V. Dasarathy (Ed.), *Proceedings of the SPIE Conference on Data Mining and Knowledge Discovery: Theory, Tools, and Technology III*, Vol. 4384, SPIE, Orlando, FL, April 2001, pp. 267-277.

```
Rule R1-2. IF (A1 >= 46) AND (A11 < 32) AND (A15 >= 87) AND (A16 >= 94) THEN (D = 0); [6,
  60.00%, 100.00%][9, 19, 33, 40, 43, 47]
Rule R2-2. IF (A10 < 2) THEN (D = 0); [5, 5, 50.00%, 100.00%][3, 4, 26, 40, 46]
Rule R3-2. IF (A10 >= 2) AND (A15 < 87) THEN (D = 1); [27, 67.50%, 100.00%][1, 6, 7, 10, 11,
  14, 15, 16, 18, 20, 22, 24, 25, 27, 28, 30, 31, 32, 35, 36, 37, 38, 39, 42, 45, 48, 50]
Rule R4-2. IF (A16 < 115) AND (A18 >= 4.45) THEN (D = 1); [28, 70.00%, 100.00%][1, 2, 6, 7,
  8, 10, 11, 12, 13, 14, 15, 17, 18, 23, 25, 27, 28, 29, 30, 32, 34, 35, 38, 39, 41, 44, 45,
  48]
Rule R5-2. IF (A1 < 46) THEN (D = 1); [4, 10.00%, 100.00%][7, 17, 21, 49]
Rule R6-2. (A1 >= 72) THEN (D = 1); [15, 37.50%, 100.00%][2, 5, 10, 11, 12, 16, 18, 20, 22,
  23, 25, 27, 30, 38, 42]
```

Figure 10. Rule set 2.

Rule	A1	A10	A11	A15	A16	A18	Decision	Support
R1-2	>=46		<32	>=87	>=94		0	6
R2-2		<2					0	5
R3-2		>=2		<87			1	27
R4-2					<115	>=4.45	1	28
R5-2	<46						1	4
R6-2	>72						1	15

Figure 11. Unstructured rule-feature matrix for the rule set 2 of Figure 10.

The two matrices in Figures 9 and 11 are integrated into the matrix in Figure 12, which in turn is transformed by the rule-structuring algorithm (Kusiak 2000a) in the matrix in Figure 13.

Rule	A1	A2	A3	A6	A7	A8	A9	A10	A11	A15	A16	A17	A18	Decision	Support
R1-2	>=46								<32	>=87	>=94			0	6
R2-2								<2						0	5
R3-2								>=2		<87				1	27
R4-2											<115		>=4.45	1	28
R5-2	<46													1	4
R6-2	>72													1	15
R1-1			<5.3									>=25		0	5
R2-1		<2.75			<2							[37, 57]		0	3
R3-1		<2.75		>=3		<1								0	2
R4-1			>=5.3											1	25
R5-1		<=1.1												1	5
R6-1		[1.1,2.1]					<1							1	6
R7-1		>=2.75												1	15
R8-1												<25		1	10
R9-1		<.65												1	1

Figure 12. Unstructured rule-feature matrix for the rule sets 1 and 2 of Figures 8 and 10.

Rule	A1	A11	A15	A10	A16	A18	A3	A17	A6	A7	A8	A9	A2	Decision	Support
R5-2	<46													1	4
R6-2	>72													1	15
R1-2	>=46	<32	>=87		>=94									**0**	6
R3-2			<87	>=2										1	27
R2-2				<2										**0**	5
R4-2					<115	>=4.45								1	28
R4-1							>=5.3							1	25
R1-1							<5.3	>=25						**0**	5
R2-1								[37, 57]		<2			<2.75	0	3
R3-1									>=3		<1		<2.75	0	2
R6-1												<1	[1.1,2.1]	1	6
R7-1													>=2.75	1	15
R9-1													<.65	1	1
R5-1													<=1.1	1	5
R8-1								<25						1	10

Figure 13. Structured rule-feature matrix for the rule sets 1 and 2 of Figures 8 and 10.

The structured matrix in Figure 13 contains valuable information for decion making, for example:

- ❑ No single group of features is associated with the same decision, which may explain low clinical diagnostic accuracy (40% to 60% accuracy according to Gupta *et al.* 1996 and Hubner *et al.* 1996)
- ❑ Same outcome is associated with alternative set of features, e.g., D = 1 is associated with A1 < 46 (Rule 5-2) and A17 < 25 (Rule 8-1)
- ❑ Rule support can be easily associated with the corresponding set of features, e.g., rule support of 28 corresponds to the features A16 and A18.
- ❑ The usage of rules across different rules is clearly visible, e.g., feature A1 is used by three rules R5-2, R6-2, and R1-2.

In addition to the relationships that are derived from the structured matrix in Figure 13 other relationships can be derived from the rules. Consider the rules derived from the data sets 1 and 2 and the corresponding supporting objects shown in Figures 14 and 15.

Rule	Decision	Supporting objects
R1-1	0	4, 19, 26, 40, 43
R2-1	0	3, 33, 47
R3-1	0	9, 46
R4-1	1	2, 6, 7, 8, 10, 11, 14, 15, 17, 18, 20, 22, 23, 25, 27, 28, 30, 34, 35, 36, 38, 41, 42, 44, 49
R5-1	1	1, 12, 24, 29, 39
R6-1	1	5, 14, 18, 21, 31, 32
R7-1	1	6, 7, 11, 15, 17, 20, 27, 28, 34, 35, 36, 37, 38, 44, 48
R8-1	1	7, 13, 16, 18, 22, 25, 27, 32, 35, 45
R9-1	1	50

Figure 14. Supporting objects for the rules in set 1 of Figure 8.

B.V. Dasarathy (Ed.), *Proceedings of the SPIE Conference on Data Mining and Knowledge Discovery: Theory, Tools, and Technology III*, Vol. 4384, SPIE, Orlando, FL, April 2001, pp. 267-277.

Rule	Decision	Supporting objects
R1-2	0	9, 19, 33, 40, 43, 47
R2-2	0	3, 4, 26, 40, 46
R3-2	1	1, 6, 7, 10, 11, 14, 15, 16, 18, 20, 22, 24, 25, 27, 28, 30, 31, 32, 35, 36, 37, 38, 39, 42, 45, 48, 50
R4-2	1	1, 2, 6, 7, 8, 10, 11, 12, 13, 14, 15, 17, 18, 23, 25, 27, 28, 29, 30, 32, 34, 35, 38, 39, 41, 44, 45, 48
R5-2	1	7, 17, 21, 49
R6-2	1	2, 5, 10, 11, 12, 16, 18, 20, 22, 23, 25, 27, 30, 38, 42

Figure 15. Supporting objects for the rules in set 2 of Figure 10.

The relationships between the rules in the two sets (measured by the inclusion of the supporting objects) is quite obvious and is illustrated in the bipartite graph in Figure 16, where a clear partition of the graph by the value of decion D is visible. The rules R1-1, R2-1 and R3-1 relate to the rules R1-2 and R2-2, all with D = 0. No relationships among rules with different values of D exist.

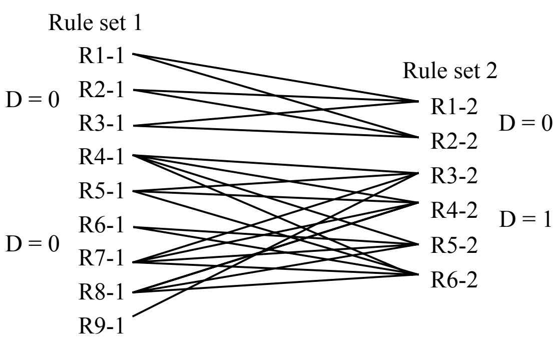

Figure 16. Bipartite graph representing the relationships between the rules in sets 1 and 2.

An example relationship between the three rules R1-1, R1-2, and R2-2 defined on the supporting objects is shown in Figure 17 (see also the shaded are in the body of the matrix in Figure 13). This relationship is expressed as a decision path, which may become a component of a decision process, e.g., a disease management process.

The objects supporting rule R1-1 are shared by two rules R1-2 and R2-2 derived from an alternative set of features. An object with an unknown outcome that would match the conditions of one of the two paths {R1-1, R1-2} or {R1-1, R2-2} would be assigned decion D = 0 with greater user's confidence than in the case of a singular rule made decision.

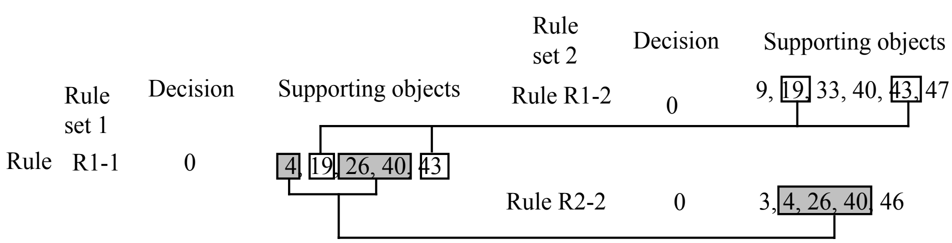

Figure 17. Example decision path involving three rules.

The structured matrix in Figure 13 provides additional information regarding the relationship of the three rules with the remaining decion rules (see the shaded area in the body of the matrix in Figure 13).

5. CONCLUSION

The role of decomposition in managing complexity of data mining tasks and modeling hybrid systems was discussed. The concept of a structured rule-feature matrix was applied to a medical data set. The structured rule-feature matrix becomes the backbone of a system for informed decion-making. Such a system can be implemented as a decision table, a decision map, or a decision atlas. The ideas presented were illustrated with examples and a solitary pulmonary nodule data set.

B.V. Dasarathy (Ed.), *Proceedings of the SPIE Conference on Data Mining and Knowledge Discovery: Theory, Tools, and Technology III*, Vol. 4384, SPIE, Orlando, FL, April 2001, pp. 267-277.

REFERENCES

Anderberg, M.R. (1973), *Cluster Analysis for Applications*, Academic Press, New York.

Bazan, J.G. (1998), A Comparison of Dynamic and non-dynamic rough set methods for extracting laws from decision tables, in L. Polkowski and A. Skowron (Eds), *Rough Sets in Knowledge Discovery* 1: *Methodology and Applications*, Springer-Verlag, New York, pp. 322-365.

Berge, C. (1973), *Graphs and Hypergraphs*, North Holland, Amsterdam, The Netherlands.

Clark, P. and R. Boswell (1989), The CN2 induction algorithm, *Machine Learning*, Vol. 3, No. 4, pp. 261-283.

Friedman, J., Y. Yun and R. Kohavi (1996), Lazy decision trees, *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, AAAI Press and MIT Press.

Grossman, R., S. Kasif, R. Moore, D. Rocke, and J. Ullman (1999), Data Mining Research: Opportunities and Challenges. *Report of Three NSF Workshops on Mining Large, Massive, and Distributed Data*. http://www.ncdm.uic.edu/M3D-final-report.htm

Grzymala-Busse, J.W. (1997), A new version of the rule induction system LERS, *Fundamenta Informaticae*, Vol. 31, pp. 27-39.

Guo, Y. and J. Sutiwaraphun (1988), Knowledge Probing in Distributed Data Mining, *Fourth International Conference on Knowledge Discovery and Data Mining*. http://www.eecs.wsu.edu/~hillol/kdd98ws.html

Gupta, N.C., J. Maloof, and E. Gunel (1996), Probability of malignancy in solitary pulmonary nodules using flourine-18-FDG and PET, *Journal of Nuclear Medicine*, Vol. 37, No. 6, pp. 943-948.

Hubner, K.F., E. Buonocore, H.R. Gould, J. Thie, G.T. Smith, S. Stephens, and J. Dickey (1996), Differentiating benign from malignant lung lesions using "quantitative" parameters of FDG-PET images, *Clinical Nuclear Medicine*, Vol. 21, No. 12, pp. 941-949.

Khourini, H.F., M.A. Meziane, E.A. Zerhouni *et al.* (1987), The solitary pulmonary nodule: assessment, diagnosis and management, *Chest*, Vol. 91, p. 128.

Kusiak, A. (2000), *Computational Intelligence in Design and Manufacturing*, John Wiley, New York.

Kusiak, A. (2000a), Decomposition in data mining: An industrial case study, *IEEE Transactions on Electronics Packaging Manufacturing*, Vol. 23, No. 4, pp. 345-353.

Kusiak, A. (2000b), Evolutionary computation in process modeling and programming, *Robotica,* Vol. 40, No. 3, pp. 15-19.

Kusiak, A., J.A. Kern, K.H. Kernstine, and T.L. Tseng (2000), Autonomous Decision-Making: A Data Mining Approach, *IEEE Transactions on Information Technology in Biomedicine*, Vol. 4, No. 4, pp.274-284.

Lillington G.A. (1993), Management of the solitary pulmonary nodule, *Hospital Practice*, May, pp. 41-48.

Michalski, R.S., I. Mozetic, J. Hong, and N. Lavrac (1986), The multi-purpose incremental learning system AQ15 and its testing application to three medical domains, *Proceedings of the 5th National Conference on Artificial Intelligence*, AAAI Press, Palo Alto, CA, pp. 1041-1045.

Pawlak Z. (1982), Rough sets, *International Journal of Information and Computer Science*, Vol. 11, No. 5, pp. 341-356.

Shi, C.-J. (1992), A signed hypergraph model of constrained via minimization, *Microelectronics Journal*, Vol. 23, pp. 533-554.

Stefanowski, J. (1998), On rough set approaches to induction of decision rules, in L. Polkowski and A. Skowron (Eds), *Rough Sets in Knowledge Discovery* 1: *Methodology and Applications*, Springer-Verlag, New York, pp. 501-529.

Tsumoto, S. (2000), Automated discovery of positive and negative knowledge in clinical databases, *IEEE Engineering in Medicine and Biology*, Vol. 19, No. 4, pp. 56-62.

Zakarian, A. and A. Kusiak (2000), Analysis of Process Models, *IEEE Transactions on Electronics Packaging Manufacturing*, Vol. 23, No. 2, pp. 137- 147.

Quinlan, J.R. (1986), Induction of decision trees, *Machine Learning,* Vol. 1, No 1, pp. 81-106.

Zaki, M.J. and C.-T. Ho (Eds) (2000), *Large-Scale Parallel Data Mining,* Springer-Verlag, New York.

Zaki, M.J., C.-T. Ho, and R. Agrawal (1999), Scalable parallel classification for data mining on shared-memory multiprocessors, *IEEE International Conference on Data Engineering*, Sydney, Australia, pp. 198-205. http://www.cs.rpi.edu/~zaki/papers.html#WKDD99

APPENDIX: Solitary Pulmonary Nodule Data Set

A0	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13	A14	A15	A16	A17	A18	D
1	68	1	3.1	1	3	2	3	1	1	2	150	27	0	104	79	97	79	12	1
2	73	1.5	5.7	1	0	3	1	1	0	2	5	37	1	99	108	92	91	9.4	1
3	57	1.2	3.8	3	0	3	0	0	1	1	20	24	0	90	58	12	41	0	0
4	71	0.8	1.8	1	0	2	0	0	1	1	20	23	0	55	118	78	25	0	0
5	85	1.6	6.2	2	0	3	4	0	0	2	140	19	0	60	100	61	45	3.4	1
6	65	3	28	4	0	2	0	0	0	2	60	30	0	87	71	103	34	11	1
7	39	4	39	2	0	1	0	0	1	2	40	37	0	33	41	25	18	10	1
8	57	1.8	10	1	0	2	3	0	0	2	30	26	1	97	104	85	98	11	1
9	49	2.5	20	1	0	3	0	0	1	2	30	42	0	104	106	106	91	3.6	0
10	72	1.5	7.1	2	0	2	0	0	1	2	20	30	0	64	71	113	31	4.6	1
11	77	3	20	2	0	3	0	0	1	2	50	26	1	116	45	107	80	5.5	1
12	73	1	3.1	2	0	3	0	1	1	2	0	25	1	96	113	91	67	9.1	1
13	62	2	13	1	0	2	2	1	1	2	100	21	1	50	89	68	22	10	1
14	64	2	13	2	0	2	2	1	0	2	60	23	1	75	58	82	37	5.1	1
15	71	4	50	2	0	1	1	0	0	2	100	39	0	94	62	88	73	4.6	1
16	76	0.8	2	3	0	2	4	0	0	2	25	20	1	76	69	112	16	1.2	1
17	34	4	39	2	0	3	0	0	0	2	0	23	1	104	117	102	92	18	1
18	72	2	9.6	2	0	2	1	1	0	2	30	27	0	37	59	43	20	9.3	1
19	70	1.5	4.9	2	0	3	0	0	0	2	0	20	0	109	109	102	84	4.3	0
20	74	3	13	2	0	2	0	0	1	2	20	27	0	106	86	92	87	2.3	1
21	34	2	9.6	3	0	1	0	0	0	2	0	19	1	99	96	101	78	1.8	1
22	72	2	7.1	4	0	2	3	1	1	2	100	26	0	56	56	77	24	3.5	1
23	73	2	7.1	4	0	3	0	1	1	2	30	24	0	68	90	59	66	9.9	1
24	70	1	3.1	1	0	3	0	0	0	2	0	29	0	112	84	109	77	2.6	1
25	75	2	13	2	0	3	0	1	1	2	24	26	0	32	56	57	15	9.2	1
26	55	1	3.1	2	0	2	2	1	0	1	70	22	1	79	77	97	31	0	0
27	79	3	20	3	0	2	0	0	1	2	35	33	1	44	61	61	24	6.4	1
28	61	5	79	2	0	1	0	0	0	2	0	29	1	74	58	63	74	9.5	1
29	69	1	3.1	1	0	3	4	0	1	2	60	24	0	93	88	99	50	14	1
30	72	2	13	4	0	2	1	0	1	2	50	24	0	102	78	80	31	7.1	1
31	69	1.2	3.8	2	0	2	0	0	0	2	50	32	0	74	51	92	25	2.8	1
32	64	1.5	4.9	3	0	1	0	0	0	2	75	37	0	53	63	80	21	6.1	1
33	64	1	3.1	3	0	3	0	1	1	2	0	31	1	104	115	116	56	7.8	0
34	49	5	50	1	0	1	1	0	1	2	30	27	0	75	114	69	65	6.6	1
35	64	4	39	1	0	3	1	1	0	2	92	25	0	70	80	88	24	18	1
36	66	3.5	33	4	0	3	2	0	1	2	81	37	0	88	67	83	59	3.6	1
37	66	6	95	2	0	2	0	0	0	2	50	20	0	90	52	98	59	3.3	1
38	76	5	50	1	0	1	1	1	0	2	60	30	1	87	79	87	79	7.1	1
39	62	1	3.1	3	0	3	0	0	0	2	40	28	0	79	61	93	32	18	1
40	56	1.5	4.9	1	0	2	0	0	0	1	0	21	1	109	105	107	85	0	0
41	68	2.5	20	2	1	3	0	1	1	2	60	26	0	115	136	114	76	6.3	1
42	72	2.5	20	3	0	3	2	0	1	2	30	28	1	56	59	57	36	3.2	1
43	53	1	3.1	3	0	3	3	1	0	2	15	19	0	88	87	94	52	4	0
44	52	3.2	30	2	0	2	0	0	1	2	35	25	1	92	109	105	46	7.6	1
45	71	1	3.1	3	0	3	0	1	0	2	50	22	0	52	48	81	15	11	1
46	54	1	3.1	2	0	3	1	0	0	1	41	32	0	44	81	44	28	0	0
47	47	2.2	11	4	0	2	0	1	0	2	30	22	1	81	89	100	37	2.4	0
48	70	3	28	4	0	1	0	1	1	2	90	20	0	89	82	90	34	14	1
49	45	2	9.6	1	0	2	0	0	0	2	11	23	0	121	128	124	86	2.2	1
50	64	0.5	0.8	1	0	3	1	1	0	2	90	46	0	75	69	84	33	4.2	1

B.V. Dasarathy (Ed.), *Proceedings of the SPIE Conference on Data Mining and Knowledge Discovery: Theory, Tools, and Technology III*, Vol. 4384, SPIE, Orlando, FL, April 2001, pp. 267-277.