*Journal of Korean*
*Data & Information Science Society*
*2005, Vol. 16, No. 4, pp. 801~813*

# Comparison of Multiway Discretization Algorithms for Data Mining[1)]

Jeong-Suk Kim[2)] · Young-Mi Jang[3)] · Jong-Hwa Na[4)]

Abstract

The discretization algorithms for continuous data have been actively studied in the area of data mining. These discretizations are very important in data analysis, especially for efficient model selection in data mining. So, in this paper, we introduce the principles of some mutiway discretization algorithms including KEX, 1R and CN4 algorithm and investigate the efficiency of these algorithms through numerical study. For various underlying distribution, we compare these algorithms in view of misclassification rate.

***Keywords*** : Binary Splitting, Data Mining, Discretization Algorithm, Misclassification Rate, Threshold

## 1. 서론

연속형 자료에 대한 범주화 알고리즘은 크게 이진분리(binary splitting)와 다원분리(multiway splitting) 알고리즘으로 나뉘어 진다. 다원(multiway)의 의미는 범주화의 매 단계에서 두 개 이상의 분리기준값을 허용하는 것을 의미하며, 따라서 이진분리보다 융통성이 많은 방법이라 말할 수 있다. 또한 분리의 각 단계에서 분리 구간에 속한 자료들은 해당 자료들이 가장 밀집해 있는 목표변수의 특정 범주로 분류가 이루어진다. 다원분리와 관련된 연구로는 Holte (1993), Nevill-Manning et al. (1995), Bruha

1) 본 연구는 산업자원부의 지역혁신 인력양성사업의 연구결과로 수행되었음.

2) First Author : Principal Researcher(Ph. D.), Information & Communication Dept., HIRA, Seocho3-Dong, Seocho-Ku, Seoul, 137-706, Korea.
E-mail: chastity@hiramail.net

3) Doctoral Student, Dept. of Information and Statistics, Chungbuk National University, Cheongju, Chungbuk, 361-763, Korea.

4) Corresponding Author : Professor, Dept. of Information and Statistics & Institute for Basic Science Research, Chungbuk National University, Cheongju, Chungbuk, 361-763, Korea.
E-mail: cherin@cbu.ac.kr

and Berka (1993), Berka (1993a, 1993b), Berka and Bruha (1995, 1998), Kerber (1992), Kohavi and Sahami (1996), Wang and Goh (1997) 등의 연구가 있다. 이들 알고리즘 간의 효율성 비교에 대한 연구로는 Bruha and Berka (1993), Kralik and Bruha (1997) 등이 있으나, 이들의 연구는 주로 실제 자료에 대해 두 알고리즘간의 비교를 부분적으로 수행하고 있다.

본 논문에서는 최근까지 개발된 많은 다원분리 범주화 알고리즘 가운데 가장 많이 사용되고 있는 KEX, 1R, CN4 알고리즘을 소개하고, 다양한 모의실험을 통해 다원분리 알고리즘간의 효율성 비교를 수행하였다. 본 논문의 2절에서는 대표적인 다원분리 알고리즘을 소개와 함께 실제 자료에의 적용과정을 다루었다. 3절에서는 다양한 모집단 분포 하에서 모의실험을 통해 각 알고리즘에 대한 비교분석을 실시하였으며, 4절은 결론으로 구성되었다. 본 논문에서의 수행된 모든 모의실험은 이진분리 알고리즘을 다룬 선행논문에서의 결과와의 비교가 용이하도록 동일한 조건에서 모의실험을 수행하였다.

## 2. 다원분리 알고리즘과 실제자료에의 적용

### 2.1 연속형 자료에 대한 다원분리 알고리즘

이 절에서는 대표적인 다원분리 범주화 알고리즘들을 소개한다. 본 논문의 전반에 걸쳐 사용되는 자료의 형태는 $n$개의 1차원 연속형자료로써 $x_1, x_2, \cdots, x_n$으로 표기한다. 해당 알고리즘에 대한 보다 자세한 절차는 관련논문을 참고하기 바란다.

**[Algorithm Ⅰ] KEX 알고리즘**

KEX(Knowledge EXplorer) 알고리즘은 1993년 Petr Berka에 의해 고안된 것으로 연속형 데이터에 대해 다원분리를 수행한다. KEX 알고리즘을 통한 연속형 데이터의 범주화 방법은 다음과 같다.

【단계1】 전체 데이터를 오름차순으로 정렬한다.

【단계2】 각 관측값에 대하여 다음을 수행한다.

2.1 목표변수의 범주별 빈도를 계산한다.

2.2 ASSIGN 과정을 통해 모든 값을 목표변수의 특정 범주에 할당한다.

【단계3】 INTERVAL 과정을 통해 관측값들의 구간을 만든다.

여기서, ASSIGN과정과 INTERVAL과정은 각각 다음과 같다.

ASSIGN 과정 :

[단계2.2.1] 동일한 크기의 관측값들 모두가 목표변수의 특정범주에 속하면, 그 관측값에 특정범주를 할당한다.

[단계2.2.2] 동일한 크기의 관측값들이 목표변수의 모든 범주들에 분포되어 있으면, 적합도 검정을 수행한 후 유의한 차이가 있는 것으로 판명되면, 그 관측값에 가장 많은 빈도를 갖는 범주를 할당한다.

[단계2.2.3] 위의 두 가지 경우에 속하지 않는 관측값들은 “UNKNOWN″을 할당한다.

INTERVAL 과정 :

[단계3.1] 관측값들의 수열이 동일한 범주에 할당되어 있으면, 이 값들을 사용해서 구간을 만든다. [단계3.2] 구간 번째의 구간 $INT_i$가 "UNKNOWN"으로 할당되어 있고, 이웃한 두 구간 $INT_{i-1}$, $INT_{i+1}$이 같은 범주에 할당되어 있으면 세 구간을 합하여 새로운 구간 $INT_{i-1} \cup INT_i \cup INT_{i+1}$을 만든다. 구간 $INT_i$가 "UNKNOWN"으로 할당되어 있고, 이웃한 두 구간 $INT_{i-1}$, $INT_{i+1}$이 서로 다른 범주에 할당되어 있으면 구간 $INT_{i-1} \cup INT_i$과 구간 $INT_i \cup INT_{i+1}$의 적합도 검정을 수행하여 더 큰 $\chi^2$값을 갖는 구간을 선택한다. [단계3.3] 구간 $INT_i$의 하한값에 구간 $INT_{i-1}$의 상한값을 넣는다. LBOUND( $INT_i$) := UBOUND( $INT_{i-1}$)

**[Algorithm Ⅱ] CN4 알고리즘**

CN4 알고리즘은 1993년 Bruha, I.에 의해 고안되었다. CN4 알고리즘은 잘 알려진 CN2(Clark and Nibblet) 알고리즘의 확장 형태이고, 두 개 이상의 구간으로 연속형 데이터를 분리하는 알고리즘이다. CN4 알고리즘을 이용해 연속형 데이터를 범주화하는 과정은 다음과 같다.

【단계1】 전체 데이터 $A_n$을 크기순으로 정렬한다.

$$A_n = \{V_{\min}, V_{\min+1}, \cdots, V_{\max}\}.$$

【단계2】 $A_n$의 유일한 값을 $V_j \in \{V_{\min}, \cdots, V_{\max-1}\}$라 하고 다음을 계산한다.

2.1 $V_j$의 값들로 나누어지는 자료의 집단 $C_r$, $r=1, \cdots, R$에 대해서 $V \le V_j$ [$V > V_j$]의 빈도를 계산하고, 이를 $Dleft_r$ [$Dright_r$]이라 하자.

2.2 $Hleft(V_j) = Rank(Dleft_1, \cdots, Dleft_R)$를 계산한다. 이때 $V_j$는 잠재 상한이 된다. 유사하게 $Hright(V_j) = Rank(Dright_1, \cdots, Dright_R)$를 계산하고, 이때 $V_j$는 잠재 하한이 된다.

2.3 $Hleft(V_j)$가 비증가 로컬 최대값, 즉, $Hleft(V_j - 1) \le Hleft(V_j) > Hleft(V_{j+1})$이면, 경계의 배열에 $A_n \le V_j$를 추가한다. 마찬가지 방법으로 $Hright(V_j)$가 비감소 로컬 최소값이면, 경계의 배열에 $A_n > V_j$를 추가한다.

【단계3】 경계의 배열에서 모든 가능한 $V_1 < V_2$에 대해 다음을 계산한다.

3.1 각 집단 $C_r$에 대해서 구간 $V_1 < V \le V_2$내의 값 $V$의 빈도( $D_r$)를 계산한다.

3.2 $H(V_1, V_2) = Rank(D_1, \cdots, D_R)$를 계산한다.

3.3 구해진 $H(V_1, V_2)$의 크기순으로 경계의 배열에 $V_1 < A_n \le V_2$을 삽입한다.

【단계4】 가장 상위에 있는 $V_1 < A_n \le V_2$에 속하는 자료를 데이터 셋에서 삭제하고, 위의 과정을 반복한다.

위의 【단계2】 와 【단계3】 의 함수 $Rank(D_1, \cdots, D_R)$는 다음과 같은 두 가지 방

법에 의해 계산된다.

$$\text{Entropy} : \ Rank(D_1, \cdots, D_R) = \sum_r (D_r / D) \log_2 (D_r / D) .$$

$$\text{Laplacian Estimate} : \ Rank(D_1, \cdots, D_R) = (D_r + 1)/(D + R) .$$

위식에서, $D$는 모든 $D_r$의 합이고, $D_r$은 개체수가 가장 많은 집단의 $Dleft_r$값이다.

**[Algorithm Ⅲ] 1R 알고리즘**

1R(1Rule) 알고리즘은 Robert Holte가 1993년 고안한 것으로 연속형 데이터에 대해 다원분리를 수행하는 알고리즘이다. 1R 알고리즘을 이용해 연속형 데이터를 범주화하는 과정은 다음과 같다. 먼저, 범주화 대상이 되는 연속형 데이터의 관측값과 관측값들의 개별값을 각각 다음과 같이 정의하도록 하자.

관측값 $= \{x_1, x_2, \cdots, x_n\}$, 개별값 $= \{V_1, V_2, \cdots, V_m\}$.

【단계1】 전체 데이터를 크기순으로 정렬한다.

【단계2】 각 관측값에 대하여 다음을 수행한다.

2.1 $w_1 = \{V_i\}$에 속한 관측값들의 정도(accuracy)를 계산한다.

2.2 $w_1$에 개별 값들을 하나씩 포함시켜가며 정도를 계산한다.

(여기서, 정도는 구간 내에 포함된 관측값의 수와 목표변수의 특정범주에 과반수 이상 포함되는 관측값의 수의 비로써 계산된다.)

【단계3】 정도가 감소할 때까지 【단계2】를 반복한다.

【단계4】 정도가 감소했을 때의 개별값이 $V_{j+1}$이었다면, $(V_j + V_{j+1})/2$을 분리 기준값으로 선택한다.

【단계5】 전체 데이터에 대하여 【단계2】 ~ 【단계4】를 수행한다.

이 알고리즘은 하나의 구간이 단 한 개의 관측값 만을 포함할 수도 있는데, 이를 방지하기 위해 Holte는 경험적 분석에 의해 하나의 구간이 포함하는 최소 관측값의 수를 제안하였다. 데이터의 수가 50이상인 경우 하나의 구간에 포함되는 최소 관측값의 수를 6으로 설정하고, 50이하인 경우 최소 관측값의 수를 3으로 설정하여 범주화를 수행하도록 제안하였다.

## 2.2 실제자료에의 적용

다음의 표2.1의 자료는 C. Sano (1992)가 작성한 일본의 신용 평가 자료의 일부분으로 직장인 124명에 대한 근무연수(단위: 연)에 따른 신용상태 만을 발췌한 자료이다. 이 중 신용상태(목표변수)가 우량은 (+)로, 불량은 (-)로 표기하였다. 본 논문에서는 연속형 변수인 근무연수를 범주화하는 과정에 대해 살펴보기로 한다.

먼저 KEX 알고리즘에 적용하여 범주화시키는 과정을 살펴보도록 하겠다. 먼저, 전체 데이터를 정렬하고, 각 관측값에 대하여 목표변수의 범주별 빈도를 계산한 후 알고리즘의 ASSIGN 과정을 통해 목표변수의 특정 범주(+, -, UK)를 각 근무연수에 할당한다. 여기서 UN은 “UNKNOWN“을 줄여서 표현한 것이다. (표2.1의 4열과 8열을

참고할 것.)

<표 2.1> 근무연수에 따른 신용평가 자료

| 근무연수 | (+) 빈도 | (-) 빈도 | Class | 근무연수 | (+) 빈도 | (-) 빈도 | Class |
|---|---|---|---|---|---|---|---|
| 0.00 | 3 | 13 | - | 12.00 | 1 | 1 | UK |
| 1.00 | 10 | 10 | UK | 13.00 | 6 | 0 | + |
| 2.00 | 8 | 10 | UK | 14.00 | 1 | 0 | + |
| 3.00 | 6 | 1 | UK | 15.00 | 3 | 0 | + |
| 4.00 | 2 | 0 | + | 18.00 | 2 | 0 | + |
| 5.00 | 10 | 4 | UK | 20.00 | 5 | 0 | + |
| 6.00 | 3 | 0 | + | 25.00 | 4 | 0 | + |
| 7.00 | 6 | 0 | + | 27.00 | 1 | 0 | + |
| 8.00 | 1 | 0 | + | 30.00 | 1 | 1 | UK |
| 9.00 | 3 | 0 | + | 35.00 | 1 | 0 | + |
| 10.00 | 3 | 0 | + | 37.00 | 1 | 0 | + |
| 11.00 | 3 | 0 | + | | | | |

다음으로 [단계3.1]의 과정을 통해 동일한 범주에 속하는 값들을 통합하여 새로운 구간을 만든다.

<표 2.2> [단계3.1]의 수행결과

| # | 하한(LBound) | 상한(UBound) | (+)의 빈도 | (-)의 빈도 | 범주 |
|---|---|---|---|---|---|
| 1 | 0.00 | 0.00 | 3 | 13 | - |
| 2 | 1.00 | 3.00 | 24 | 21 | UK |
| 3 | 4.00 | 4.00 | 2 | 0 | + |
| 4 | 5.00 | 5.00 | 10 | 4 | UK |
| 5 | 6.00 | 11.00 | 19 | 0 | + |
| 6 | 12.00 | 12.00 | 1 | 1 | UK |
| 7 | 13.00 | 27.00 | 22 | 0 | + |
| 8 | 30.00 | 30.00 | 1 | 1 | UK |
| 9 | 35.00 | 37.00 | 2 | 0 | + |

[단계3.2]의 방법을 이용하여 범주가 UK로 할당된 관측값들이 목표변수의 특정 범수에 할당되도록 만든다.

<표 2.3> [단계3.2]의 수행결과

| # | 하한(LBound) | 상한(UBound) | (+) 빈도 | (-) 빈도 | 범주 |
|---|---|---|---|---|---|
| 1 | 0.00 | 3.00 | 27 | 34 | - |
| 2 | 4.00 | 37.00 | 57 | 6 | + |

끝으로 구간 $INT_i$의 하한값에 구간 $INT_{i-1}$의 상한값을 넣는다.

<표 2.4> [단계3.3]의 수행결과

| # | 하한(LBound) | 상한(UBound) | (+) 빈도 | (-) 빈도 | 범주 |
|---|---|---|---|---|---|
| 1 | 0.00 | 3.00 | 27 | 34 | - |
| 2 | 3.00 | 37.00 | 57 | 6 | + |

따라서 KEX 알고리즘을 이용하여 일본의 신용평가 자료의 근속연수를 범주화하면 위의 표2.4와 같이 근속연수가 3년 이하인 값을 갖는 경우와 3년 이상인 값을 갖는 경우로 나누어 볼 수 있는데, 3년 이하인 경우 (-)범주에 속하게 되고, 3년 이상인 경우 (+)범주에 속하게 된다.

다음으로 동일한 자료에 대해 CN4 알고리즘의 적용과정은 다음과 같다. 먼저 데이터를 크기순으로 정렬한 후, $D_0, D_1, Hleft(V_j)$를 계산하면 표2.5를 얻게 된다. 여기서 $Hleft(V_j)$ 의 계산은 라플라스 추정치(Laplacian estimate)를 이용하였다.

<표 2.5> 각 경계값에서의 $Hleft(V_j)$

| $V \le V_j$ | $D_0$ | $D_1$ | $Hleft(V_j)$ | $V \le V_j$ | $D_0$ | $D_1$ | $Hleft(V_j)$ |
|---|---|---|---|---|---|---|---|
| V≤0 | 3 | 13 | 0.778 | V≤11 | 58 | 38 | 0.602 |
| V≤1 | 13 | 23 | 0.632 | V≤12 | 59 | 39 | 0.600 |
| V≤2 | 21 | 33 | 0.607 | V≤13 | 65 | 39 | 0.623 |
| V≤3 | 27 | 34 | 0.556 | V≤14 | 66 | 39 | 0.626 |
| V≤4 | 29 | 34 | 0.538 | V≤15 | 69 | 39 | 0.636 |
| V≤5 | 39 | 38 | 0.506 | V≤10 | 71 | 39 | 0.643 |
| V≤6 | 42 | 38 | 0.524 | V≤20 | 76 | 39 | 0.658 |
| V≤7 | 48 | 38 | 0.557 | V≤25 | 80 | 39 | 0.669 |
| V≤8 | 49 | 38 | 0.562 | V≤27 | 81 | 39 | 0.672 |
| V≤9 | 52 | 38 | 0.576 | V≤30 | 82 | 40 | 0.669 |
| V≤10 | 55 | 38 | 0.589 | V≤35 | 83 | 40 | 0.672 |

위의 표2.5에서 근무연수가 0, 11, 27, 35일 때 $Hleft(V_j)$ 값이 로컬 최대가 됨을 알 수 있다. 다음으로 [단계3]의 과정을 수행하면 표2.6과 같다.

<표 2.6> 각 잠재분리 구간에서의 $H(V_1, V_2)$

| $V_1$ | $V_2$ | $D_0$ | $D_1$ | $H(V_1, V_2)$ |
|---|---|---|---|---|
| 11 | 27 | 1 | 23 | 0.923 |
| 11 | 35 | 2 | 26 | 0.897 |
| 0 | 27 | 39 | 80 | 0.745 |
| 0 | 35 | 39 | 83 | 0.743 |
| 0 | 11 | 38 | 57 | 0.683 |
| 27 | 35 | 1 | 3 | 0.600 |

$H(V_1, V_2)$의 값이 $H(11, 27)$일 때 0.923으로 가장 크다. 따라서 구간 $11 < A_n \leq 27$에 속하는 데이터를 전체 데이터로부터 삭제하고, 위의 과정을 반복하면 다음과 같은 최종결과가 얻어진다.

if 근무연수 ≤ 0 then class is -(불량)
else if 0 < 근무연수 ≤11 then class is +(우량)
else if 11 < 근무연수 ≤ 27 then class is +(우량)
else class is +(우량)

마지막으로 1R알고리즘의 개념을 이해하기위해 편의상 다음의 골프데이터(Quinlan, 1994)를 이용하기로 한다.

<표 2.7> 골프데이터[Temp: 기온(단위: 화씨), Class: Play여부]

| Temp | 64 | 65 | 68 | 69 | 70 | 71 | 72 | 72 | 75 | 75 | 80 | 81 | 83 | 85 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Class | P | D | P | P | P | D | P | D | P | P | D | P | P | D |

(P:Play, D:Don't Play를 의미.)

이 데이터의 크기는 50이하이므로 1R알고리즘에 의해 분리되는 각 구간이 포함하는 최소의 데이터 크기는 3이상이어야 한다. 따라서 최초 68.5이하인 값을 시작으로 정도(accuracy)를 계산해 보면 $2/3$이고, 69.5에서는 $3/4$이 되고, 70.5에서는 $4/5$로 계속 증가한다. 그러나 71.5에서는 정도가 $4/6$으로 감소하게 된다. 따라서 정도가 가장 클 때의 기온은 70.5이고, 이 값이 분리기준값으로 선택된다. 이와 같은 방법으로 분리준값을 계속 찾아나가면 최종적으로 70.5, 84.0이 분리기준값으로 선택된다. 그런데 기온이 70.5이하면 목표변수인 Class가 Play로 판정되고, 기온이 70.5이상 84.0이하의 값을 갖는 경우도 Play로 판정이 된다. 두 구간에 포함된 데이터는 목표변수의 같은 범주로 판정됨으로 두 구간을 합한다. 이로부터 1R알고리즘에 의해 최종적으로 얻

어지는 분리기준값은 84.0이고, 84.0이하인 값을 갖는 경우 Play로 판정하고, 84.0이상인 값을 갖는 경우 Don't Play로 판정한다. 참고로 동일한 자료에 대해 이진분리 알고리즘을 적용한 선행논문에서의 결과(Na & Jang, 2005)를 제시하면 표2.8과 같다.

<표 2.8> 골프데이터에 대한 분리기준값

| 1R 알고리즘 | 이진분리 알고리즘 | | |
|---|---|---|---|
| | C4.5 | QUEST | CART |
| 84.0 | 70.5 | 83.64 | 70.5 |

위의 결과를 보면 1R의 결과는 QUEST의 결과와 비슷하며, C4.5와 CART의 결과가 동일함을 알 수 있다. 그러나 이러한 현상은 본 자료에 국한된 것으로 일반적인 사항이라 말할 수는 없다.

## 3. 범주화 알고리즘들의 효율성 비교

앞 절에서 소개한 세 가지의 범주화 알고리즘들의 효율성을 비교하기 위해 본 논문에서는 다양한 모의실험 과정을 통해 알고리즘간의 비교분석을 실시하였다.

### 3.1 오분류율을 통한 효율성 비교

이 절에서는 오분류율(misclassification rate)을 통한 각 알고리즘의 효율을 비교하기로 한다. 먼저 오분류율의 계산과정은 다음과 같다. 예를 들어 표3.1의 (a)의 경우, 두 모집단 $N(0,1)$과 $N(1,1)$으로부터 각각 30개의 난수를 생성하고, 이들의 목표변수를 각각 0과 1로 설정한다. 생성된 총 60개의 난수에 대해 각 범주화 알고리즘에 적용하여 분리기준값을 찾는다. 분리기준값으로부터 나뉘어 지는 두 그룹의 자료에 대해 보다 비율이 높은 목표변수의 값을 할당한다. 각 알고리즘으로부터 구해진 분리기준값들에 의해 할당된 목표변수의 값(0 또는 1)과 실제 알고 있는 목표변수의 값을 비교하여 잘못 분류된 비율을 구하고, 이러한 과정을 10000번 반복하여 구해진 비율의 평균값을 오분류율로 정의한다. 본 논문에서의 모의실험은 Splus6 언어를 사용하였다.

표3.1에서 표3.4는 각각 모집단의 분포가 위치모형, 척도모형, 위치-척도모형, 혼합모형인 경우에 대한 다원분리 알고리즘의 오분류율을 나타내고 있다. 각 표에서 (*)표시는 효율이 가장 좋음을 나타낸다. 이 결과들을 살펴보면 다양한 모집단의 전반에서 KEX 알고리즘이 가장 우수한 것으로 나타나고 있다. 다음으로 표3.2와 표3.3의 경우 지수분포 관련 모형에서 1R 알고리즘의 효율이 상당히 좋은 것이 특징으로 나타나고 있다. 각 표에서 CART 열의 결과는 이진분리와의 비교를 위한 것으로 다음절에서 자세히 다루기로 한다.

<표 3.1> 위치모형에서의 오분류율

| 위치모형 | 두 모집단의 분포 | CART | 오분류율 | | |
|---|---|---|---|---|---|
| | | | KEX | 1R | CN4 |
| 정규분포 | (a) $N(0,1)$ *vs* $N(1,1)$ | 0.275 | 0.214 | 0.250 | 0.279 |
| | (b) $N(0,1)$ *vs* $N(3,1)$ | 0.049 | 0.033 | 0.054 | 0.072 |
| 분포 | (c) $t(2)$ *vs* $t(2)+1$ | 0.302 | 0.125 | 0.268 | 0.239 |
| | (d) $t(2)$ *vs* $t(2)+3$ | 0.117 | 0.040 | 0.116 | 0.136 |
| 균일분포 | (e) $U(0,1)$ *vs* $U(0,1)+0.2$ | 0.355 | 0.241 | 0.286 | 0.333 |
| | (f) $U(0,1)$ *vs* $U(0,1)+0.6$ | 0.167 | 0.174 | 0.170 | 0.235 |
| 지수분포 | (g) $\mathrm{Exp}(1)$ *vs* $\mathrm{Exp}(1)+1$ | 0.185 | 0.152 | 0.169 | 0.177 |
| | (h) $\mathrm{Exp}(1)$ *vs* $\mathrm{Exp}(1)+3$ | 0.025 | 0.020 | 0.025 | 0.093 |

<표 3.2> 척도모형에서의 오분류율

| 척도모형 | 두 모집단의 분포 | CART | 오분류율 | | |
|---|---|---|---|---|---|
| | | | KEX | 1R | CN4 |
| 정규분포 | (a) $N(0,1)$ *vs* $N(0,2^2)$ | 0.368 | 0.195 | 0.264 | 0.241 |
| | (b) $N(0,1)$ *vs* $N(0,4^2)$ | 0.307 | 0.107 | 0.173 | 0.176 |
| 로지스틱 분포 | (c) $L(0,1)$ *vs* $L(0,2)$ | 0.369 | 0.129 | 0.261 | 0.211 |
| | (d) $L(0,1)$ *vs* $L(0,4)$ | 0.313 | 0.075 | 0.181 | 0.166 |
| 지수분포 | (e) $\mathrm{Exp}(1)$ *vs* $\mathrm{Exp}(1/2)$ | 0.345 | 0.351 | 0.307 | 0.316 |
| | (f) $\mathrm{Exp}(1)$ *vs* $\mathrm{Exp}(1/4)$ | 0.243 | 0.281 | 0.234 | 0.260 |
| 이중 지수분포 | (g) $DE(0,1)$ *vs* $DE(0,2)$ | 0.376 | 0.177 | 0.288 | 0.235 |
| | (h) $DE(0,1)$ *vs* $DE(0,4)$ | 0.328 | 0.109 | 0.220 | 0.193 |

<표 3.3> 위치-척도모형에서의 오분류율

| 위치-척도 모형 | 두 모집단의 분포 | CART | 오분류율 | | |
|---|---|---|---|---|---|
| | | | KEX | 1R | CN4 |
| 정규분포 | (a) $N(0,1)$ *vs* $N(1,2^2)$ | 0.304 | 0.172 | 0.250 | 0.234 |
| | (b) $N(0,1)$ *vs* $N(3,4^2)$ | 0.193 | 0.085 | 0.159 | 0.167 |
| 로지스틱 분포 | (c) $L(0,1)$ *vs* $L(1,2)$ | 0.341 | 0.126 | 0.257 | 0.209 |
| | (d) $L(0,1)$ *vs* $L(3,4)$ | 0.251 | 0.064 | 0.169 | 0.168 |
| 지수분포 | (e) $\mathrm{Exp}(1)$ *vs* $\mathrm{Exp}(1/2)+1$ | 0.192 | 0.176 | 0.152 | 0.161 |
| | (f) $\mathrm{Exp}(1)$ *vs* $\mathrm{Exp}(1/4)+3$ | 0.026 | 0.021 | 0.018 | 0.060 |
| 이중 지수분포 | (g) $DE(0,1)$ *vs* $DE(1,2)$ | 0.309 | 0.156 | 0.262 | 0.231 |
| | (h) $DE(0,1)$ *vs* $DE(3,4)$ | 0.204 | 0.068 | 0.167 | 0.153 |

<표 3.4> 혼합모형에서의 오분류율

| 혼합모형 | 두 모집단의 분포 | CART | 오분류율 | | |
|---|---|---|---|---|---|
| | | | KEX | 1R | CN4 |
| 정규분포 관련 | (a) $N(1,1)$ *vs* $t(2)$ | 0.288 | 0.193 | 0.244 | 0.231 |
| | (b) $N(1,1)$ *vs* $\mathrm{Exp}(1)$ | 0.385 | 0.264 | 0.285 | 0.271 |
| | (c) $N(1,1)$ *vs* $L(0,1))$ | 0.301 | 0.186 | 0.248 | 0.246 |
| | (d) $N(0,1)$ *vs* $N(4,1)$ | 0.249 | 0.206 | 0.188 | 0.209 |
| 오염 정규분포 관련 | (e) $0.05N(0,1)+0.95N(2,1)$ *vs* $N(1,1)$ | 0.319 | 0.244 | 0.297 | 0.265 |
| | (f) $0.05N(0,1)+0.95N(3,1)$ *vs* $N(1,1)$ | 0.178 | 0.138 | 0.175 | 0.181 |
| | (g) $0.05N(0,1)+0.95N(2,1)$ *vs* $t(2)+1$ | 0.337 | 0.204 | 0.289 | 0.254 |
| | (h) $0.05N(0,1)+0.95N(2,1)$ *vs* $L(0,1)$ | 0.227 | 0.150 | 0.209 | 0.197 |

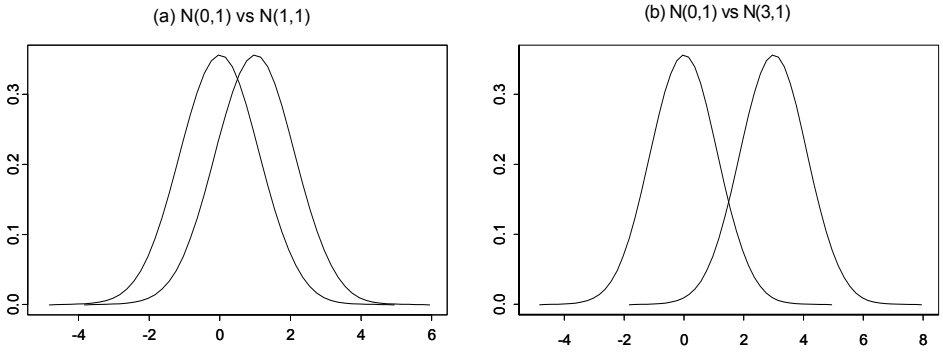
## 3.2 이진분리 알고리즘과의 비교

이 절에서는 본 논문에서 다루어진 다원분리 알고리즘과 선행 논문(Na & Jang, 2005)에서 다루어진 이진분리 알고리즘에 대한 비교에 대해 다루기로 한다. 앞 절에서

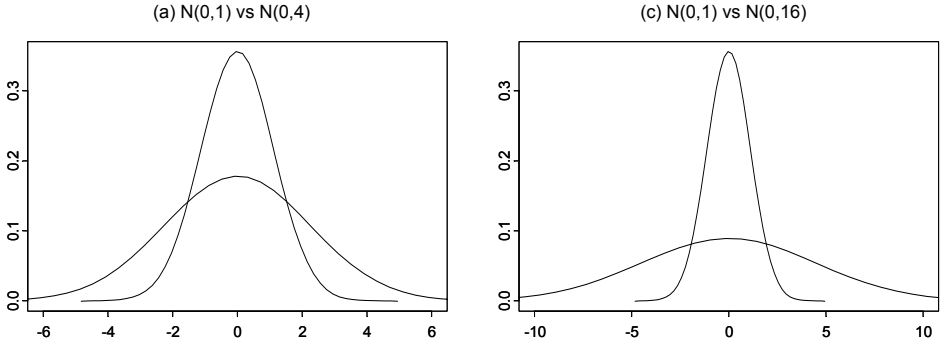
제시된 표3.1~표3.4의 모의실험 결과에서 CART 열(column)의 결과는 이진분리 알고리즘간의 비교를 위해 제시하였다. (비교의 편의를 위해 이진분리 알고리즘 가운데 가장 효율이 뛰어난 CART의 결과를 제시하였다.) 이를 살펴보면 모의실험의 전반에 걸쳐 KEX, 1R, CN4 등의 다원분리 알고리즘의 효율이 이진분리 알고리즘 보다 뛰어남을 알 수 있다. 특히 표3.1의 두 모집단이 단순히 위치 이동(location translation)의 관계에 있는 경우에는 이진분리 CART의 효율이 크게 떨어지지는 않음을 알 수 있다. 그러나 표3.2~표3.4의 경우처럼 모형이 복잡한 경우에는 다원분리의 효율이 이원분리 보다 크게 좋아짐을 알 수 있다. 이러한 현상은 다음의 그림3.1과 그림3.2로부터 좀 더 쉽게 이해될 수 있다.

아래의 그림3.1과 그림3.2는 앞 절의 모의실험에 사용된 모집단의 일부를 나타낸 것이다. 이 그림에서 알 수 있듯이 모집단의 형태가 그림3.1과 같은 위치모형의 경우보다 그림3.2의 척도 모형에서 다원분리 알고리즘의 효율이 더욱 뛰어날 것으로 쉽게 예상할 수 있다. 그 이유는 그림3.2에서처럼 두 모집단의 밀도함수의 교차점이 2개인 경우에는 여러 개의(2개 이상의) 분리기준값을 허용하는 다원분리 알고리즘이 보다 효율적임은 자명하기 때문이다.

또한 이원분리와의 비교로부터 다원분리 알고리즘의 실제의 자료분석에서도 항상 효율이 뛰어나다고 말할 수는 없다. 그 이유는 데이터마이닝 등의 실제의 자료분석 과정에서 이들 알고리즘들은 단 한 번의 수행으로 그치는 것이 아니라 여러 단계에 걸쳐 수행된 결과를 최종 모형으로 선택하기 때문이다. 그러나 이원분리를 반복적으로 수행한다 하더라도 다원분리의 결과와 동일하게 된다는 보장은 없기 때문에 실제 자료의 분석시에 알고리즘의 선택에 주의를 기울일 필요가 있다.




<그림 3.1> 표 3.1의 위치모형(a)와 (b)




<그림 3.2> 표 3.2의 척도모형(a)와 (b)

## 4. 결론

본 논문에서는 최근 개발된 연속형 자료에 대한 다원분리 알고리즘의 효율성 비교를 수행하였다. 모의실험 결과를 종합해 보면, 오분류율의 기준에서는 1R이나 CN4 알고리즘에 비해 KEX 알고리즘이 전반적으로 우수한 것으로 나타났다. 그러나 지수모형과 관련된 모집단 유형에 대해서는 1R 알고리즘의 CN4 또는 KEX 알고리즘에 비해서 효율이 뛰어난 특징을 발견할 수 있다. 또한 다원분리 알고리즘은 전반적으로 이진분리 알고리즘 보다 효율이 뛰어나다고 말할 수 있으며, 자료의 분포형태에 따라 이들 알고리즘의 효율에 차이가 크게 남을 확인하였다. 그러나 실제자료의 분석 시에는 단 한 번의 분기만으로 모형이 완성되는 것이 아니라 이들 분기가 반복적으로 수행되는 점을 고려한다면 반드시 다원분류 알고리즘이 효과적이라 말할 수는 없다. 특히 대용량의 자료를 취급해야하는 데이터마이닝 등의 영역에서는 알고리즘의 간편성과 분류과정에 소요되는 시간상의 문제도 동시에 고려되어야 할 것이다.

이상에서 다루어진 연속형 자료들의 범주화 과정에 사용되는 알고리즘들의 이해를 통해 통계분석의 효율을 높일 수 있는 알고리즘의 선택과 새로운 알고리즘의 개발이 지속되어야 할 것이다. 아울러 본 연구에서 제시된 모의실험 방법을 통한 각 알고리즘들 간의 비교연구가 좀 더 폭넓게 수행되어야 할 것으로 생각된다.

## 참고문헌

1. Berka. P. (1993a). Knowledge EXplorer : A tool for automated knowledge acquisition from data, Technical Report TR-93-03, Austrian Research Institute for AI, Vienna.
2. Berka, P. (1993b). Discretization of numerical attributes for Knowledge EXplorer, Technical Report LISP-93-03, Laboratory of Intelligent Systems.
3. Berka, P. and Bruha, I. (1998). Discretization and grouping: preprocessing steps for data mining, *Principles of Data Mining and*

*Knowledge Discovery*, 239-245.
4. Berka, P. and Bruha, I. (1995). Empirical comparisons of various discretization procedures," Technical Report LISP-95-04, Laboratory of Intelligent Systems.
5. Bruha, I. and Berka, P. (1993). Continuous classes in rule induction: Empirical comparison of two approaches, Manuscript.
6. Holte, R. C. (1993). Very simple classification rules perform well on most commonly used datatsets, Machine Learning, 11, 63-90.
7. Kerber, R. (1992). ChiMerger: Discretization of numeric attributes, Proceeding of the Tenth National Conference on Artificial Intelligence, MIT Press, 123-128.
8. Kohavi, R. and Sahami, M. (1996). Error-based and entropy-based discretization of continuous features, Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, 114-119.
9. Kralik, P. and Bruha, I. (1997). Discretizing numerical attributes in a genetic attribute-based learning algorithm, Manuscript.
10. Na, J. H., Jang, Y. M. (2005). Comparison of binary discretization algorithms for data mining, submitted.
11. Nevill-Manning, C., Holmes, G. and Ian H. (1995). The development of Holte's 1R classifier, Manuscript.
12. Wang, K. and Goh, H. C. (1997). Minimum splits based discretization for continuous features, Manuscript.

[ 2005년 8월 접수, 2005년 9월 채택 ]