NEIGHBORHOOD GRAPHS IN CLASSIFICATION PROBLEMS FOR SYMBOLIC DATA

Manabu Ichino*

ABSTRACT

This paper presents new neighborhood graphs useful to solve feature selection problems in pattern recognition for symbolic data. In pattern recognition for symbolic data, each sample pattern is described not only by quantitative features but also by qualitative features. We introduce the Cartesian System Model (CSM) as a mathematical model to treat symbolic data. Then, we define the Generality Ordered Mutual Neighborhood Graph and the Generality Ordered Interclass Mutual Neighborhood Graph based on the CSM. These neighborhood graphs play central roles in seeing details of the interclass structures. We outline the basic idea of our classifier by using simple examples.

1. Introduction

In many actual classification problems, it is important, for the generalization of data, to manipulate symbolic data such as ordered qualitative features and interval data. In this work we define an individual (a sample, a sample pattern) based on the Cartesian System Model (CSM) (Ichino, 1994, 1998), which is a mathematical model to manipulate symbolic data.

In Section 2, we explain CSM, the basis of our method. The CSM is a triplet \((U^{(d)}, \boxplus, \boxtimes)\), where \(U^{(d)}\) is the feature space in which each sample is represented by a mixture of various kinds of features, \(\boxplus\) is the Cartesian join operator which generates a generalized description from given several descriptions in the feature space, and \(\boxtimes\) is the Cartesian meet operator which extracts a common description from several given descriptions in the feature space.

In Section 3, we confine ourselves to two-class problems, then we define the Generality Ordered Mutual Neighborhood Graph (GOMNG) and the Generality Ordered Interclass Mutual Neighborhood Graph (GOICMNG) based on the CSM. The GOMNG is a (non-oriented) weighted graph. Two samples of pattern class \(C_1\) are connected by an edge, if the Cartesian join region spanned by these samples includes only samples in the same pattern class \(C_1\). Then, the total number of samples included in the join region is attached to the edge as its weight value. We use this weight value as a measure of generality (we call it the inside generality) for the Cartesian join region to describe pattern class \(C_1\). The GOICMNG is an (oriented) weighted graph, where two samples, one is from class \(C_1\), another one the opposite class \(C_2\), are connected by an oriented edge if the Cartesian join region spanned by these samples includes only samples from the opposite pattern class \(C_2\). Then, the total number of samples from \(C_2\) included in the join region is attached to the edge as its weight value.

* Department of Information and Arts, Tokyo Denki University, Hatoyama, Saitama 350-0394, Japan

Key words: Cartesian System Model; Neighborhood graph; Pattern classifier; Symbolic data
We use this weight value as a measure of generality (we call it the outside generality) of the Cartesian join region to describe the opposite pattern class $C_2$. These graphs are useful to find an effective feature subset and to find samples that play a central role in designing a classifier.

We outline our approach to designing a pattern classifier based on the GOMNG and the GOICMNG in Section 4. Section 5 is a summary.

2. The Cartesian System Model

2.1. The feature space and events

Let $U_k$ be the domain of feature $X_k$, $k = 1, 2, \ldots , d$. Then, the feature space is defined by the product set

$$U^{(d)} = U_1 \times U_2 \times \cdots \times U_d.$$ (1)

Since we permit the simultaneous use of various feature types, we use the notation $U^{(d)}$ for the feature space in order to distinguish it from the usual $d$-dimensional Euclidean space $U^d$. Each sample pattern in $U^{(d)}$ is represented by

$$E = E_1 \times E_2 \times \cdots \times E_d \text{ or } E = (E_1, E_2, \ldots, E_d),$$ (2)

where $E_k$, $k = 1, 2, \ldots , d$, is the feature value taken by the feature $X_k$. We are able to use the following five feature types.

1) Continuous quantitative feature (height, weight, etc.)

2) Discrete quantitative feature (the number of family members, etc.)

3) Ordinal qualitative feature (academic career, etc., where there is some kind of ordered relationships between elements)

4) Nominal qualitative feature (sex, blood type, etc.)

5) Tree-structured feature (Figure 1 shows a tree structure in which each node is represented by a nominal qualitative feature)

When we use feature types 1), 2) and 3), we permit interval values of the form $E_k = [a, b]$, and in feature types 4) and 5), we permit finite sets as feature values. The Cartesian product (2) described in terms of feature types 1) - 5) is called an event. It should be noted that a sample pattern is an event.

2.2. The Cartesian join operator

The Cartesian join, $A \boxtimes B$, of a pair of events $A = (A_1, A_2, \ldots , A_d)$ and $B = (B_1, B_2, \ldots , B_d)$ in the feature space $U^{(d)}$, is defined by

$$A \boxtimes B = [A_1 \boxplus B_1] \times [A_2 \boxplus B_2] \times \cdots \times [A_d \boxplus B_d],$$ (3)

where $[A_k \boxplus B_k]$ is the Cartesian join of feature values $A_k$ and $B_k$ for feature $X_k$ and is defined as follows. When $X_k$ is a quantitative or an ordinal qualitative feature, let $A_k = [A_{kL}, A_{kU}]$ and $B_k = [B_{kL}, B_{kU}]$, then $[A_k \boxplus B_k]$ is the closed interval given by

$$[A_k \boxplus B_k] = [\min(A_{kL}, B_{kL}), \max(A_{kU}, B_{kU})].$$ (4)
When \( X_k \) is a nominal feature, \([A_k \oplus B_k]\) is the union:

\[
[A_k \oplus B_k] = A_k \cup B_k. \quad (5)
\]

When \( X_k \) is a tree structured feature, let \( N(A_k) \) be the nearest parent node, which is common to all terminal values included in \( A_k \). Then, if \( N(A_k) = N(B_k) \)

\[
[A_k \bowtie B_k] = A_k \cup B_k. \quad (6)
\]

On the other hand if \( N(A_k) \neq N(B_k) \),

\[
[A_k \oplus B_k] = \{\text{the set of all terminal values branched from the nodes } N(A_k) \text{ and } N(B_k)\}, \quad (7)
\]

where, for each feature value, we assume that

\[
[A_k \boxtimes A_k] = A_k. \quad (8)
\]

For example, in Figure 1, “Intel” is the nearest parent node for \( A_k = \{80286, 80386\} \). If \( A_k = \{80286, 80386\} \) and \( B_k = \{80386\} \), then \( N(A_k) = N(B_k) = \text{“Intel”} \). Thus \([A_k \boxtimes B_k] = A_k \cup B_k = \{80286, 80386\} \). Furthermore, if \( A_k = \{80286, 80386\} \) and \( B_k = \{68020, 68030\} \), then \( N(A_k) \neq N(B_k) \). Therefore, (7) leads to \([A_k \oplus B_k] = \{80286, 80386, 80486, 68020, 68030, 68040\} \).

2.3. The Cartesian meet operator

The Cartesian meet, \( A \boxast B \), of a pair of events \( A = (A_1, A_2, \ldots, A_d) \) and \( B = (B_1, B_2, \ldots, B_d) \) in the feature space \( U^{(d)} \), is defined by

\[
A \boxast B = [A_1 \boxast B_1] \times [A_2 \boxast B_2] \times \cdots \times [A_d \boxast B_d], \quad (9)
\]

where \([A_k \boxast B_k]\) is the Cartesian meet of feature values \( A_k \) and \( B_k \) for feature \( X_k \) defined by the intersection

\[
[A_k \boxast B_k] = A_k \cap B_k. \quad (10)
\]
When the intersection \(10\) takes the empty value \(\emptyset\), for at least one feature, the events \(A\) and \(B\) have no common part. We denote this fact by

\[
A \boxtimes B = \emptyset, \tag{11}
\]

and we say that \(A\) and \(B\) are completely distinguishable.

In Figure 2, (a) and (b) show the Cartesian join and the Cartesian meet, respectively, in the Euclidean plane.

We call the triplet \((U^{(d)}, \boxplus, \boxtimes)\) the **Cartesian System Model (CSM)** (Ichino, 1994, 1998).

3. Neighborhood graphs

3.1. The Generality Ordered Mutual Neighborhood Graph

In the following we consider only two pattern classes \(C_1\) and \(C_2\) with \(N_1\) and \(N_2\) objects respectively. For pattern class \(C_k\), \(k = 1, 2\), let

\[
S_k = \{s_{kp}, p = 1, 2, \ldots, N_k\}, k = 1, 2 \tag{12}
\]

be the given sets of sample patterns, where we assume that \(S_1\) and \(S_2\) have no common sample patterns. Let

\[
F_0 = \{1, 2, \ldots, d\} \tag{13}
\]

be the set of labels attached to the \(d\) features \(X_1, X_2, \ldots, X_d\). It will be called the feature set. A sample pattern \(s_{kp}\) is represented in the feature space \(U^{(d)}\) as follows:

\[
E_{kp} = E_{kp1} \times E_{kp2} \times \cdots \times E_{kpd}. \tag{14}
\]

**Definition 1. Join region**

For a pair of samples \(s_{kp}, s_{kq} \in S_k\), let \(R(s_{kp}, s_{kq} \mid F)\) be the Cartesian join region in the feature space spanned by a feature subset \(F \subseteq F_0\) i.e.,

\[
R(s_{kp}, s_{kq} \mid F) = \Pi_{r \in F}[E_{kpr} \boxplus E_{kqr}], \tag{15}
\]

![Fig. 2: Cartesian join and meet in the Euclidean plane](image)
where $\Pi$ is the operator for the Cartesian product and square brackets “[“ and “]” mean here that the boundary values of the Cartesian join for feature $X_r$ are included in the join region (i.e., a closed region).

**Definition 2. Mutual neighborhood**

Samples $s_{1p}, s_{1q} \in S_1$ of pattern class $C_1$ are called the *mutual neighbors* against the opposite class $C_2$ under a feature subset $F \subseteq F_0$, if there is no sample of $S_2$ within the join region $R(s_{1p}, s_{1q} \mid F)$ (see Figure 3(a)).

**Definition 3. Inside generality**

Let samples $s_{1p}$ and $s_{1q}$ be the mutual neighbors, and let the join region $R(s_{1p}, s_{1q} \mid F)$ cover $n$ samples from $S_1$. Then, we say that the join region $R(s_{1p}, s_{1q} \mid F)$ has the *inside generality* $n$ to describe class $C_1$, denoted by $gi(s_{1p}, s_{1q} \mid F)$. From the assumption for the sets of sample patterns, it follows that

$$1 \leq gi(s_{1p}, s_{1q} \mid F) \leq N_1,$$  
(16)

where the lower bound occurs when $p = q$.

**Definition 4. Generality Ordered Mutual Neighborhood Graph**

For a given selection $F$ of features, the *Generality Ordered Mutual Neighborhood Graph*, written $GOMNG(C_1 : C_2 \mid F)$, is a graph which is non-oriented and weighted, and is constructed as follows.

1) If two samples $s_{1p}$ and $s_{1q}$ in $S_1$ are mutual neighbors against $C_2$ under a feature subset $F$, $s_{1p}$ and $s_{1q}$ are connected by an edge.

2) For the edge which connects $s_{1p}$ and $s_{1q}$, we attach the inside generality value $gi(s_{1p}, s_{1q} \mid F)$ as the weight for the edge.

In Figure 3 (a), samples $s_{1p}$ and $s_{1q}$, for example, are mutual neighbors, since no samples from class $C_2$ exist in the region $R(s_{1p}, s_{1q} \mid X_1, X_2)$, while the region includes 3 samples...
from class $C_1$. Figure 3 (b) shows the resulting $GOMNG(C_1 : C_2 \mid X_1, X_2)$, where we have omitted all edges with the inside generality 1.

**Definition 5 Generality Ordered Mutual Neighborhood Matrix**

We denote the *Generality Ordered Mutual Neighborhood Matrix* by $GOMNM(C_1 : C_2 \mid F)$, where element $(p, q)$ of this matrix is the inside generality $gi(s_{1p}, s_{1q} \mid F)$ if the corresponding samples are mutual neighbors, otherwise zero.

The $GOMNM(C_1 : C_2 \mid F)$ is an $N_1 \times N_1$ symmetric matrix and is used in Section 4 to design a classifier.

**3.2. Generality Ordered Interclass Mutual Neighborhood Graph**

**Definition 6 Interclass mutual neighborhood**

Two samples $s_{1p} \in S_1$ and $s_{2q} \in S_2$ are called *interclass mutual neighbors* under a feature subset $F \subseteq F_0$, if there is no sample of $S_1$ within the join region

$$R(s_{1p} : s_{2q} \mid F) = \Pi_{r \in F}(E_{1pr} \oplus E_{2qr}),$$

where the meanings of symbols are the same in Definitions 1 and 2, and the left parenthesis "(" means that sample $s_{1p}$ is not included in the join region by the feature subset $F$. If the samples $s_{1p}$ and $s_{2q}$ become the same in a feature subset $F$, we define that these two samples are not interclass mutual neighbors under the feature subset $F$.

In Figure 4 (a), two samples $s_{1p} \in S_1$ and $s_{2q} \in S_2$ are interclass mutual neighbors, where the two dotted line segments show the fact that sample $s_{1p}$ is not included in the join region $R(s_{1p} : s_{2q} \mid X_1, X_2)$.

**Definition 7 Outside generality**

Let samples $s_{1p} \in S_1$ and $s_{2q} \in S_2$ be interclass mutual neighbors under a feature subset $F$, and let the join region $R(s_{1p} : s_{2q} \mid F)$ include $m$ samples from $S_2$. Then, we say that the join region $R(s_{1p} : s_{2q} \mid F)$ has the *outside generality* $m$ to describe class $C_2$, denoted

![Diagram](a)

Fig. 4: Generality Ordered Interclass Mutual Neighborhood Graph
Neighborhood Graphs in Classification Problems for Symbolic Data

by $g_0(s_{1p} : s_{2q} | F)$, where it follows that

$$1 \leq g_0(s_{1p} : s_{2q} | F) \leq N_2. \quad (18)$$

The lower bound in (18) occurs when samples $s_{1p}$ and $s_{2q}$ are different with respect to the selected feature subset $F$ and only sample $s_{2q} \in S_2$ is included in the join region $R(s_{1p} : s_{2q} | F)$.

**Definition 8. Generality Ordered Interclass Mutual Neighborhood Graph**

The Generality Ordered Interclass Mutual Neighborhood Graph, written $GOICMNG(C_1 : C_2 | F)$, is a graph which is oriented and weighted, and is constructed as follows.

1) If two samples $s_{1p} \in S_1$ and $s_{2q} \in S_2$ are interclass mutual neighbors under a feature subset $F$, an oriented edge goes from $s_{1p}$ to $s_{2q}$.

2) To this oriented edge from $s_{1p}$ to $s_{2q}$, we attach the outside generality value $g_0(s_{1p} : s_{2q} | F)$ as its weight.

Figure 4 (b) illustrates a $GOICMNG(C_1 : C_2 | X_1, X_2)$. In this figure, the sample pair $(s_{1p}, s_{2q})$ has the largest outside generality. Moreover, sample $s_{1p} \in S_1$ has a particular property that it is connected to all samples in $S_2$.

**Definition 9 Generality Ordered Interclass Mutual Neighborhood Matrix**

We denote the Generality Ordered Interclass Mutual Neighborhood Matrix by $GOICMNM(C_1 : C_2 | F)$, where element $(p, q)$ of this matrix is the outside generality $g_0(s_{1p} : s_{1q} | F)$ in Definition 7 if the corresponding samples are interclass mutual neighbors, otherwise zero.

The $GOICMNM(C_1 : C_2 | F)$ is an $N_1$ by $N_2$ rectangular matrix and is used also in Section 4 to design a classifier.

4. Outline of a design method for symbolic classifier

4.1. Region-based classifier

Now, let $E_{kp}(F), p = 1, 2, \ldots , N_k$, denote the $p$-th sample of class $C_k, k = 1, 2$, with respect to the feature subset $F \subseteq F_0$. Furthermore, let $R_{kp}(F_{kp}), p = 1, 2, \ldots , m_k(\leq N_k)$, be the $p$-th event to describe class $C_k$ under the feature subset $F_{kp} \subseteq F_0$ such that

$$E_{kp}(F_{kq}) \subseteq \bigcup_{q=1}^{m_k} R_{kq}(F_{kq}), p = 1, 2, \ldots , N_k, k = 1, 2 \quad (19)$$

$$E_{kp}(F_{jq}) \approx R_{jq}(F_{jq}) = \emptyset, p = 1, 2, \ldots , N_k, q = 1, 2, \ldots , m_j, j = 1, 2(k \neq j), \quad (20)$$

where $\Phi$ denotes two events $E_{kp}$ and $R_{jq}$ that are completely distinguishable with respect to feature subset $F_{jq}$. Since our training sets in (12) have no common samples, the existence of events in (19) and (20) may be clear by setting that $R_{kp}(F_0) = E_{kp}(F_0), p = 1, 2, \ldots , N_k, k = 1, 2$. Then, we can construct a region-based classifier as follows.
Region-based classifier:

For a given pattern \( E \),

1) \( E \) is decided to originate from class \( C_k \), if there exists an \( R_{kq}(F_{kq}) \) for which \( E(\mathbf{F}_{kq}) \subseteq R_{kq}(F_{kq}) \) and \( E \) is completely distinguishable from all \( R_{jq}(F_{jq}), q = 1, 2, \ldots, m_j, (j \neq k) \).

2) \( E \) is rejected as type-I reject, if it is covered by events of the two pattern classes.

3) \( E \) is rejected as type-II reject, if any event does not cover it.

Some basic statistical properties of this type of classifier are given by the author (Ichino, 1976).

In the region-based classifier, our basic problem is to find events \( R_{jp}(F_{jp}), p = 1, 2, \ldots, m_j, j = 1, 2 \), defined in (19) and (20). For each event \( R_{jp}(F_{jp}) \), it is important to select the feature subset \( F_{jp} \) as small as possible. This is because a smaller sized feature subset \( F_{jp} \) achieves a higher generality in the description of class \( C_j \). This assertion may be supported by the properties (Ichino, 1994, 2000):

\[
\text{if } F' \subseteq F \subseteq F_0, \text{ then }
\]

\[
gi(s_{1p}, s_{1q} \mid F) \leq gi(s_{1p}, s_{1q} \mid F') \quad \text{and} \quad go(s_{1p} : s_{2q} \mid F) \leq go(s_{1p} : s_{2q} \mid F')
\]

for all sample pairs \((s_{1p}, s_{1q})\) and \((s_{1p}, s_{2q})\). Therefore, the following ideas for feature selection may be useful.

1) Feature selection based on the Generality Ordered Mutual Neighbors (GOMN)

Find a minimum feature subset \( F \subseteq F_0 \) and a sample pair \((s_{1p}, s_{1q})\) assuring that the samples \( s_{1p} \) and \( s_{1q} \) are mutual neighbors and the inside generality \( gi(s_{1p}, s_{1q} \mid F) \) is maximal.

2) Feature selection based on the Generality Ordered Interclass Mutual Neighbors (GOICMN)

Find a minimum feature subset \( F \subseteq F_0 \) and a sample pair \((s_{1p}, s_{2q})\) assuring that the samples \( s_{1p} \) and \( s_{2q} \) are interclass mutual neighbors and the outside generality \( go(s_{1p} : s_{2q} \mid F) \) is maximal.

We should use 1) or 2) according to the interclass structures. Simple examples given below illustrate the use of these feature selection methods. In these methods, our remaining problem is how to search a desired feature subset and a pair of samples. The successive intersection procedure (Ichino, 1981, cited in Bow, 1992) is an approach to solve our problem. We summarize here the procedure.

Let \( s_{1p}, s_{1q} \in S_1 \) be selected samples of class \( C_1 \). For each sample \( s_{2k} \in S_2 \) of class \( C_2 \), let \( L_{2k} \) be the set of feature number \( r \) such that the join region \( R(s_{1p}, s_{1q} \mid X_r) \) excludes the sample \( s_{2k} \). Then, our successive intersection process based on the GOMN is summarized as follows.

1) Let \( E_{2k}(F), k = 1, 2, \ldots, N_2 \), denote the \( k \)-th sample \( s_{2k} \) of class \( C_2 \) with respect to a feature subset \( F \subseteq F_0 \).

2) Let \( F \) be \( F_0 \) and let \( T \) be the list of sample numbers \( \{1, 2, \ldots, N_2\} \).

3) For each \( k \), iterate the process: if \( E_{2k}(F \cap L_{2k}) \not\subseteq R(s_{1p}, s_{1q} \mid F \cap L_{2k}) = \emptyset \), then replace \( F \) by \( F \cap L_{2k} \) and delete \( k \) from \( T \), otherwise retain the sample number \( k \) in the list \( U \).
4) If the list $U$ is empty, then stop the process, otherwise transfer $U$ to the list $T$ and repeat from step 2) to obtain other feature subsets.

If the step 3) is repeated $m$ times, we have $m$ feature subsets $F_1, F_2, \ldots, F_m$. Each feature subset $F_r$ classifies a subset of $S_2$ from the join region spanned by samples $s_{1p}$ and $s_{1q}$. In other words, the join region by samples $s_{1p}$ and $s_{1q}$ is surrounded by $m$ different clusters of samples for class $C_2$. On the other hand, in the case of $m = 1$, the training set $S_2$ is recognized as a single cluster distinguished from the join region by samples $s_{1p}$ and $s_{1q}$.

Feature selection based on the $GOICMN$ may be realized by a similar algorithm to the above. We illustrate our feature selection and classification methods by using two simple artificial data sets and Quinlans’ data (Quinlan, 1986) as a case for symbolic data.

4.2. A two-dimensional problem

We use a two-dimensional two-class problem shown in Figure 5. Pattern class $C_1$ has 8 samples {1, 2, 3, 4, 5, 6, 7, 8}, while $C_2$ has 7 samples {a, b, c, d, e, f, g}. In this example, we can discriminate two pattern classes by using only feature $X_1$, although feature $X_2$ has a partital discrimination power between two sets {1, 2, 7, 8} and {a, b, c, d, e, f, g}. Table 1 is the description of $GOMNM$’s and $GOICMNM$’s, where Figure 6 specifies the concrete arrangements of these matrices.

From Table 1, we see the following facts:

1) Samples {1, 2, 3, 4, 5, 6} of class $C_1$ constitute a tight cluster against class $C_2$. Especially, the pairs (1, 6) and (2, 5) have the largest inside generality 6. Samples “7” and “8” of $C_1$ constitute another cluster.

2) Samples {2, 4, 6, 7} of class $C_1$ have nonzero outside generality for all samples in class $C_2$.

![Fig. 5: A two-class problem](image)

![Fig. 6: The arrangement of $GOMNM$’s and $COICMNM$’s in Table 2](image)
We apply the feature selection based on the GOICMNM to sample "2" of class $C_1$. The join region $2 \sqsubseteq a$, for example, requires only feature $X_1$ to assure that samples "2" and "a" become the interclass mutual neighbors. We have the same result for sample pairs $(2,b),(2,c),(2,d),(2,e),(2,f),$ and $(2,g)$. We should try several other samples, "4", "6", "7" etc., to confirm the importance of feature $X_1$. Then, we have Table 2 as the results in the reduced feature space. We see again that samples "2", "4", "6", and "7" of $C_1$ and samples "d", "e", and "g" of $C_2$ have larger outside generality values. It should be noted that the results of Table 2 figure out the interclass structure more clearly than the results in Table 1. Samples "2", "4", and "6" have the same properties for the outside generality and the inside generality. Therefore, we have several possibilities for the selection of events in (19) and (20). One possible solution to realize our region-based classifier is that $R_{11}(\{1\}) = 1 \sqsubseteq 2$ and $R_{12}(\{1\}) = 7 \sqsubseteq 8$ for class $C_1$, and $R_{21}(\{1\}) = d \sqsubseteq c$ for class $C_2$.

Table 1: $GOMNM$ and $GOICMNM$ for $X_1$ and $X_2$

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
<th>f</th>
<th>g</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>6</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>3</td>
<td>6</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>3</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>7</td>
<td>6</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>4</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>7</td>
<td>6</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>a</td>
<td>a</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>b</td>
<td>b</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td>4</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>c</td>
<td>c</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>7</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>d</td>
<td>d</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>7</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>e</td>
<td>e</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>f</td>
<td>f</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>g</td>
<td>g</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>6</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>3</td>
</tr>
</tbody>
</table>

Table 2: $GOMNM$ and $GOICMNM$ for $X_1$

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
<th>f</th>
<th>g</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>6</td>
<td>3</td>
<td>6</td>
<td>3</td>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>6</td>
<td>3</td>
<td>6</td>
<td>3</td>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>6</td>
<td>3</td>
<td>6</td>
<td>3</td>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>6</td>
<td>3</td>
<td>6</td>
<td>3</td>
<td>6</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>5</td>
<td>7</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>6</td>
<td>3</td>
<td>6</td>
<td>3</td>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>3</td>
<td>6</td>
<td>3</td>
<td>6</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>5</td>
<td>7</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>6</td>
<td>4</td>
<td>2</td>
<td>7</td>
<td>6</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>a</td>
<td>a</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>b</td>
<td>b</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>5</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>c</td>
<td>c</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>6</td>
<td>4</td>
<td>2</td>
<td>7</td>
<td>6</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>d</td>
<td>d</td>
<td>6</td>
<td>3</td>
<td>6</td>
<td>3</td>
<td>6</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>5</td>
<td>7</td>
<td>1</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>e</td>
<td>e</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>f</td>
<td>f</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>5</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>g</td>
<td>g</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>6</td>
<td>4</td>
<td>2</td>
<td>7</td>
<td>6</td>
<td>4</td>
</tr>
</tbody>
</table>
4.3. An XOR problem

Our next example is an XOR problem given in Figure 7. In this figure, \{1, 2, 3, 4, 5, 6, 7, 8\} is the training set for pattern class C1, and \{a, b, c, d, e, f, g, h\} is for pattern class C2.

Table 1 is the simultaneous description of GOMNM (C1 : C2 \mid X_1, X_2) and GOICMNM (C1 : C2 \mid X_1, X_2) for the data set in Figure 7. It should be noted that samples "4" and "5" in class C1 have the following particular properties.

1) Samples "4" and "5" have three other mutual neighbor samples in C1, respectively, against class C2. These samples have the maximum inside generality 4 with samples "1" and "8", respectively.

2) Samples "4" and "5" are, respectively, interclass mutual neighbors with all samples "a" to "h" in class C2.

Table 3 tells us the fact that samples "1" and "5" in Figure 7 are important to understand the interclass structure between C1 and C2. Sample pairs (4, 1) and (5, 8) are mutual neighbors with the largest inside generality four. The region 4 \oplus 1 excludes samples \{a, b, c, d\} by feature X2, samples \{e, f, g, h\} by feature X1. On the other hand 5 \oplus 8 excludes samples \{a, b, c, d\} by feature X1, samples \{e, f, g, h\} by feature X2. These facts are easily obtained by the feature selection based on the GOMN.

Since our XOR problem is symmetric for two pattern classes, the events given in (19) and (20) should be selected as: R_{11}(\{1, 2\}) = 4 \oplus 1 and R_{12}(\{1, 2\}) = 5 \oplus 8 for class C1, and R_{21}(\{1, 2\}) = b \oplus c and R_{22}(\{1, 2\}) = g \oplus h for class C2. Then, we can use our region based classifier.

4.4. Quinlan’s data

Table 4 is the data by Quinlan (1986) used for the well-known ID3. In this data, for example, samples "1" and "2" of class C1 are mutual neighbors in the four-dimensional feature space, and the inside generality of the join region R(1, 2 \mid F0) is gi(1, 2 \mid F0) = 2. If we apply the feature selection based on the GOMN for samples "1" and "2" of class C1, only the feature “Outlook” is obtained. Then, the join region R(1, 2 \mid \{Outlook\}) includes four samples \{1, 2, 4, 6\}, i.e., gi(1, 2 \mid \{Outlook\}) = 4. The same properties are valid for all different pairs of samples in \{1, 2, 4, 6\}. These results tell us that our first event to describe class C1 against class C2 should be selected as R_{11}(\{Outlook\}) = (Outlook = “overcast”), since this event has the largest inside generality and is able to discriminate all samples of class C2. Table 5 summarizes GOMNM(C1 : C2 \mid Outlook).

\[ X_2 \]

\[
\begin{array}{cccc}
1 & 2 & e & f \\
3 & 4 & g & h \\
a & b & 5 & 6 \\
c & d & 7 & 8 \\
\end{array}
\]

\[ X_1 \]

Fig. 7: An XOR problem

--- 213 ---
Table 3: \textit{GOMNM and GOICMNM}

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
<th>f</th>
<th>g</th>
<th>h</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 4: Quinlan's data

<table>
<thead>
<tr>
<th>Class</th>
<th>Outlook</th>
<th>Temp(°F)</th>
<th>Humidity(%)</th>
<th>Windy?</th>
</tr>
</thead>
<tbody>
<tr>
<td>(C_1): 1 (Play)</td>
<td>overcast</td>
<td>72</td>
<td>90</td>
<td>true</td>
</tr>
<tr>
<td>2</td>
<td>overcast</td>
<td>83</td>
<td>78</td>
<td>false</td>
</tr>
<tr>
<td>3</td>
<td>rain</td>
<td>75</td>
<td>80</td>
<td>false</td>
</tr>
<tr>
<td>4</td>
<td>overcast</td>
<td>64</td>
<td>65</td>
<td>true</td>
</tr>
<tr>
<td>5</td>
<td>sunny</td>
<td>75</td>
<td>70</td>
<td>true</td>
</tr>
<tr>
<td>6</td>
<td>overcast</td>
<td>81</td>
<td>75</td>
<td>false</td>
</tr>
<tr>
<td>7</td>
<td>rain</td>
<td>68</td>
<td>80</td>
<td>false</td>
</tr>
<tr>
<td>8</td>
<td>rain</td>
<td>70</td>
<td>96</td>
<td>false</td>
</tr>
<tr>
<td>9</td>
<td>sunny</td>
<td>69</td>
<td>70</td>
<td>false</td>
</tr>
<tr>
<td>(C_2): a (Don'tPlay)</td>
<td>rain</td>
<td>71</td>
<td>96</td>
<td>true</td>
</tr>
<tr>
<td>b</td>
<td>rain</td>
<td>65</td>
<td>70</td>
<td>true</td>
</tr>
<tr>
<td>c</td>
<td>sunny</td>
<td>80</td>
<td>90</td>
<td>true</td>
</tr>
<tr>
<td>d</td>
<td>sunny</td>
<td>85</td>
<td>85</td>
<td>false</td>
</tr>
<tr>
<td>e</td>
<td>sunny</td>
<td>72</td>
<td>95</td>
<td>false</td>
</tr>
</tbody>
</table>

Table 5: \textit{GOMNM}(C_1 : C_2 \mid \text{Outlook})

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>4</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>4</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>4</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>4</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
For remained samples \(\{3, 5, 7, 8, 9\}\) of \(C_1\), we try again the feature selection based on the GOMN. For example, we find two features \(\{\text{Outlook}, \text{Windy}\?\}\) for samples “3” and “7” and \(R(3, 7 | \{\text{Outlook}, \text{Windy}\?\})\) includes three samples \(\{3, 5, 7\}\), i.e., \(g_i(3, 7 | \{\text{Outlook}, \text{Windy}\?\}) = 3\). Therefore, we have the second event \(R_{12}(\{\text{Outlook}, \text{Windy}\?\}) = (\text{Outlook} = \text{"rain"}) \times (\text{Windy}\? = \text{false}).\) Similarly, samples “5” and “9” lead the third event \(R_{12} (\{\text{Outlook}, \text{Humidity}\}) = (\text{Outlook} = \text{"sunny"}) \times (\text{Humidity} = 70).\)

We repeat the same event generation process for class \(C_2\) (Don’t play), and obtain the following results.

\[
\begin{align*}
R_{21} &= (\text{Outlook} : \text{"sunny"}) \times (\text{Humidity} : [80, 95]); \text{ and} \\
R_{22} &= (\text{Outlook} : \text{"rain"}) \times (\text{Windy}\? : \text{"true"}).
\end{align*}
\]

These results may be summarized as the decision rules in Figure 8. By adding a slight modification to these rules, we have very close results to ID3.

\[
\begin{align*}
\text{Outlook} &= \text{sunny:} \\
\text{Humidity} = 70: \text{play} \\
\text{Humidity} = [80, 95]: \text{don’t play} \\
\text{Outlook} &= \text{overcast: play} \\
\text{Outlook} &= \text{rain:} \\
\text{Windy}\? = \text{true}: \text{Don’t play} \\
\text{Windy}\? = \text{false}: \text{Play}
\end{align*}
\]

\[
\begin{align*}
\text{Humidity} \leq 70: \text{play} \\
\text{Humidity} > 70: \text{don’t play}
\end{align*}
\]

(Modification)

(In ID3, Humidity 75 was selected instead of 70)

Fig. 8: Obtained decision tree and its modification

5. Concluding remarks

We presented two new neighborhood graphs, the Generality Ordered Mutual Neighborhood Graph (GOMNG) and the Generality Ordered Interclass Mutual Neighborhood Graph (GOIMNG) based on the Cartesian system model (CSM) which is our mathematical model to treat symbolic data. Then, we described briefly our design method for a symbolic classifier based on three simple examples.

Acknowledgements

The author gratefully acknowledges Prof. Hans-Hermann Bock and Prof. Noboru Ohsumi for their interest in the work. The author also wishes to thank Prof. Yutaka Tanaka and a referee for many valuable comments and corrections. This work was supported in part by the Grant-in-Aid of the Ministry of Education, Science, Sports and Culture of Japan.

— 215 —
REFERENCES