Proceedings of KFIS Fall Conference, 2003. 5. 3

데이터 분포를 고려한 연속 값 속성의 이산화

Discretization of continuous-valued attributes considering data distribution

이상훈, 박정은, 오경환
서강대학교 컴퓨터학과

Sanghoon Lee, Jung-eun Park and Kyung-whan Oh
Department of Computer Science, Sogang University
E-mail : {sadclan@ailab, fayemint@ailab, kwoh@ccs}.sogang.ac.kr

ABSTRACT

본 논문에서는 특정 매개변수의 입력 없이 속성(attribute)에 따른 목적속성(class)값의 분포를 고려하여 연속형(conti-nuous) 값을 범주형(categorical)의 형태로 변환시키는 새로운 방법을 제안하였다. 각각의 속성에 대해 목적속성의 분포를 1차원 공간에 사상(mapping)하고, 각 목적속성의 밀도, 다른 목적속성과의 중복 정도 등의 기준에 따라 구간을 군집화 한다. 이렇게 생성된 군집들은 각각 목적속성을 예측할 수 있는 확률적 수치에 기반한 것으로, 각 속성이 제공하는 정보의 손실을 최소화 하는 이산화 경계선을 갖고 있다. 제안된 데이터 이산화 방법의 향상된 성능은 C4.5 알고리즘과 UCI Machine Learning Data Repository 데이터를 사용하여 확인할 수 있다.

Key words : 이산화(discretization), 데이터 분포(data distribution), 군집화(clustering) 의사결정나무(decision tree)

1. 서 론

기계학습(machine learning) 알고리즘에 적용되는 실 세계 데이터의 속성은 연속형(continuous)과 범주형(categorical)의 혼합된 형태를 가지고 있다. 하지만 대부분의 기계학습 알고리즘은 한가지 형태의 데이터만을 다룰 수 있기 때문에, 이러한 데이터를 기계학습 알고리즘에 적용시키기 위해서는 데이터 속성의 형변환이 요구된다. 일반적으로 범주형 값을 연속형으로 변환시키는 문제의 복잡성으로 인해, 연속형을 범주형으로 변환하는 기법인, 이산화(discretization) 방법을 많이 사용한다[1].

이산화를 하는 데 있어 중요한 기준은 크게 정보의 손실을 최소화 하는 것, 그리고 미지의 데이터에 대한 범주 값의 일반성을 최대화 하는 것의 두 가지로 나뉜다. 그러나 이 두 기준은 서로 상충되는데, 그것은 이산화 구간의 수가 많아질수록 정보의 손실은 적어지지만, 반대로 각 구간의 학습 집합(training set)에 대한 종속성은 증가(over-fitting)하기 때문이다. 따라서 원래 데이터의 정보를 유지하면서 동시에 일반화된 대표 값을 산출할 수 있는 적절한 수준의 이산화 구간을 찾는 것이 이산화 알고리즘의 주된 목적이다.

기존의 방법들은 보통 엔트로피(entropy), 카이스퀘어 통계학(chi-square statistics) 등을 사용하여 각 속성의 이산화된 값과 목적속성(class)간의 상관관계를 구한다. 그리고 이 값이 최대로 되는 구간을 이산화 구간으로 결정한다. 그러나 이들 대부분은 초기 구간을 결정하는 과정, 그리고 생성된 이산화 구간을 병합(일반화)하는 과정에 필요한 임계값(threshold)으로 매개변수(parameter)를 요구한다[2][3][4]. 보통 이러한 매개변수의 값에 따라 이산화 알고리즘의 성능은 큰 영향을 받으므로 최적의 매개변수를 찾는 것은 결국 또 하나의 최적화 문제를 야기한다. 따라서 본 논문에서는 매개변수의 고려 없이 최적의 이산화 구간을 결정할 수 있는 새로운 이산화 방법을

제안한다. 각 속성에 따른 목적속성의 밀도, 분포에 따라 1차원 상에서 데이터를 군집화(clustering)하고, 이때 결정되는 군집의 경계선에 따라 이산화 경계선을 결정한다. 이 결정 과정에 데이터 분포만이 고려되기 때문에, 제안한 방법은 최적화(optimization)과정 없이 하나의 프로세스로 일반화된 이산화 구간을 얻을 수 있다.

2. 관련 연구

2.1 이산화 방법의 분류

이산화 방법은 크게 이산화 과정에서 목적속성 값을 고려하는지 아닌지 여부에 따라 다음 두 가지로 분류될 수 있다[1].

1. **목적속성에 독립적인 방법(unsupervised method)**: 일정한 간격으로 구간을 정하는 equal-width 방법, 일정한 데이터 빈도로 구간을 정하는 equal-frequency 방법 등이 있다.
2. **목적속성에 의존적인 방법(supervised method)**: 속성 값과 목적속성 값과의 상관 관계를 구하기 위해 사용된 평가 함수에 따라 엔트로피 방법, 카이스퀘어 방법, 러프집합(rough set theory) 방법 등이 있다.

일반적으로 목적속성을 고려하지 않은 방법은 정확한 이산화 경계를 갖지 않기 때문에 손실되는 정보의 양이 많다. 그리고 최적의 구간 수를 결정하는 방법 또한 주관적으로 이루어지기 때문에 결정된 구간의 일반성을 보장할 수 없다. 따라서 이러한 문제점을 보완하기 위한 목적속성에 의존적인 이산화 방법들이 연구되어 왔다.

2.2 이산화 방법에 관한 기존의 연구

R. Kerber의 ChiMerge 알고리즘은 χ^2 통계학에 기반해서 연속형의 값을 이산화시킨다[2]. ChiMerge 알고리즘은 초기화와 병합의 두 단계로 구성되어 있는데, 병합단계에서는 초기화 단계에서 생성된 구간들을 종료조건을 만날 때까지 병합한다. 이 때 종료조건을 결정짓는 α (χ^2-*threshold*)는 매개변수로, 주관적으로 입력되는 값이다.

H. Liu는 ChiMerge 알고리즘의 매개변수 α 값을 자율적으로 결정하기 위해 ChiMerge 알고리즘을 변형한 Chi2 알고리즘을 제안하였다[3]. Chi2 알고리즘은 큰 α 값에서 감소하며 적절한 α 값을 찾아가는데, 종료 조건으로 *inconsistency rate* δ 를 사용한다. 비록 α 에 비해 δ 값을 다루는 것이 더 간단하지만, Chi2 알고리즘 역시 종료조건을 결정하기 위해 매개변수를 요구한다는 단점을 갖는다.

앞서 제시한 두 방법과 같이 대부분의 목적속성에 의존적인 이산화 알고리즘은 병합의 종료 조건을 결정하기 위해 매개변수 값을 요구한다. 그런데 이러한 매개변수의 값은 데이터의 특성에 따라 결정 되는 것이기 때문에 이 값의 주관적인 결정은 해당 이산화 알고리즘의 최적화된 성능을 보장하지 못한다. 따라서 이러한 주관적인 매개변수 없이 데이터 분포를 통해 자율적으로 이산화 구간을 결정지을 수 있는 방법이 요구된다.

3. DENDIS 알고리즘(DENsity based DIScretization algorithm)

연속된 속성값의 도메인(domain)을 이산화된 n개의 구간으로 자르는 문제는 목적속성의 분포에 따라 속성값의 도메인을 n개의 군집으로 군집화 하는, 군집화의 특별한 경우로 해석할 수 있다. 이산화 문제에서 각각의 이산화 구간은 최대의 유사성(similarity)을 가진 목적속성의 분포를 갖도록 조정되어야 하며, 이러한 기준은 군집 내 유사성(intra-similarity)을 그 평가 척도로 삼는 군집화의 기본 개념에 대응될 수 있다. 따라서 군집화 방법을 사용해 이산화 구간의 경계선을 찾는 것이 가능하다.

본 논문에서 제안한 DENDIS 알고리즘은 이러한 군집화 방법에 기반하여 이산화를 수행한다. 그런데 일반적인 군집화 문제와 달리 이산화 구간을 결정하는 문제에서는 각 구간이 목적속성에 관한 정보를 유지해야 하기 때문에, 군집 내 유사성을 판단하기 위한 기준으로 목적속성 밀도의 비율이 추가로 고려되어야 한다. 따라서 DENDIS 알고리즘에서는 유사성을 판단하기 위한 척도로 다음의 두 가지 기준을 사용한다. 1) 목적속성의 밀도, 2) 해당 밀도에서 각 목적속성이 차지하는 비율의 순위 값.

밀도를 통해서는 목적속성의 분포가 비교적 명확히 구분되어 있는 지점을 군집화 할 수 있으며, 각 목적속성 밀도 비율의 순위 값을 통해서는 분포가 섞여 있는 구간에 대해 목적속성의 정보 손실을 최소화하는 군집을 결정 할 수 있다. DENDIS 알고리즘은 이러한 유사성 측도로 각 속성에 대해 군집화 하고, 그 군집의 경계를 이산화 경계선으로 사용한다.

DENDIS 알고리즘은 다음과 같이 크게 세 개의 처리 단위로 구성되어 있다. 첫 번째 단계에서는 각 지점의 밀도함수를 구하고, 두 번째 단계에서는 이 밀도의 지역 최소값을 통해 초기 이산화 경계를 생성한다. 마지막 단계에

서는 앞서 생성된 구간 내에서 밀도 비율을 기준으로 구간을 세분화 시킨다.

3.1 목적속성의 분포를 반영한 밀도 함수

밀도 기반의 군집화 방법은 군집의 밀도가 주변의 밀도보다 높다는 사실에 근거하여 군집화를 수행한다. 대표적인 밀도 기반 군집화 알고리즘 중 하나인 *DENCLUE* 알고리즘은 density function을 통해 n-차원 속성 공간의 밀도를 구하고, 그 값을 이용해 군집화를 수행한다[5]. density function은 해당 좌표의 밀도 값을 구하는 함수로 다음과 같이 정의된다.

$$f_{Gauss}^{D}(x)=\sum_{i=1}^{n} e^{-\frac{d(x,x_i)^2}{2\sigma^2}} \qquad 식(1)$$

전체 구간에 대한 목적속성의 밀도를 각각의 목적속성 값과 관계없이 전체적으로 구할 경우, 각 목적속성의 값에 따른 밀도 변화는 전체 밀도 값에 잘 반영되지 않는다. 따라서 DENDIS 알고리즘은 전체 구간의 밀도를 구하기 위해 각 목적속성의 밀도를 독립적으로 구한 후 각각을 누적한 값을 사용한다. 누적밀도를 계산하는 식은 다음과 같다.

$$f_{Gauss}^{D_C}(x)=\sum_{i=1}^{n_C} e^{-\frac{d(x,x_i)^2}{2\sigma_C^2}} \qquad 식(2)$$

$$f_{Gauss}^{D_{total}}(x)=\sum_{C=1}^{k} f_{Gauss}^{D_C}(x) \qquad 식(3)$$

식(2)는 전체 분포에서 목적속성 C값의 분포를 독립적으로 고려했을 때의 밀도 값을 나타내며, 식(3)은 식(2)의 밀도 값들을 누적한 값이다. 만약 각 목적속성의 분포가 명확히 구분되어 있다면, 식(3)의 값은 그 구분된 지점에서 지역최소값(local minima)을 갖게 된다. 반대로 여러 목적속성 값이 섞여 있을 경우에는 그 분포들이 누적되어 큰 밀도 값을 갖게 된다.

3.2 지역 최소값을 통한 이산화 경계 생성

계산된 밀도 값을 사용하여 군집화 하는 방법에는 크게 두 가지가 있는데, 그것은: 1)지역최대값(local maxima)을 갖는 점들을 군집의 중심(cluster center)으로 선택하고 그 점을 기반으로 군집화를 수행하는 방법, 2)임계값 ξ 이상의 밀도 값을 갖는 연속된 점들을 군집화하는 방법이다.

본 논문에서 제안한 *DENDIS* 알고리즘은 탐색 공간이 1차원이기 때문에 일반적인 n-차원 공간에서와는 다른 방법을 사용하여 군집화를 수행할 수 있다. 그것은 군집의 중심으로부터 군집의 경계를 결정하는 것이 아닌, 지역최소값을 통해 직접적으로 경계를 결정하는 것이다. 군집의 경계가 이산화 경계와 일대일로 대응되기 때문에, 지역최소값을 사용하여 군집을 결정하는 것은 직접 이산화 경계점으로 지역최소값을 선택하는 것과 동일한 의미를 갖는다. 따라서 DENDIS 알고리즘은 우선 밀도가 지역최소값을 갖는 점을 초기 이산화 경계선으로 결정한다.

3.3 밀도 비율을 통한 이산화 경계 생성

3.2절에서 목적속성의 분포가 명확히 구분되는 점은 지역최소값을 통해 결정될 수 있음을 보았다. 그런데, 지역최소값에 의해서는 명확한 분포에 대해서만 이산화 경계를 결정지을 수 있을 뿐 그 분포가 섞여있는 구간에 대해서는 이산화 경계를 생성하지 못한다. 이러한 이유로 앞서 생성된 이산화 구간은 정보의 손실을 초래하는 큰 구간을 포함할 수 있다. 따라서 손실되는 정보의 양을 줄이기 위해 이러한 구간을 좀 더 세분화하는 과정이 필요하다.

각 목적속성 밀도 비율에 따라 구간을 세분화 하면, 목적속성의 값을 예측할 수 있는 확률적 수치가 보존된 구간을 얻을 수 있다. DENDIS 알고리즘은 이러한 목적속성 밀도 비율을 판단하기 위해 앞서 생성된 모든 구간에 대해 각 목적속성 밀도 값이 교차하는 점을 찾는다. 개별 목적속성 밀도 값이 교차할 때, 그 점을 기준으로 양 쪽은 서로 다른 목적속성의 주도적인 영향을 받기 때문에 이러한 점들을 경계로 구간을 나누는 것은 정보 손실을 최소화하는 일반화된 이산화 구간을 보장한다.

지금까지 설명한 방법을 간략한 알고리즘으로 표현하면 다음과 같다. 알고리즘은 크게 밀도 계산, 지역최소값 검사, 각 밀도의 교차점 검사 부분으로 구성되어 있다.

```
DENDIS algorithm
For each continuous attribute a {
    For each target class c {
        calculate_each_density d(c,a);
    }                          //density of each class
    total_den(a):=TOTAL(d(c,a));
}                          //total density of all class
For each continuous att{
    cut_point set(a)
    :=LOCAL_MIN(total_den(a));
            //step1: create initial cut point set
  For each cut_point set(a){
      ADD(set(a),CROSS(d(c,a));
}}                         //step2: add cut points
```

4. 실험 및 결과

제안한 DENDIS 알고리즘의 성능을 검증하기 위해 UCI Machine Learning Data Repository의 데이터를 통해 예측 정확도를 측정하였다[6]. 실험은 Iris, Breast cancer, Heart diseases, Balance 등의 데이터를 사용해 수행되었다. 각각의 데이터를 학습 집합(training set) 60%와 검증 집합(validation set) 40%로 분리하였으며, 이 때 각 학습 집합과 검증 집합은 원래의 데이터와 동일한 목적속성 비율을 갖도록 임의 추출하였다.

의사결정나무(decision tree)는 범주형 데이터만을 처리할 수 있지만, 이를 구현한 알고리즘 중 하나인 C4.5는 연속형 속성값을 처리하기 위한 binarization이라는 이산화 방법을 알고리즘 안에 포함하고 있어 연속형 값을 갖는 데이터를 바로 처리할 수 있다. 본 실험에서는 비교를 위해 원래의 연속형 값과 DENDIS 알고리즘을 통해 나온 범주형 값을 각각 C4.5 알고리즘을 통해 학습한 후, 그 예측 정확도를 측정하였다. C4.5에서 사용한 binarization 방법은 다른 이산화 방법과 비교했을 때, 거의 비슷한 정도의 정확도를 보여주기 때문에 이를 통한 비교로 제안한 방법의 성능을 검증할 수 있다[7]. 표.1에 C4.5를 사용한 예측 정확도가 나타나 있다.

표. 1. DENDIS 알고리즘의 예측 정확도

	iris	breast	heart	balance
Binarization	97.64%	**95.07%**	75.44%	80.51%
DENDIS	**98.31%**	93.07%	**80.95%**	**81.23%**

표에서 볼 수 있는 것처럼 DENDIS 알고리즘을 통해 이산화된 값은 복잡한 최적화 과정이 없었음에도 불구하고 전반적으로 좋은 예측 정확도를 보여주고 있다. 그 이유는 이산화 과정에서 목적속성을 예측하기 위해 필요한 속성값의 정보를 각 이산화 구간이 적절히 분리해 주었기 때문이다. 또 데이터의 분포가 비교적 잘 분리되어 있는 영역 뿐 아니라 그 분포가 섞여 있는 영역에 대해서도 각 목적 속성 값의 분포를 고려해줌으로 인해 정보의 손실을 최소화하는 이산화 경계를 결정지어 줄 수 있었다.

5. 결 론

본 논문에서는 연속형 값을 범주형 값으로 변환시키기 위한 새로운 이산화 방법을 제안하였다. 기존의 평가함수 기반 이산화 방법은 이산화 구간을 결정하기 위해 매개변수를 요구한다는 단점을 갖는다. 본 논문에서는 이러한 문제를 해결하기 위해 데이터 분포만을 통해 적절한 이산화 경계를 결정지어 줄 수 있는 새로운 알고리즘을 제안하였다.

DENDIS 알고리즘은 목적속성의 밀도와 그 분포 비율을 기준으로 이산화 경계를 결정한다. 이 과정에서 구간을 조정하는 단계 없이 최적의 이산화 구간이 결정되기 때문에 어떠한 매개변수 값도 요구되지 않는다. 실험 결과를 통해 제안한 알고리즘의 타당성을 확인할 수 있었다.

감사의 글

본 연구는 과학 기술 부 주관 뇌 신경 정보학 사업에 의해 지원되었음.

6. 참고문헌

[1] Ian H. Witten, Eibe Frank, "Data Mining", Morgan Kaufmann Publishers, 2000, page(s): 238-246

[2] Ren-Pu Li, Zheng-Ou Wang, "An entropy-based discretization method for classification rules with inconsistency checking", Machine Learning and Cybernetics, 2002. Proceedings. 2002 International Conference, On page(s): 243- 246

[3] R. Kerber. "ChiMerge: Discretization of numeric attribute." In Proc. Tenth National Conf. on Artificial Intelligence (AAAI-92), San Jose, CA, 123-127, 1992.

[4] H. Liu, R. Setiono, "Feature selection via discretization", IEEE Transactions on Knowledge and Data Engineering, vol.9, page(s): 642-645, 1997

[5] J. Han and M. Kamber, "Data Mining Conceip and Techniques", Morgan Kaufmann Publishers, 2001, page(s): 363-369

[6] http://www.ics.uci.edu/~mlearn

[7] T. Elomaa, J. Rousu, "General and Efficient Multisplitting of Numerical Attributes", Kluwer Academic Publishers, 1999