韓國 컴퓨터情報學會 論文誌
第11卷 第1號, 2006. 3.

2006-11-1-1-2

최적화에 기반을 둔 LAD의 패턴 생성 기법

장 인 용*, 류 홍 서**

Optimization-Based Pattern Generation for LAD

In-Yong Jang*, Hong-Seo Ryoo**

요 약

LAD(Logical Analysis of Data)는 Boolean-logic에 기반을 둔 데이터 마이닝 방법론이다. LAD에 의한 데이터 분석 시 중요한 과정은 데이터 집합에 숨겨진 구조적 정보를 패턴의 형식으로 발견해내는 패턴 생성 단계이다. 기존의 패턴 생성 방법은 열거법에 기반을 두고 있어 높은 차수의 패턴을 생성하는 것은 실질적으로 불가능하였다. 본 논문에서는 최적화에 기반을 둔 패턴 생성 방법론을 제안하고 혼합 정수 선형 모형과 SCP(Set Covering Problem)의 두 가지 모형을 제안한다. 기계학습 분야에서 널리 쓰이는 데이터 집합에 대해 제안된 패턴 생성 방법을 이용한 분석 실험을 통하여 기존의 패턴 생성 방법으로는 생성될 수 없는 패턴을 쉽게 생성하는 효율성을 입증하였다.

Abstract

The logical analysis of data(LAD) is a Boolean-logic based data mining tool. A critical step in analyzing data by LAD is the pattern generation stage where useful knowledge and hidden structural information in data is discovered in the form of patterns. A conventional method for pattern generation in LAD is based on term enumeration that renders the generation of higher degree patterns practically impossible. In this paper, we present a novel optimization-based pattern generation methodology and propose two mathematical programming models, a mixed 0-1 integer and linear programming (MILP) formulation and a well-studied set covering problem (SCP) formulation for the generation of optimal and heuristic patterns, respectively. With benchmark datasets, we demonstrate the effectiveness of our models by automatically generating with ease patterns of high complexity that cannot be generated with the conventional approach.

▸ Keyword : 기계학습(Machine Learning), 데이터 마이닝(Data Mining), 분류(Classification), 최적화(Optimization)

• 제1저자 : 장인용
• 접수일 : 2005.11.18, 심사완료일 : 2005.12.29
* 고려대학교 산업시스템정보공학과 석사, ** 고려대학교 산업시스템정보공학과 교수

Ⅰ. 서 론

LAD(Logical Analysis of Data)는 Boolean-logic에 기반을 둔 데이터 분류 방법론으로 두 가지 종류의 데이터들을 가지는 이진 분류 문제(binary classification problem)에 대해서 데이터의 이진화(binarization), 지지 집합(support set)의 선택, 패턴 생성(pattern generation), 분류기(classifier) 생성의 4단계의 절차를 거쳐 데이터를 분석하게 된다[1, 2, 3]. 여기에서 지지 집합은 두 종류의 데이터를 구별할 수 있는 전체 속성 집합의 부분 집합을 나타내며 패턴은 데이터 집합의 숨겨진 논리적 정보의 기본 단위를 나타낸다. LAD 분류기는 패턴의 논리합(disjunction)으로 이루어진다. 따라서 LAD 분류기의 분류 능력은 어떤 패턴이 발견되는가에 크게 영향을 받는다. [2]에서 제시된 기존의 패턴 생성 방법은 열거(enumeration)에 기반을 두고 있어 고려해야 될 패턴의 개수는 패턴 차수(degree)의 지수 함수 형태로 급격하게 증가하게 된다. 따라서 분석 대상인 데이터 집합의 속성의 개수가 많을 경우 가능한 패턴들을 모두 고려하여 그 중 판별력이 우수한 패턴을 선별하기에 비효율적인 방법이 된다. 본 논문에서는 최적화 기법을 이용하여 열거법에 의존하지 않은 효율적인 패턴 생성 방법을 새롭게 제안한다.

본 논문은 다음과 같이 구성되어 있다. 두 번째 절에서는 [2]에서 제시된 패턴 생성 방법이 설명될 것이며 이어서 세 번째 절에서는 최적화에 근거한 패턴 생성 방법이 개발될 것이다. 네 번째 절에서는 기계 학습에 통용되는 여러 데이터 집합들에 대하여 본 논문에서 제안된 패턴 생성 방법들을 이용한 분석과 실험 결과들을 제시할 것이며 제안된 방법이 효율적이라는 것을 입증할 것이다.

Ⅱ 기존 LAD의 패턴 생성 방법

패턴 생성법의 설명에 필요한 몇 가지 용어들의 정의는 다음과 같다. 하나의 이진 변수를 리터럴(literal)이라 부르며, 하나 이상의 리터럴들의 논리곱(conjunction)으로 이루어진 것을 텀(term)이라 한다. 또한 텀에 포함된 리터럴의 수를 그 텀의 차수(degree)라 한다. 임의의 텀 t와 이진 데이터 p에 대해서 t와 동일한 형태의 리터럴의 논리곱이 p에 포함이 되어 있을 때 $t(p)=1$로 나타내며 텀 t가 데이터 p를 커버(cover)한다고 명명한다. 양성(positive, +)과 음성(negative, −)의 두 가지 종류의 데이터를 갖는 데이터 집합에 대하여 임의의 텀이 양성 데이터를 적어도 하나 커버하며 동시에 음성 데이터를 하나도 커버하지 않을 때 그 텀을 양성 패턴이라 한다. 반대로 음성 데이터를 적어도 하나 커버하며 양성 데이터를 하나도 커버하지 않는 텀을 음성 패턴이라 한다.

[2]에서 제시된 패턴 생성 방법은 크게 상향식(bottom-up)·하향식(top-down)·혼합(hybrid) 접근법의 세 부류로 나눌 수 있다. 상향식 접근법은 가능한 모든 차수 1의 텀으로부터 그 텀이 패턴이 될 때까지 리터럴을 하나씩 더하는 과정을 반복하여 패턴을 생성한다. 하향식 접근법은 상향식 접근법과는 반대로 이진화된 데이터 자체를 하나의 패턴으로 간주하고 그것이 최소 차수의 패턴이 될 때까지 리터럴을 제거하는 과정을 반복하여 패턴을 생성한다. 혼합 접근법은 상향식 접근법과 하향식 접근법을 혼합한 방법으로 특정 차수까지(보통 4~5) 상향식 접근법을 이용하여 패턴을 생성하고 생성된 패턴에 의해서 커버가 되지 않는 데이터들에 대해서만 하향식 접근법을 적용하여 패턴을 생성하게 된다.

이와 같이 [2]에서 제시된 패턴 생성 방법들은 모두 열거 기법에 의존을 하고 있다. 그렇기 때문에 전체 이진 속성의 개수를 n, 열거할 텀의 차수를 d라고 할 때 $2^d\binom{n}{d}$개의 텀을 고려해야 하며 이는 텀의 차수에 대해 지수 함수 형태로 급격하게 증가한다. 따라서 분석할 데이터의 이진 속성이 많을 경우에는 기존의 패턴 생성 방법으로는 분석이 불가능할 수도 있다.

III 최적화에 기반을 둔 패턴 생성 방법

수리적 패턴 생성 모형을 소개하기에 앞서 이진화된 데이터 집합을 $B^\circ(\circ \in \{+,-\})$, B°에 포함된 이진 속성의 개수를 n, i번째 데이터의 j번째 리터럴을 B_{ij}°, 데이터의 인덱스(index) 집합을 J°라 하자. 또한 이진 집합에서 $\circ$가 아닌 다른 원소를 $\overline{\circ}$라 하고 $a_{ijk}^\circ, b_{ijk}^\circ$를 다음과 같이 정의하자.

$$a_{ijk}^\circ = \begin{cases} 1 & \text{if } B_{ik}^\circ = B_{jk}^\circ \\ 0 & \text{otherwise} \end{cases}$$

$$b_{ijk}^\circ = \begin{cases} 1 & \text{if } B_{ik}^\circ = B_{jk}^{\overline{\circ}} \\ 0 & \text{otherwise} \end{cases}$$

3.1 혼합 정수 선형 패턴 생성 모형

위의 정의에 의하여 B°에 속하는 i번째 데이터(B_i°)를 커버하는 $\circ$ 패턴을 생성하는 문제는 다음과 같은 혼합 정수 선형 계획법(Mixed Integer and Linear Programming)으로 모형화가 될 수 있다.

$$\min \quad \sum_{j \in J^\circ \setminus i} y_j \quad \cdots (1)$$

$$\text{s.t.} \quad \sum_{k=1}^{n} a_{ijk}^\circ x_k + y_j \geq d,\ j \in J^\circ \setminus i \quad \cdots (2)$$

$$\sum_{k=1}^{n} b_{ijk}^\circ x_k \leq d-1,\ j \in J^{\overline{\circ}} \quad \cdots (3)$$

$$\sum_{k=1}^{n} x_k = d \quad \cdots (4)$$

$$x_k \in \{0,1\},\ k=1,...,n \quad \cdots (5)$$

$$0 \leq y_j \leq n,\ j \in J^\circ \setminus i \quad \cdots (6)$$

$$1 \leq d \leq n \quad \cdots (7)$$

위 모형에서 결정 변수 x_k는 이진 변수로 도출될 패턴에서의 B_{ik}°의 포함 여부를 나타내며 이 정의에 의하여 그 패턴은 B_i°를 자동적으로 커버하게 된다. 결정 변수 y_j는 0에서 n까지 값을 갖는 실변수로 패턴에 의한 B_j°의 커버 여부를 나타내며 d는 패턴의 차수를 나타내는 결정 변수이다. 앞에서 언급된 패턴의 정의는 (2)와 (3)으로 표현될 수 있다. (2)에서 패턴이 B_j°를 커버할 경우 그 패턴과 일치하는 B_j°의 리터럴의 수는 패턴의 차수 d와 동일하게 되고 이때 (1)에 의하여 B_j°에 해당하는 y_j는 0의 값을 가지게 된다. 패턴의 차수가 d라는 조건은 (4)로 표현된다. 따라서 (1)의 목적 함수를 가진 위의 데이터 분류 모형은 일반적으로 차수가 높은 적은 개수의 패턴을 찾는 모형이 된다.

3.2 SCP 패턴 생성 모형

O_{ik}°를 $\{j \in J^\circ | a_{ijk}^\circ = 1\}$로 c_k를 $|J^\circ|/|O_{ik}^\circ|$로 정의하자. 데이터 B_i°를 커버하는 $\circ$ 패턴을 생성하는 문제는 다음과 같은 SCP(Set Covering Problem)으로 모형화될 수 있다.

SCP_i° :

$$\min \quad \sum_{k=1}^{n} c_k x_k \quad \cdots (8)$$

$$\text{s.t.} \quad \sum_{k=1}^{n} \overline{b}_{ijk}^\circ x_k \geq 1,\ j \in J^{\overline{\circ}} \quad \cdots (9)$$

$$x_k \in \{0,1\},\ k=1,...,n \quad \cdots (10)$$

결정 변수 x_k는 앞에서 소개된 혼합 정수 선형 모형에서와 같은 의미를 가지며 역시 정의상 도출될 패턴에 의하여 B_i°는 자동적으로 커버가 된다. 그 패턴이 $\overline{\circ}$ 데이터들을 커버하지 말아야 한다는 조건은 (9)으로 주어지며, 즉 패턴과 $\overline{\circ}$ 데이터에는 서로 다른 리터럴이 적어도 하나 이상 존재해야 한다는 것을 의미한다. 혼합 정수 선형 모형에서 y_j를 통해 고려되었던 패턴이 커버하는 데이터의 수를 위 모형에서는 목적함수의 계수 c_k를 통하여 고려하고 있다.

집합 $O_{ik}^{\diamond}$는 리터럴 $B_{ik}^{\diamond}$가 커버하는 $\diamond$ 데이터의 인덱스 집합을 의미하며 커버하는 데이터의 수가 많을수록 c_k는 작은 값을 가지게 된다. 따라서 위 모형은 커버하는 데이터 수가 많은 리터럴로 이루어진 최소 차수의 패턴을 찾는 모형이 된다.

3.3 잡음을 고려한 패턴 생성 모형

일반적으로 실제의 데이터 집합에는 분류 오류(classification error), 측정 오류(measurement error), 속성 값의 분실(missing attribute values) 등에 의한 잡음(noise)들이 섞여있다. 이런 잡음을 포함한 데이터 집합에 대한 과대적합(overfitting)을 방지하기 위하여 앞에서 제시되었던 혼합 정수 선형 모형은 다음과 같이 변형될 수 있다.

$MILP_i^{\diamond}$:

$$\min \quad \sum_{j \in J^{\diamond} \setminus i} y_j \quad \cdots\cdots (11)$$

$$\text{s.t.} \quad \sum_{k=1}^{n} a_{ijk}^{\diamond} x_k + y_j \geq d,\ j \in J^{\diamond} \setminus i \quad \cdots\cdots (12)$$

$$\sum_{k=1}^{n} b_{ijk}^{\diamond} x_k - z_j \leq d-1,\ j \in \overline{J^{\diamond}} \quad \cdots\cdots (13)$$

$$\sum_{j \in \overline{J^{\diamond}}} z_j \leq \alpha \bullet |\overline{J^{\diamond}}| \quad \cdots\cdots (14)$$

$$\sum_{k=1}^{n} x_k = d \quad \cdots\cdots (15)$$

$$x_k \in \{0,1\},\ k=1,...,n \quad \cdots\cdots (16)$$

$$0 \leq y_j \leq n,\ j \in J^{\diamond} \setminus i \quad \cdots\cdots (17)$$

$$0 \leq z_j \leq 1,\ j \in \overline{J^{\diamond}} \quad \cdots\cdots (18)$$

$$1 \leq d \leq n \quad \cdots\cdots (19)$$

위 모형은 α를 이용하여 $\diamond$ 패턴의 정의를 (3)에서 (13)과 (14)으로 완화시킨다. 이때 α는 $[0,1)$ 사이의 실수로 사용자가 제공하는 파라미터이다.

$SCP_i^{\diamond}$ 모형은 해의 도출에 사용되는 그리디(greedy) 절차를 변형함으로써 잡음에 의한 과대적합을 방지할 수 있다. $[0,1)$ 사이의 파라미터인 α를 이용하여 현재의 해가 $\alpha \bullet |\overline{J^{\diamond}}|$개 이상의 제약식을 만족시킬 경우에는 절차를 종료하도록 변형을 하여 $\overline{\diamond}$ 데이터를 어느 정도 커버하는 패턴들도 고려될 수 있도록 할 수 있다.

3.4 패턴 생성 절차

앞에서 소개된 패턴 생성 모형의 해를 $x_k(k=1,...,n)$, k번째 리터럴을 위한 문자 리터럴을 a_k, 그리고 논리곱 연산자를 $\wedge$로 정의하자. 주어진 해에 상응하는 패턴 p는 다음과 같이 정의된다.

$$p = \bigwedge_{\substack{x_k=1, B_{ik}^{\diamond}=1 \\ k=1,\ldots,n}} a_k \bigwedge_{\substack{x_k=1, B_{ik}^{\diamond}=0 \\ k=1,\ldots,n}} \overline{a_k} \quad \cdots\cdots (20)$$

앞에서 소개된 두 가지 모형을 이용하여 모든 $\diamond$ 데이터를 커버할 수 있는 $\diamond$ 패턴들은 다음의 OPG 절차를 통하여 생성될 수 있다. OPG 절차에서는 발견된 패턴으로 생성되는 분류기의 분류 능력을 높이기 위하여 모든 $\diamond$ 데이터를 r번 커버할 때까지 패턴을 생성하였다. 여기에서 r은 1 이상의 정수로 사용자가 결정하는 파라미터이다. 이때 한번 이상 커버가 된 데이터에서 같은 패턴이 생성되는 것을 방지하기 위하여 현재까지 생성된 패턴들 중 대상이 되는 데이터를 커버하는 패턴들에 포함된 리터럴들에 대해서는 형성될 모형의 목적함수의 계수를 임의의 매우 큰 수로 변경하였다.

Ⅳ 실험 결과

4.1 실험 방법 및 데이터 집합

본 논문에서 개발된 패턴 생성 방법의 성능을 확인하기 위하여 두 가지의 실험을 실행하였다. 첫 번째 실험에서는 [2]에서 사용된 데이터 집합들에 대하여 분류 정확도를 측정하였다. 두 번째 실험에서는 지지 집합의 선택 없이 패턴 생성을 실시하여 기존의 패턴 생성 방법으로는 분석이 불가능한 상황에서의 분류 정확도와 패턴 생성에 걸린 시간을 측정하였다. 실험에 사용된 데이터 집합들은 모두 [4]에서 획득되었으며 분실된 속성 값을 가지는 모든 데이터들은 실험에서 제외되었다. 실험에 사용된 데이터 집합에 대한 정보는 <표 1>에 정리되어 있다. 생성된 LAD 분류기의 분류 정확도는 holdout 방법을 사용하여 측정하였다. 첫 번째 실험에서는 무작위로 선택된 학습 데이터의 비율을 50%와

80%로 두 번을 실험 하였고 두 번째 실험에서는 50%로 실험을 하였다. 정확도와 시간은 30번의 실행의 평균값을 구하여 추정하였다.

최적화에 기반을 둔 ◇ 패턴 생성 절차 (Optimization-Based Pattern Generation: OPG)

단계 1: 초기화

단계 1.1: $P^{\diamond} = \varnothing, C = J^{\diamond}, q_i = 0, \forall i \in J^{\diamond}$ 로 둔다.

단계 2: C의 임의의 원소 i에 대하여

단계 2.1: $P^{\diamond}$에 포함된 패턴 중 $B_i^{\diamond}$를 커버하는 모든 패턴에 포함된 리터럴들에 대하여 형성될 최적화 문제의 목적함수 계수를 임의의 매우 큰 수로 변경한다.

단계 2.2: $MILP_i^{\diamond}(SCP_i^{\diamond})$를 형성하고 푼다.

단계 2.3: 도출된 해로부터 (20)에 따라서 패턴 p를 생성하고 $P^{\diamond} \leftarrow P^{\diamond} \cup \{p\}$로 한다.

단계 2.4: $p(B_j^{\diamond}) = 1, j \in C$를 만족하는 모든 j에 대하여 $q_j \leftarrow q_j + 1$로 한다.

단계 2.5: $q_j = r, j \in C$를 만족하는 모든 j에 대하여 $C \leftarrow C \setminus \{j\}$로 한다.

단계 2.6: $C = \varnothing$ 이면 종료하고 그렇지 않으며 단계 2로 간다.

표 1 데이터 집합 정보
Table 1. Information on Datasets Studied

Dataset	Number of				
	classes	attributes			points
		Total	Categorical	Numerical	
Wisconsin breast cancer(WBC)	2	9	·	9	683
Cleveland heart disease(CHD)	2	13	6	7	297
Pima Indian diabetes(PID)	2	8	·	8	768
credit card scoring(CCS)	2	15	9	6	653
Boston housing(BH)	2	13	1	12	506
congressional voting(CV)	2	16	16	·	435

표 2 분류 정확도 비교
Table 2. Accuracy Comparison between Pattern Generation Methods

Dataset	Training rate(%)	MILP		SCP		기존방법	
		AVG	STD	AVG	STD	AVG	STD
WBC	50	96.1	0.9	96.6	0.9	96.9	0.9
	80	97.1	1.1	97.2	1.4	97.2	1.3
CHD	50	80.4	3.2	82.1	2.8	82.3	1.7
	80	81.1	4.5	84.0	4.5	83.8	5.2
PID	50	74.4	1.8	74.8	1.6	71.9	1.9
	80	75.4	3.1	75.8	2.9	72.3	2.4
CCS	50	86.5	1.4	86.3	1.6	85.4	1.2
	80	86.5	3.1	86.3	2.7	85.5	2.6
BH	50	84.5	2.3	84.6	2.1	84.0	1.6
	80	84.5	3.9	85.1	3.2	85.2	3.0
CV	50	95.3	1.1	95.4	1.1	96.2	1.1
	80	96.1	1.9	95.7	2.4	96.6	1.8

4.2 LAD의 구현 방법

앞에서 제시된 패턴 생성 방법이 포함된 LAD는 인텔 포트란을 이용하여 구현하였으며 실험은 리눅스 PC(Intel 2.66 GHz CPU, 512 RAM)에서 이루어졌다. SCP_i°의 경우 간단한 그리디 절차를 구현하여 해를 구했으며 사용된 규칙은 다음과 같다.

$$j^* \leftarrow \text{argmax}\{j \in J | |I_j \cap M_u|\}$$

위에서 I_j는 열 j에 의해 커버되는 행의 인덱스 집합, J는 열의 인덱스 집합, 그리고 M_u는 현재의 해로 커버가 되지 않는 행의 인덱스 집합으로 정의된다. $MILP_i^\circ$의 해는 CPLEX 9.0을 이용하여 구하였다.

4.3 수치 결과

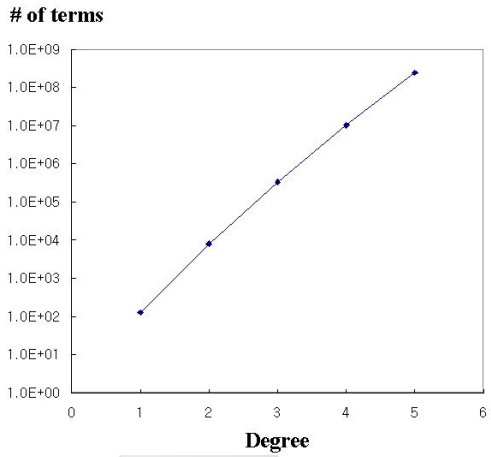


그림 1. 차수에 따른 열거 가능한 텀의 개수
Figure 1. Number of Enumerative Terms according to Degree

표 3. 이진 속성과 텀의 개수
Table 3. Number of Binary Attributes and Terms

Dataset	지지 집합 선택		
	후	전	
	이진 속성의 개수		열거 가능한 차수 5텀의 개수
WBC	14	64	2.4×10^8
CHD	14	219	1.2×10^{11}
PID	37	561	8.3×10^{12}
CCS	23	503	3.8×10^{13}
BH	30	682	1.4×10^{13}
CV	15	48	5.4×10^7

표 4. 최적화에 기반한 방법의 성능
Table 4. Performance of Optimization-Based Pattern Generation

Dataset	SCP				MILP			
	정확도		경과시간		정확도		경과시간	
	AVG	STD	AVG	STD	AVG	STD	AVG	STD
WBC	96.65	0.75	0.02	0.01	96.89	0.62	1.04	0.80
CHD	82.73	2.10	0.01	0.01	81.01	2.05	2.98	0.63
PID	74.86	2.01	0.37	0.03	74.38	1.35	263.85	59.65
CCS	86.00	1.36	0.16	0.05	85.11	1.66	67.31	21.31
BH	85.14	2.13	0.27	0.03	84.45	1.90	16.57	4.15
CV	96.41	1.13	0.01	0.01	95.58	1.04	1.07	0.74

〈표 2〉에서는 본 논문에서 개발된 패턴 생성 방법과 기존의 방법으로 만들어진 분류기의 분류 정확도를 비교하고 있다. 분류 정확도에 있어서는 CHD 데이터 집합에서 $MILP_i^\circ$에 의해 생성된 패턴으로 만들어진 분류기가 약간 낮게 나왔고 PID 데이터 집합의 경우에는 기존 방법이 본 논문에서 제안된 방법보다 낮게 나온 것을 제외하고는 통계적으로 유의한 차이를 찾기 힘들었다. 그러나 분류 오류, 측정 오류, 속성 값의 분실을 모두 고려하며 5가지 이상의 파라미터를 요구하는 [2]의 실험과 달리 본 실험에서는 분류 오류에 대한 과대 적합을 α와 r의 두 가지 파라미터만을 사용하여 방지하였음을 알린다.

(그림 1)은 지지 집합의 선택 없이 모든 이진 속성을 패턴 생성에 이용할 경우 WBC 데이터 집합에서의 차수에 따른 열거 가능한 텀의 수를 보여준다. 텀의 차수에 따라 텀의 개수의 차이가 크기 때문에 텀의 개수를 나타내는 축은 로그 스케일로 표시되었다. 열거법에 기반을 둔 패턴 생성 방법을 사용할 경우 고려해야할 차수 4의 텀의 개수는 10^7 이상임이 (그림 1)에 나타나있다. 〈표 3〉은 실험에 사용된 각 데이터 집합에 대하여 지지 집합의 선택 전후에서의 이진 속성의 개수와 선택 전의 열거 가능한 차수 5 텀의 개수를 보여준다. 열거 가능한 차수 5 텀의 개수가 CV집합에서는 10^7이상으로 가장 적은 것으로 나타났으며 CCS집합과 BH집합에서는 10^{13} 이상으로 가장 많은 것으로 나타났다. (그림 1)과 〈표 3〉은 모두 지지 집합의 선택이 없을 경우 실험에 사용된 모든 데이터 집합에 대해서 열거법을 이용한 패턴 생성 방법은 적용이 불가능하다는 점을 잘 보여주고 있다.

〈표 4〉는 모든 이진 속성을 이용하였을 때 최적화에 기반을 둔 방법으로 생성된 패턴으로 만들어진 LAD 분류기의 분류 정확도와 패턴 생성에 걸린 시간을 보여준다. 분류 정확도에 있어서는 지지 집합을 선택한 〈표 2〉의 결과와 큰 차이를 찾기 힘들었다. 경과 시간에 있어서는 $MILP_i^\circ$ 접근법이 SCP_i° 접근법보다 분석에 더 많은 시간이 걸렸다. 이는 $MILP_i^\circ$ 접근법의 경우 CPLEX를 사용하여 최적해 또는 그에 근접한 해를 구하였지만 SCP_i° 접근법의 경우에는 간단한 그리디 절차를 이용하여 해를 구했기 때문이다. 〈표 4〉의 결과는 본 논문에서 제안된 패턴 생성 방법이 지지 집합의 선택 없이도 분류 정확도를 희생하지 않고 효율적으로 패턴을 생성할 수 있음을 잘 보여준다.

V 결론

본 논문에서는 LAD에서의 패턴 생성 문제를 혼합 정수 선형 계획법과 SCP를 이용하여 모형화하고 그것들을 이용한 구체적인 패턴 생성 절차를 소개하였다. 기계 학습에 통용되는 6개의 데이터 집합에 대해서 개발된 방법으로 실험한 결과, 첫째로 개발된 방법이 기존의 열거법에 기반을 둔 방법의 분류 능력을 저하시키지 않는다는 점과 둘째로 열거법에 의존하지 않고 빠르게 패턴을 생성시켰다는 점이 입증되었다.

이 결과는 본 논문에서 개발한 최적화에 기반을 둔 패턴 생성 방법이 LAD에 의한 데이터 분석을 보다 실용적이고 효율적으로 개선시켰음을 잘 보여준다고 할 수 있다.

참고문헌

[1] E. Boros, P.L. Hammer, T. Ibaraki, and A. Kogan. Logical Analysis of Numerical Data. Mathematical Programming, 79:163-190, 1997.

[2] E. Boros, P.L. Hammer, T. Ibaraki, A. Kogan, E. Mayoraz, and I. Muchnik. An Implementation of Logical Analysis of Data. IEEE Transactions on Knowledge and Data Engineering, 12:292-306, 2000.

[3] Y. Crama, P.L. Hammer and T. Ibaraki. Cause-Effect Relationships and Partially Defined Boolean Functions. Annals of Operations Research, 16:299-326, 1988.

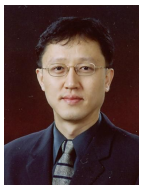
[4] P.M. Murphy and D.W. Aha. Uci repository of machine learning databases: Readable data repository. Department of Computer Science, University of California at Irvine, CA, 1994. Website at http://www.ics.uci.edu/mlearn/MLRepository.html

저 자 소 개

장 인 용

고려대학교 산업시스템정보공학과 석사과정.

〈관심분야〉 최적화, 패턴 인식, 데이터 마이닝



류 홍 서

1999년 The University of Illinois at Urbana-Champaign에서 박사학위를 취득. 미국 메릴랜드 주립대학 수학과와 미국 일리노이주립대학 기계·산업공과 tenure-track 교수직을 거쳐 현재 고려대학교 산업시스템정보공학과 교수로 재직 중

〈관심분야〉 정확한 바이오메디컬 의사결정 지원을 위한, 전·광역 최적화에 기반을 둔 데이터마이닝 및 기계학습 이론과 해법 개발