# STRUCTURAL PATTERN RECOGNITION, ARRANGEMENTS AND THEORY OF COVERS

Robert M. Haralick
Departments of Electrical Engineering and Computer Science
University of Kansas
Lawrence, Kansas 66045

## Abstract

This paper discusses a structural pattern recognition methodology which combines some ideas about relation homomorphisms and theory of covers. We take as a basic pattern a labeled N-ary relation which we call an arrangement. Features from any arrangement are determined by calculating to which basis arrangements the given arrangement is a homomorphism and calculating which basis arrangements are isomorphic to some part of the given arrangement. A decision rule then decides which class the given arrangement is assigned using the theory of covers. The methodology suggested in the paper provides an alternative to syntactic pattern recognition.

## I. Introduction

Statistical pattern recognition uses the n-tuple for its basic data structure. Each pattern must be an ordered list of the same length. Syntactic pattern recognition uses a grammar to describe the relationships within a string which tell whether the string belongs to a particular category. The length of the pattern string can be arbitrary and if a parser for the grammar at each step finds the string legal, then the string belongs to the category associated with the grammar.

Much work has been done to generalize the classical concept of grammar and the string on which its parser operates. We now have web grammars, tree grammars, fuzzy grammars and stochastic grammars (Fu, 1974; Pfaltz and Rosenfeld, 1969). However, the power of the grammar, which is its recursiveness, all too often is its weakness. The structural pattern recognition methodology discussed in this paper is one approach to generalizing the basic pattern data structure in a way which seems to permit flexibility without the necessity of recursiveness.

The data structure we suggest for a pattern is a labeled N-ary relation which we call an arrangement. An arrangement is quite rich in structural variety. To use an arrangement as a pattern, we must discover the natural mathematical operations on arrangements. For the n-tuple in a vector space the natural opeation is to compare it to some other prototype vector by a distance function and convert the distance measurement to a probability or density. The arrangement, however, has no meaningful and convenient metric space in which it is a point so we must look elsewhere.

The N-ary relation is basically an algebraic structure and the natural mathematics or algebraic structures involves homomorphisms. Thus, features are arrangements, patterns are arrangements and the value of a feature for a given pattern tells whether the pattern is homomorphic to the feature, whether the pattern has a copy of the feature, or whether the feature is a homomorphic image of the pattern. The feature extraction mechanism then converts the arrangement pattern to an n-tuple which is classified by a decision rule using a covering methodology. The theory of covers shows how a collection of sets can be constructed so that the sets in the collection cover the training subset $S_0$ and do not cover any measurement in a subset $S_1$ which is contained in $S_0$ complement. The sets in the collection have a simple description so that a decision rule using the description of the sets in the cover have an easily implementable and simple form.

Section II of the paper describes the concept of the arrangement and arrangement homomorphisms Section III describes a feature extractor which uses these concepts to translate an arrangement to an n-tuple. Sections IV and V of the paper review how decision rules can be constructed using the covering methodology.

## II. The Arrangement

Let A be the set of elements whose arrangement is being described. Each group of related elements from A is given a label from the label set L. Let R be the labeled N-ary relation which consists of labeled N-tuples of elements from A.

Definition 1. A simple order-N arrangement is a triple (R, A, L) where $R \subseteq A^N \times L$.

Definition 2. A general arrangement is a set of simple arrangements, each simple arrangement being of different order, being defined on the same set, and having the same label set. If there are K simple arrangements in the arrangement $\mathcal{A}$, then we write

$\mathcal{A} = \{R_1, R_2, \ldots, R_K;\ A,\ L\}$ where

$R_k \subseteq A^{N_k} \times L, \quad k = 1, \ldots, K.$

Definition 3. Let $\mathcal{A} = \{R_1, \ldots, R_K;\ A,\ L\}$ be an arrangement and $H \subseteq A \times B$. The composition of arrangement $\mathcal{A}$ with H results in an arrangement $\mathcal{B}$ which we define as

$$\mathcal{A} \circ H = \mathcal{B} = \{S_1, S_2, \ldots, S_K;\ B,\ L\}, \text{ where}$$

$$S_k = \{(b_1, b_2, \ldots, b_{N_k}, \ell) \mid (a_1, \ldots, a_{N_k}, \ell) \in R_k \cdot (a_n, b_n) \in H,\ n = 1, \ldots, N_k\}$$

Definition 4. An arrangement $\mathcal{A} = \{R_1, \ldots R_K;\ A,\ L\}$ is contained in an arrangement $\mathcal{D} = \{T_1, \ldots, T_K;\ A,\ L\}$ if and only if $R_k \subseteq T_k$, $k = 1, \ldots, K$. In this case we write

$A \subseteq \mathcal{D}$.

Definition 5. Two arrangements $\mathcal{A} = \{R_1,\ldots,R_K;\ A,\ L\}$ and $\mathcal{B} = \{S_1,\ldots,S_N,\ B,M\}$ are comparable if the number of relations in each arrangement is the same (K=N) the label sets are the same (M=L), and the relation $R_k$ has the same order as the relation $S_k$: ($R_k \subseteq A^{N_k} \times L$ and $S_k \subseteq B^{N_k} \times L$).

Definition 6. Let $\mathcal{A} = \{R_1,\ldots,R_K;\ A,L\}$ and $\mathcal{B} = \{S_1,\ldots,S_K;\ B,L\}$ be two comparable arrangements. Let H: $A \to B$. The function H is a homomorphism from arrangement $\mathcal{A}$ to arrangement $\mathcal{B}$ if and only if $\mathcal{A} \circ H \subseteq \mathcal{B}$.

## III. An Arrangement Feature Extractor Example

In this section we give a structural pattern recognition method using the arrangement concept and theory of covers. Given are the arrangement features, one training pattern from class 0 and one training pattern from class 1. The first problem will be to determine a decision rule and the second problem will be to assign a class to a new test pattern.

The arrangement features and the training arrangements are shown in Figure 1. To keep the example simple, only simple order 2 arrangments are used. There are five arrangement features: $A_1, A_2, A_3, A_4$, and $A_5$. Arrangement $T_0$ is the training arrangement for class 0 and $T_1$ is the training arrangement for class 1. Each training arrangement uses the label set $L = \{0, 1\}$ and is defined on the set $S = \{a,b,c,d,e,f\}$.

Thus, $T_0 \subseteq S \times S \times L$ and $T_1 \subseteq S \times S \times L$.

The feature arrangements use the label set L but are defined on different sets $B = \{v,w\}$, $C = \{x,y,z\}$, $D = \{\alpha,\beta,\delta\}$, $E = \{p,q,r\}$, and $\{s,t,u\}$ for feature arrangements $A_1$ through $A_5$.

We will use a feature extractor which asks only two kinds of questions: is there a copy of the feature arrangement in the training arrangement and is the feature arrangement a homomorphic image of the training arrangement. The first question asks about the existence of a one-to-one homomorphism from the feature arrangement onto its range in the training relation. The second question asks about the existence of a homomorphism from the training relation onto the feature arrangement.

Our example feature extractor asks the first question for feature arrangements $A_1$, $A_2$ and $A_4$ and asks the second question for feature arrangements $A_3$ and $A_5$. The algorithms for determining whether a homomorphism (question 1) or partial isomorphism (question 2) exist is given in (Haralick, 1976) and we do not discuss it here. Tables 2 and 3 give the results. From Table 2 we see that class 0 can be distinguished from class 1 simply by whether the arrangement has a copy of feature arrangement $A_3$ or $A_5$. From Table 3, we see that the test arrangement has a copy of feature arrangement $A_3$ so that we would assign it to class 0.

In the next section we discuss in a systematic way a method of constructing decision rules like the one used in this example.

## IV. Pattern Discrimination From The Perspective of Covers

The pattern discrimination problem is concerned with how to construct a decision rule which assigns a unit to a particular category class on the basis of the measurement patterns in the training set. The decision rule which makes the assignment is a function from measurement space E to the set of classes. Each and every element in E is assigned by the decision rule to one and only one class. When such a decision rule is deterministic it determines a partition of E.

Assume that we are given disjoint training subsets $S_0$ and $S_1$ of measurements from the set E. Our main concern will be how to find a cover for $S_0$ which does not include any of $S_1$. Our point of view, however, will not be statistical. We will not assume any probability distributions. Rather, we will concentrate on the simplest way of distinguishing elements of one set from another. This problem is a frequent one in Boolean Switching theory and the algorithms due to Quine and McCluskey are well know (Quine, 1955; McCluskey, 1956; Hill and Peterson, 1968; Rhyne, 1973).

We discuss the natural generalization of the Quine-McCluskey method to the general non-Boolean case where each variable can have K possible values. There is also a natural generalization of the Quine-McCluskey method to take care of don't-care conditions for Boolean variables (Hill and Peterson, 1968). Extension of this case to non-Boolean variables is possible, and we work one such example. We are particularly interested in the case where the variables are not Boolean, where there are perhaps more don't-care conditions than specified conditions, and where there are so many variables that working the problem bottom up via Quine-McCluskey requires too many operations. Following Michalski (1969; 1971; 1973; 1974) we will work the problem top down.

We begin our discussion with Cartesian products and projection operators.

Definition 7. Let $J = \{j_1, j_2, \ldots, j_N\}$ be a linearly ordered finite set whose elements satisfy $j_n < j_{n+1}$, $n = 1,\ldots,N-1$. Then,

$\chi_{j \in J} D_j = D_{j_1} \times D_{j_2} \times \ldots \times D_{j_N}$ where x denotes the Cartesian product.

Definition 8. Let $S \subseteq \chi_{i \in I} D_i$ and $J \subseteq I$. The projection of S into $\chi_{j\varepsilon J} D_j$ is defined by

$$\Delta_J S = \{(x_j: j\varepsilon J) \varepsilon \chi_{j\varepsilon J} D_j \mid \text{for some}$$
$$(y_1,\ldots,y_n) \varepsilon S,\ y_j = x_j,\ j\varepsilon J\}$$

Definition 9. Let $T \subseteq \chi_{j\varepsilon J} D_j$ and $J \subseteq I$. The inverse projection of T into $\chi_{i\varepsilon I} D_j$ is defined by

$$\Delta_J^{-1} T = \{(y_i: i\varepsilon I) \varepsilon \chi_{i\varepsilon I} D_i \mid \text{for some}$$
$$(x_j: j\varepsilon J) \varepsilon T,\ y_j = x_j,\ j\varepsilon J\}$$

The set $\Delta_J^{-1} T$ is called a cylinder set having T for its base. The order of the cylinder set is #J. Figure 2 illustrates a couple of order 1 cylinder sets.

Ashby (1964) uses the intersection of the inverse projection of cylinder sets to determine simpler descriptions of complex sets. We modify Ashby's intersection method to a union method and use this with Michalski's idea of set covers (Michalski, 1969, 1971, 1973, 1974) to generate a decision rule which distinguishes one set from another.

## V. Covers

Definition 10. A cover of $S_0$ against $S_1$ is any set $C$ of cylinder sets satisfying $S_0 \subseteq \bigcup_{L\varepsilon C} L \subseteq S_1^c$, where $S_1^c$ means the complement of set $S_1$.

The definition implies that if a collection of cylinder sets is to be a cover, its set theoretic union must completely cover the set $S_0$ and it must not cover any of the set $S_1$.

Definition 11. An order n cover of $S_0$ against $S_1$ is any collection $C$ of cylinder sets satisfying

1. $\bigcup_{L\varepsilon C} L \supseteq S_0$
2. $\bigcup_{L\varepsilon C} L \subseteq S_1^c$
3. $L\varepsilon C$ implies the order of L is less than or equal to n.

Theorem 1. Let $C_n = \{L \mid L$ is a cylinder set of order $\leq n$, $L \cap S_1 = \emptyset$ and $L \cap S_0 \neq \emptyset\}$

If $S_0 \subseteq \bigcup_{L\varepsilon C_n} L$ then $C_n$ is a cover of $S_0$ against $S_1$.

Proof: By hypothesis, $\bigcup_{L\varepsilon C} L \subseteq S_0$ and order of $L\varepsilon C_n$ is less than or equal to n. Since $L\varepsilon C_n$ implies $L \cap S_1 = \emptyset$, we must have $L \subseteq S_1^c$. But this is true for all $L\varepsilon C_n$. Hence $\bigcup_{L\varepsilon C_n} L \subseteq S_1^c$. Therefore, $C_n$ is a cover of $S_0$ against $S_1$.

The theorem enables us henceforth to confine our attention to the cylinder sets which satisfy conditions $L \cap S_1 = \emptyset$ and $L \cap S_0 \neq \emptyset$ in selecting a cover.

Example 1. (Michalski, 1972) Assume $S_1$ and $S_0$ are disjoint. Find a simple rule for distinguishing patterns of the first set from the patterns of the second where measurement space $E = D_1 x D_2 x D_3 x D_4$ and the domains are

$D_1 = \{x, y, z\}$

$D_2 = \{1, 2, 3\}$

$D_3 = \{A, B, C, D\}$

$D_4 = \{\alpha, \beta, \gamma\}$

and $S_0$ and $S_1$ are defined by

$S_0 = \{z3C\alpha, y3D\gamma, z3C\beta, y3B\beta\}$

$S_1 = \{z3D\alpha, z3B\gamma, z3D\beta, z3A\beta\}$

The problem is that of finding a cover of $S_0$ against $S_1$, i.e., to find $C_n = \{L \mid L$ is a cylinder set of order $\leq n$,

$L \cap S_1 = \emptyset$ and $L \cap S_0 \neq \emptyset\}$

The order 1 cylinder sets for $S_0$ are*

$\Delta_1^{-1}\Delta_1 S_0 = \{z***, y***\}$

$\Delta_2^{-1}\Delta_2 S_0 = \{*3**\}$

$\Delta_3^{-1}\Delta_3 S_0 = \{**C*, **D*, **B*\}$

$\Delta_4^{-1}\Delta_4 S_0 = \{***\alpha, ***\beta, ***\gamma\}$

The order 1 cylinder sets for $S_1$ are

$\Delta_1^{-1}\Delta_1 S_1 = \{z***\}$

$\Delta_2^{-1}\Delta_2 S_1 = \{*3**\}$

$\Delta_3^{-1}\Delta_3 S_1 = \{**D*, **B*, **A*\}$

$\Delta_4^{-1}\Delta_4 S_1 = \{***\alpha, ***\beta, ***\gamma\}$

---

*means any value of that component is allowed.

By theorem 1 we find that

$$C_1 = \{y***\}, \{**C*\}$$

and

$$S_0 \subseteq \bigcup_{L \varepsilon C_1} L = \{y***\} \cup \{**C*\}$$

The decision rule is: assign $(x_1, x_2, x_3, x_4)$ to class 0 when $x_1 = y$ or $x_3 = C$.

The set of all 4 tuples assigned to class 0 can be easily illustrated with the use of the Karnaugh map as shown in Figure 3.

If the relationship between the sets $S_0$ and $S_1$ were more complex so that we could not tell the difference between them from cylinder sets of order 1, then we would have to distinguish them using cylinder sets of higher order.

A minimal cylinder set covering is a difficult problem. In the remainder of this section we define what a minimal cover is and give a procedure to find a suboptimal cover.

Definition 12. A minimal order n cover of $S_0$ against $S_1$ is any collection $C$ of cylinder sets satisfying

1. $C$ is an order n cover of $S_0$ against $S_1$
2. $C'$ is an order n cover of $S_0$ against $S_1$ and $C' \supseteq C$ implies $C' = C$

To determine a minimal set of cylinders, a cylinder set table must be constructed. The table lists all the measurements that belong to training set $S_0$ across its top and lists the candidate cylinder sets in $C$ down its left side. An X is placed at the intersection of each row and column if the cylinder contains the measurement patterns. For example 1, the complete cylinder table for training set $S_0$ is shown in Figure 4.

The selection process begins with the identification of all cylinder sets which are the only ones to cover a particular measurement pattern. The corresponding X's in the table can then be circled. The cylinder sets having any circled X's in their cover must be included in any cover $C$ of $S_0$ against $S_1$. All the measurements that are now covered by the initial selection of cylinders can then be removed from the table. If there are no measurement patterns left then the cover is complete. For the given cylinder set table, the cylinder sets y*** and **C* constitute a minimal cover.

In many cases, however, the removal of the initially selected cylinders leaves a reduced cylinder set table that has two or more X's in every column. This indicates that there will be more than one set of cylinder sets that will cover the remaining measurement pattern. The decision regarding which cylinder to choose must be based on some criterion other than necessity. One algorithm is to choose one cylinder set covering the most measurement patterns and reduce the table and iterate until all events have been covered.

We will now give one example of the application of cylinder set covers to pattern recognition.

Example 2. Consider two sets of 4-tuples shown below.

| $S_0$ | $S_1$ |
|---|---|
| 0 0 0 1 | 0 1 2 2 |
| 0 1 1 1 | 1 1 2 0 |
| 1 0 2 1 | 1 0 1 0 |
| 1 1 2 1 | 1 0 1 1 |
| 2 0 1 1 | 2 1 2 1 |
| 2 1 1 2 | 2 1 2 2 |

The problem is that of finding a cover $C_n$ of $S_0$ against $S_1$. To find $C_n = \{L|L$ is a cylinder set of order $\leq n$, $L \cap S_1 = \emptyset$ and $L \cap S_0 \neq \emptyset\}$.

The order 1 cylinder sets for training set $S_0$ are

$$\Delta_1^{-1}\ \Delta_1 S_o = \{0***,\ 1***,\ 2***\}$$

$$\Delta_2^{-1}\ \Delta_2 S_o = \{*0**,\ *1**\}$$

$$\Delta_3^{-1}\ \Delta_3 S_o = \{**1*,\ **2*\}$$

$$\Delta_4^{-1}\ \Delta_4 S_o = \{***0,\ ***1,\ ***2\}$$

The order of 1 cylinder sets for training set $S_1$ are

$$\Delta_1^{-1}\ \Delta_1 S_1 = \{0***,\ 1***,\ 2***\}$$

$$\Delta_2^{-1}\ \Delta_2 S_1 = \{*0**,\ *1**\}$$

$$\Delta_3^{-1}\ \Delta_3 S_1 = \{**1*,\ **2*\}$$

$$\Delta_4^{-1}\ \Delta_4 S_1 = \{***0,\ ***1,\ ***2\}$$

It is obvious that any order 1 cylinder set of $S_0$ includes patterns from set $S_1$ and vice versa. Thus we must look at order 2 cylinder sets.

The order 2 cylinder sets for training set $S_0$ are

$\Delta_{12}^{-1}\ \Delta_{12}S_o$ = {00**, 01**, 10**, 11**, 20**, 21**}

$\Delta_{13}^{-1}\ \Delta_{13}S_o$ = {0*1*, 1*2*, 2*1*}

$\Delta_{14}^{-1}\ \Delta_{14}S_o$ = {0**0, 0**1, 1**1, 2**1, 2**2}

$\Delta_{23}^{-1}\ \Delta_{23}S_o$ = {*01*, *11*, *02*, *12*}

$\Delta_{24}^{-1}\ \Delta_{24}S_o$ = {*0*0, *1*1, *0*1, *1*2}

$\Delta_{34}^{-1}\ \Delta_{34}S_o$ = {**10, **11, **21, **12}

The order 2 cylinder sets for training set $S_1$ are

$\Delta_{12}^{-1}\ \Delta_{12}S_1$ = {01**, 11**, 10**, 21**}

$\Delta_{13}^{-1}\ \Delta_{13}S_1$ = {0*2*, 1*2*, 1*1*, 2*2*}

$\Delta_{14}^{-1}\ \Delta_{14}S_1$ = {0**2, 1**0, 1**1, 2**1}

$\Delta_{23}^{-1}\ \Delta_{23}S_1$ = {*12*, *01*}

$\Delta_{24}^{-1}\ \Delta_{24}S_1$ = {*1*2, *1*0, *0*0, *0*1, *1*1}

$\Delta_{32}^{-1}\ \Delta_{34}S_1$ = {**22, **20, **10, **11, **21}

The cylinders which satisfy $L \cap S_1 = \emptyset$ and $L \cap S_0 \neq \emptyset$ are

{00**, 20**, 0*1*, 2*1*, 0**0, 0**1, *11*, *02*, **12}

To pick up the minimal set of cylinder sets to cover the training set $S_0$, we construct the cover table (Figure 5). Here we find out that the pattern (1121) is not covered by any one of the order 2 cylinder sets (the condition $S_0 \subseteq \bigcup_{L \varepsilon C_n} L$ is not satisfied). We will solve and discuss this problem later. Now, we will just neglect the event (1121) and consider the rest of events only.

One minimal order 2 set of cylinders is {0*1*, 2*1*, *02*}. They cover all patterns in set $S_0$ except the event (1121).

We still cannot tell the difference between measurement pattern (1121) and the training set $S_1$. So we must look at order 3 cylinders.

The order 3 cylinder sets for the pattern (1121) are

$\Delta_{123}^{-1}\Delta_{123}(1121)$ = {112*}

$\Delta_{124}^{-1}\Delta_{124}(1121)$ = {11*1}

$\Delta_{134}^{-1}\Delta_{134}(1121)$ = {1*21}

$\Delta_{234}^{-1}\Delta_{234}(1121)$ = {*121}

The order 3 cylinder sets for the training set $S_1$ are

$\Delta_{123}^{-1}\Delta_{123}S_1$ = {012*, 112*, 101*, 212*}

$\Delta_{124}^{-1}\Delta_{124}S_1$ = {01*2, 11*0, 10*0, 10*1, 21*1, 21*2}

$\Delta_{134}^{-1}\Delta_{134}S_1$ = {0*22, 1*20, 1*10, 1*11, 2*21, 2*22}

$\Delta_{234}^{-1}\Delta_{234}S_1$ = {*122, *120, *010, *011, *121, *122}

An order 3 cover of the pattern (1121) against $S_1$ is {11*1, 1*21}. Picking either one of them, say 11*1 and we have, by Theorem 1

3 = { 0*1* , 2*1* , *02* , 11*1 }

$$S_o \subseteq \bigcup_{L\varepsilon C_n} L = \{0{*}1{*}\} \cup \{2{*}1{*}\} \cup \{{*}02{*}\} \cup \{11{*}1\}$$

A decision rule for discriminating $S_0$ from $S_1$ thus: assign $(x_1, x_2, x_3, x_4)$ to category 0 if $(x_1=0, x_3=1) \vee (x_1=2, x_3=1) \vee (x_2=0, x_3=2) \vee (x_1=1, x_2=1, x_4=1)$. The example 2 Karnaugh map is shown in Figure 6.

The rule thus found represents a generalization of the set $S_0$ and $S_1$ in the sense that patterns not in $S_0$ and $S_1$ will be assigned respective categories 0 and 1 by the rule. The rule is only one of very many possible generalizations of the original sets.

## IV. Conclusion

We have illustrated how a labeled N-ary relation can be a pattern andhow feature extraction can be done to transform the arrangement pattern to an n-tuple pattern on the basis of whether homomorphisms exist to or from a given set of arrangement features. We then showed how given even a few arrangement patterns a decision rule, using the theory of covers, can be constructed which assign a new test arrangement to a class. The entire procedure from beginning to end is structural in nature and offers an alternative approach to syntactic pattern recognition.

A basic theoretical problem which needs to be solved and which was not discussed in this paper is given training arrangements for different classes, what is the procedure for determining the best basis arrangement features. We will discuss solutions to this problem in a forthcoming paper.

## References


Ashby, W. R., "Constraint Analysis of Many-Dimensional Relations", General System, Vol. 9, pp. 99-105, 1964.

Fu, K. S., Syntactic Methods in Pattern Recognition, Academic Press, New York, 1974.

Haralick, R. M. and Kartus, J., "Theory of Arrangements", University of Kansas, Center for Research, Lawrence, Kansas, June 1976.

Hill, F. J., and G. R. Peterson, "Introduction to Switching Theory and Logical Design", New York: John Wiley & Sons, Inc., 1968, pp. 139-161.

McCluskey, E. J., Jr., "Minimization of Boolean Functions", Bell System Tech. J., Vol. 35, November 1956, pp. 1417-1444.

Michalski, R. S., "On the Quasi-Minimal Solution of the General Covering Problem", Proceedings of the 5th International Symposium, Yugoslavia, Bled, October 8-11, 1969.

Michalski, R. S., and B. H. McCormick, "Interval Generalization of Switching Theory", Proceedings of the Third Annual Houston Conference on Computer and System Science, Houston, Texas, April 26-27, 1971, pp. 213-226.

Michalski, R. S., "A Variable-Valued Logic System as Applied to Picture Description and Recognition", GRAPHIC LANGUAGES, Proceedings of the IFIP Working Conference on Graphic Languages, Vancouver, Canada, May 1972.

Michalski, R. S., "AQVAL/1--Computer Implemental of A Variable-Valued Logic System $VL_1$ and Example of its Application to Pattern Recognition", International Joint Conference on Pattern Recognition, Washington, D. C., October 30 - November 1, 1973, pp. 3-17.

Michalski, R. S., "Variable-Valued Logic: System $VL_1$", Proceedings of the 1974 International Symposium on Multiple-Valued Logic, West Virginia University, Morgantown, W.V., May 29-31, 1974, pp. 323-346.

Quine, W. V., "A Way to Simplify Truth Function", Am. Math. Monthly, Vol. 62, November 1955, pp. 627-631.

Rhyne, V. T., Jr., "Fundamentals of Digital Systems Design", New Jersey: Prentice-Hall, Inc., 1973, pp. 158-163.

Pfaltz, J. L . and Rosenfeld, A., "Web Grammars", Proc. 1st International Joint Conference on Artificial Intelligence, Washington, DC, May 1969, 609-19.

---

Support for this work was provided by Grant AFOSR 77-3307.

Feature Arrangements

| $A_1 \subseteq BxBxL$ | $A_2 \subseteq CxCxL$ | $A_3 \subseteq DxDxL$ |
|---|---|---|
| vv0 | xx0 | αα1 |
| wv0 | yz0 | βδ0 |
| vw1 | zy0 | βα1 |
| ww1 | xy1 | δδ0 |
| | yy1 | δα1 |
| | zx1 | |

| $A_4 \subseteq ExExL$ | $A_5 \subseteq FxFxL$ |
|---|---|
| pp0 | ss0 |
| pq1 | st1 |
| qr0 | tt1 |
| qq1 | us0 |
| rp0 | ut1 |
| rq1 | |

Training Arrangements

| $T_0 \subseteq SxSxL$ | $T_1 \subseteq SxSxL$ |
|---|---|
| aa0 | aa0 |
| ba0 | ba0 |
| ca0 | ce0 |
| da0 | de0 |
| ad1 | ea0 |
| bd1 | fa0 |
| cd1 | ad1 |
| dd1 | bd1 |
| | cd1 |
| | dd1 |
| | ed1 |
| | fd1 |

Figure 1 lists the 5 feature arrangements $A_1$, $A_2$, $A_3$, $A_4$ and $A_5$ and the training arrangements $T_0$ and $T_1$ for class 0 and class 1, respectively.

Feature Arrangement

| | | $A_1$ | $A_2$ | $A_4$ |
|---|---|---|---|---|
| Training Arrangement | $T_0$ | a,b→v<br>c,d→w | no | no |
| | $T_1$ | a,c,e→v<br>b,d,f→w | no | a,b→p<br>c,d→q<br>e,f→r |

Table 1 indicates whether the feature arrangement is a homomorphic image of the training relation. If it is, the homomorphism is given.

Feature Arrangement

| | | $A_3$ | $A_5$ |
|---|---|---|---|
| Training Arrangement | $T_0$ | α→d<br>δ→a<br>β→c | no |
| | $T_1$ | no | s→a<br>t→d<br>u→f |

Table 2 indicates whether the training relation has a copy of the feature arrangement. If it does, then the partial isomorphism is given.

Test Arrangement

| T |
|---|
| aa0 |
| ba0 |
| ce0 |
| de0 |
| ec0 |
| fc0 |
| ad1 |
| bd1 |
| cd1 |
| dd1 |
| eb1 |
| fb1 |

| | $A_1$ | $A_2$ | $A_4$ |
|---|---|---|---|
| T | a,c,e→v<br>b,d,f→w | a,b→x<br>c,d→y<br>e,f→z | no |

Arrangement a homomorphic image?

| | $A_3$ | $A_5$ |
|---|---|---|
| T | α→d<br>β→b<br>δ→a | no |

Arrangement have a copy of the feature arrangement?

Table 3 indicates the relationship between the test arrangement and the five feature arrangements.

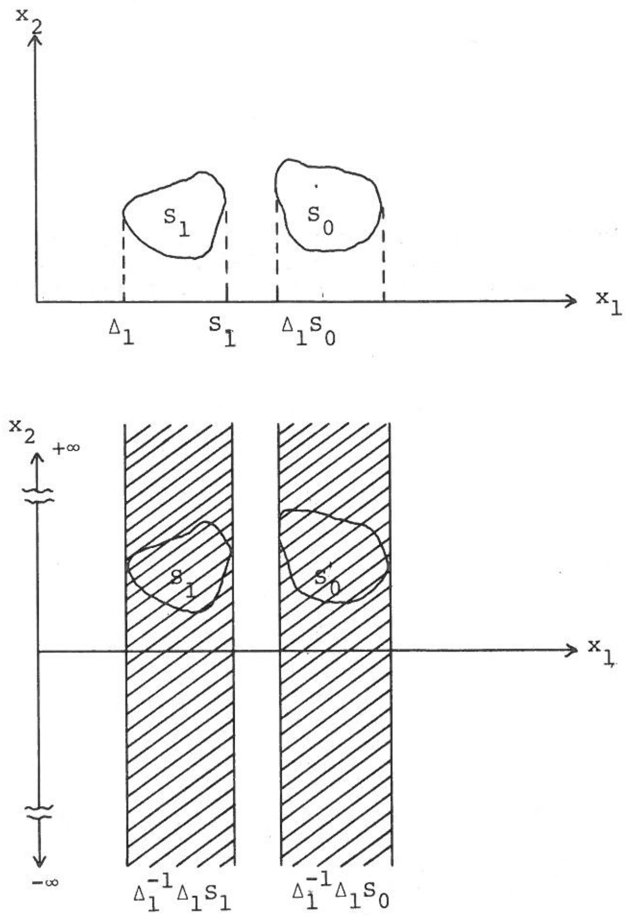


Figure 2 illustrates the projection and inverse projection operators. Assume two subsets $S_1$ and $S_0$ of E where $E = D_1 \times D_2$.

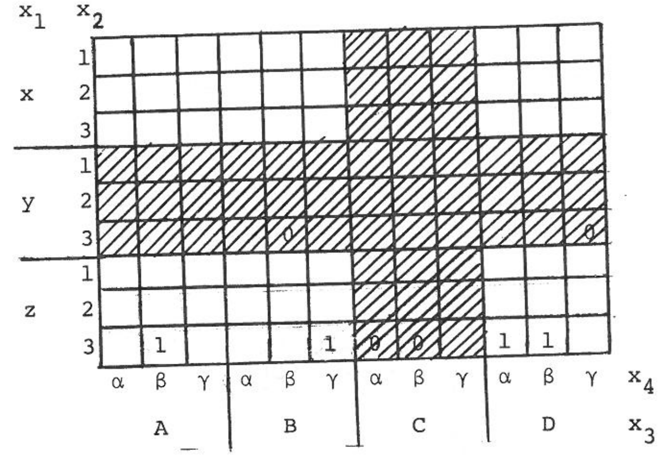


Figure 3 illustrates the sets $S_0$ and $S_1$ whose members are labeled 0 and 1, respectively and the set of 4 tuples in the order 1 cover for $S_0$: $(x_1 = y) \vee (x_3 = c)$ shown as striped.

| CYLINDER SET $x_1$ $x_2$ $x_3$ $x_4$ / MEASUREMENTS | z 3 C α | y 3 D γ | z 3 C β | y 3 B β |
|---|---|---|---|---|
| y * * * | | (X) | | X |
| * * C * | (X) | | X | |
| | ✓ | ✓ | ✓ | ✓ |

$S_0 = \{z3C\alpha, y3D\gamma, z3C\beta, y3B\beta\}$

$S_1 = \{z3D\alpha, z3D\beta, z3B\gamma, z3A\beta\}$

Figure 4 shows a table of candidate order 1 cylinder sets and events in $S_0$. The checks in the last row indicate that the two cylinder sets cover the four measurements.

| CYLINDER SET x1 | x2 | x3 | x4 | EVENTS 0010 | 0111 | 1021 | 1121 | 2011 | 2112 |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | * | * | X | | | | | |
| 2 | 0 | * | * | | | | | X | |
| 0 | * | 1 | * | X | X | | | | |
| 2 | * | 1 | * | | | | | X | X |
| 0 | * | * | 0 | X | | | | | |
| 0 | * | * | 1 | | X | | | | |
| * | 1 | 1 | * | | X | | | | X |
| * | 0 | 2 | * | | | X | | | |
| * | * | 1 | 2 | | | | | | X |
| | | | | ✓ | ✓ | ✓ | | ✓ | ✓ |

Figure 5 shows the cover table for example 2.

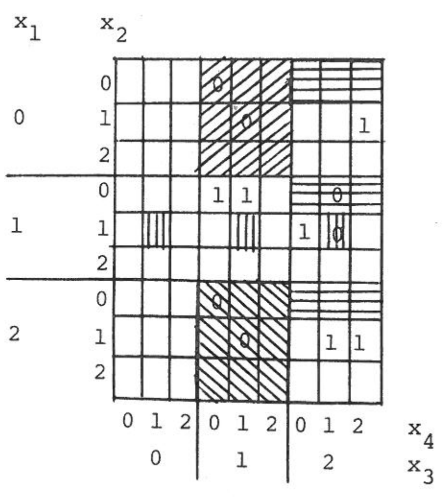


Figure 6 shows the Karnaugh map of the sets $S_0$, $S_1$ and a minimal order 3 cover for example. The cover $S_0$ is striped.