creative commons

COMMONS DEED

저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:

저작자표시. 귀하는 원저작자를 표시하여야 합니다.

비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.

변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 이용허락규약(Legal Code)을 이해하기 쉽게 요약한 것입니다.

Disclaimer

dCollection

공학석사학위논문

연속형 변수의 구간 별 임베딩과 클러스터링을 통한 다변수적 이산화 방법

Multivariate Discretization through Interval-wise Embedding and Clustering of Continuous Variables

2018년 2월

서울대학교 대학원
산업공학과
김 태 욱

초 록

이산화는 연속형 변수를 이산 변수로 변환하는 전처리 과정이다. 이산화를 수행하는 목적은 해석력 있는 기계학습 모델들이 입력 변수의 형태를 이산 변수로 요구하는 경우가 많기 때문이다. 기계학습 모델이 의사결정에 활용되는 의료, 보험과 같은 도메인에서는, 모델의 예측에 대한 해석이 필수적이다. 해당 도메인들에서는 연관규칙기반 분류기, 의사결정나무 등과 같은 해석력 있는 모델들이 많이 사용되는데, 이와 같은 모델들의 특징은 연속형 입력변수에 대한 이산화를 요구한다는 것이다. 이산화 결과는 모델의 예측성능이나 해석력에 많은 영향을 미친다. 따라서 해석력 있는 기계학습을 위한 첫 단계는 정교한 이산화 방법의 적용이라고 할 수 있다. 본 연구에서는 임베딩 기법을 활용한 비지도학습 기반의 다변수적 이산화 방법을 제안한다. 비지도학습 기반의 다변수적 이산화 방법은 클래스 변수를 필요로 하지 않으며, 변수 간의 상호작용 패턴을 보존한다. 지도학습을 위한 전처리로서 사용될 수 있음은 물론, 연관패턴분석이나 이상탐지와 같은 비지도학습을 위한 전처리로도 활용될 수 있다는 범용성 면에서 비지도학습 기반의 다변수적 이산화 방법은 다른 이산화 방법에 비해 장점이 있다. 그러나 기존방법들은 연속형 변수와 범주형 변수 간의 상호작용을 보존하지 못하는 한계가 있다. 제조, 의료, 보험 등 대부분의 도메인에서 다루어지는 데이터가 혼합형임을 고려할 때, 이산화 방법이 혼합형 데이터를 제대로 다루지 못하는 것은 문제가 될 수 있다. 제안방법은 이러한 문제를 임베딩 기법을 통해 해결한다. 먼저 각 연속형 변수를 단위구간으로 이산화 한다. 그 후, 변수 간의 상호작용 정보를 활용하여 단위구간의 임베딩을 학습한다. 마지막으로, 학습된 임베딩들에 대해 클러스터링을 수행하여 유사한 단위구간들을 병합한다. 실험을 통해 제안방법이 변수 간의 상호작용을 보존할 수 있으며, 특히 기존방법과 달리 연속형 변수와 범주형 변수 간의 상호작용도 보존할 수 있음을 확인하였다. 많은 도메인에서 다루어지는 데이터가 혼합형임을

고려할 때, 제안방법이 기계학습 모델의 해석력이 요구되는 다양한 도메인에서 범용 적으로 활용될 수 있을 것이라 기대한다.

주요어: 다변수적 이산화, 임베딩, 혼합형 데이터
학 번: 2015-22856

목 차

1. 서론 ····· 1
2. 관련 연구 ····· 7
2.1. 이산화 방법의 분류 ····· 7
2.1.1. 지도학습/비지도학습 기반 방법 ····· 7
2.1.2. 단변수적/다변수적 방법 ····· 8
2.2. 비지도학습 기반의 다변수적 이산화 방법 ····· 10
2.3. Word2Vec ····· 14
3. 제안방법 ····· 17
3.1. 연속형 변수의 단위구간 생성 ····· 20
3.2. 단위구간의 임베딩 학습 ····· 21
3.3. 유사한 단위구간의 병합 ····· 23
3.3.1. 단위구간의 임베딩 간 pairwise distance matrix ····· 23
3.3.2. 단위구간의 클러스터링 및 병합 ····· 24
4. 실험 ····· 26
4.1. 실험설계 ····· 26
4.2. 실험데이터 ····· 28
4.2.1. 연속형 변수 간 상호작용이 존재하는 합성데이터 ····· 28
4.2.2. 연속형 변수와 범주형 변수 간 상호작용이 존재하는 합성데이터 ····· 29
4.2.3. 클래스 레이블이 존재하는 혼합형의 실제데이터 ····· 31
4.3. 실험결과 ····· 32
4.3.1. 연속형 변수 간 상호작용이 존재하는 합성데이터 ····· 32
4.3.2. 연속형 변수와 범주형 변수 간 상호작용이 존재하는 합성데이터 ····· 35
4.3.3. 클래스 레이블이 존재하는 혼합형의 실제데이터 ····· 37
5. 결론 ····· 40
참고문헌 ····· 41

표 목차

[표 1] 클래스 레이블이 존재하는 혼합형의 실제데이터 ····· 31
[표 2] 이산화 방법 별 전처리 이후의 분류 성능(HR 데이터) ····· 33
[표 3] 이산화 방법 별 전처리 이후의 분류 성능(UCI 데이터) ····· 38
[표 4] 이산화 방법 간 성능 비교 ····· 38

그림 목차

[그림 1] 제안방법 소개 ····· 3
[그림 2] 단변수적 이산화 방법의 한계 ····· 8
[그림 3] MVD의 이산화 방법 ····· 11
[그림 4] UCPD의 이산화 방법 ····· 12
[그림 5] CBOW 모델 ····· 14
[그림 6] 제안방법의 notation ····· 17
[그림 7] 제안방법의 pseudo code ····· 18
[그림 8] 단위구간의 임베딩 모델 ····· 21
[그림 9] 단위구간의 임베딩 학습을 위한 input, target 생성 ····· 22
[그림 10] 단위구간의 클러스터링 및 병합 ····· 24
[그림 11] HR 데이터의 3차원 산점도 ····· 28
[그림 12] 연속형 변수와 범주형 변수 간 상호작용이 존재하는 합성데이터 ····· 30
[그림 13] 이산화 방법 별 HR 데이터의 이산화 결과 ····· 34
[그림 14] 이산화 방법 별 합성데이터에 대한 이산화 결과 ····· 36
[그림 15] 학습된 단위구간 임베딩의 2차원 시각화 결과 ····· 37

1. 서론

이산화란 연속형 변수를 이산 변수로 변환하는 데이터 전처리 과정이다. 연속형 변수의 정의역을 몇 개의 중복되지 않는 구간으로 나눈 후, 각 구간을 원 변수의 값이 포함되는지 그 여부를 나타내는 이산 변수로 표현한다[7].

이산화의 목적은 해석력 있는 기계학습을 수행하기 위함이다. 최근에는 제조, 의료, 보험, 마케팅 등 거의 모든 도메인에서 기계학습을 활용하고 있다. 그러나 도메인에 따라 활용 가능한 기계학습 모델의 범위에 제약이 있는 경우가 있다.

예를 들어 기계학습 모델을 통해 보험 청구 금액에 대한 심사를 수행한다고 한다고 하자. 이러한 심사의 경우, 모델을 통해 청구 금액의 적합 또는 부적합만을 판단하는 것으로는 불충분하다. 청구가 부적합한 것으로 판단되었다면, 그 이유에 대한 설명이 실무적으로 필요하기 때문이다. 또 다른 예로, 모델 기반으로 의학적인 진단을 보조한다고 하자. 모델이 판독 결과에 대한 근거를 제공하지 않는다면, 모델의 출력 값만을 바탕으로 처방을 내리는 것은 매우 위험할 것이다.

따라서 기계학습 모델이 의사결정에 활용되는 도메인에서는, 모델의 예측 결과를 해석할 수 없는 모델은 활용하기 곤란하다[12, 21]. 이는 딥러닝과 같은 블랙박스 모델들이 좋은 예측 성능을 보임에도 불구하고, 모델 기반의 의사결정에 전적으로 활용되지 못하고 있는 이유이다.

모델의 예측 결과에 대한 해석이 중요한 도메인에서는 연관규칙기반 분류기, 의사결정나무, 나이브 베이즈와 같이 해석력 있는 알고리즘들이 많이 사용되고 있다[2, 9, 10]. 한편으로는 블랙박스 모델의 예측 결과에 대해 해석을 수행하는 방향으로도 연구가 수행되고 있는데, 근래에는 LIME [21]과 같은 방법이 주목을 받고 있다.

이와 같은 알고리즘들은 연속형 변수를 이산화 하여 입력하는 것을 요

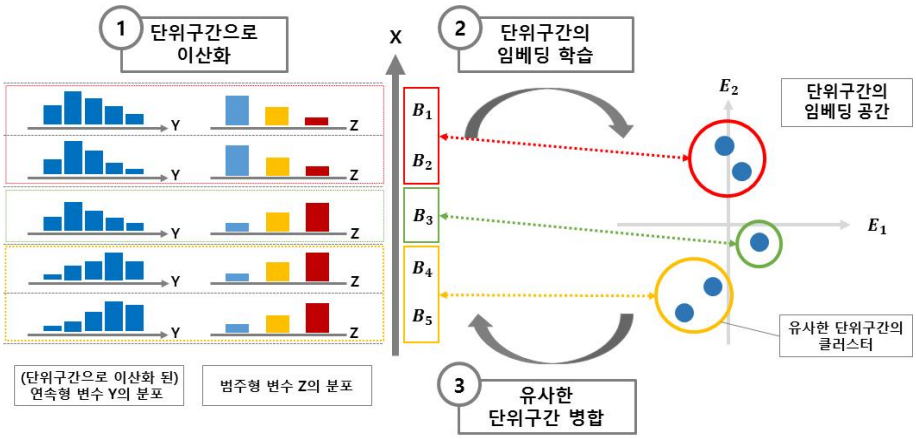
구하며, 그 성능도 이산화 결과에 많은 영향을 받는다. 연관규칙기반 분류기나 LIME의 경우 연속형 변수를 이산화 하는 것을 요구한다. 의사결정나무와 나이브 베이즈의 경우, 연속형 변수를 그대로 사용하여 모델링 하는 것보다 이산화 하여 사용했을 때 성능이 더 좋은 것으로 알려져 있다[13].

이산화는 모델의 분류/예측 성능뿐만 아니라, 모델의 해석력에도 많은 영향을 미친다. 만약 이산화 방법이 의미적으로 동질적이지 않은 변수 구간을 생성하면, 변수의 효과를 해석하는 데 있어서도 어려움이 발생할 것이다. 예를 들어 어떤 질병의 발병확률을 예측한다고 했을 때, 나이(age) 변수가 [0세, 60세), [60세, 80세)와 같이 이산화 된다면 나이 변수의 효과를 해석하기 곤란할 것이다. 0세에서 60세 사이의 인구는 상당히 이질적인 신체 특성을 가지고 있기 때문이다. 그러나 반대로, 너무 많은 구간으로 변수를 이산화하여도 모델링 결과를 해석하는데 있어 좋지 않을 것이다. 연관규칙기반 분류기를 사용한다고 했을 때, 많은 수의 규칙이 의미상 중복될 것이기 때문이다. 따라서 해석력 있는 기계학습을 위한 첫 단계는 정교한 이산화 방법을 통한 전처리라고 할 수 있다. 바람직한 이산화 방법은 이질적인 영역을 서로 다른 구간으로 분할하되, 동질적인 영역은 분할하지 않아야 한다.

이산화 방법은 (1)지도학습/비지도학습(supervised/unsupervised) 기반 방법, (2)단변수적/다변수적(univariate/multivariate) 방법으로 분류할 수 있으며, 본 연구에서 제안하는 방법은 비지도학습 기반의 다변수적 이산화 방법에 해당한다. 제안방법을 포함한 기존 비지도학습 기반의 다변수적 이산화 방법들[1, 16, 18]의 공통적인 접근 방법은, 변수 간의 상호작용 정보가 이산화 과정에서 보존되도록 하는 것이다. 즉, 이산화 되는 변수의 영역 중에서 다른 변수와의 상호작용 패턴이 유사한 영역이 하나의 구간에 속하도록 이산화 하는 것이다.

개념적인 예시로 [그림 1]에서 연속형 변수 X를 이산화 하는 방법을 살펴보자. 먼저 변수 X의 영역을 여러 개로 분할한다. 각 영역을 변수 X의

단위구간(basic interval)이라 정의하자. 그러면 변수 X의 단위구간 $B_1, ..., B_5$ 별로 변수 Y와 변수 Z가 갖는 값의 분포를 얻을 수 있다. 그 후, 변수 X의 단위구간들 중에서 다른 변수와의 상호작용 패턴이 유사한 단위구간들을 하나의 구간으로 병합함으로써 이산화가 완료된다.



[그림 1] 제안방법 소개

이러한 개념을 일반화하여 기술하면 다음과 같다. 각 변수를 $X_i\,(i \in 1, ..., N)$, 변수 X_i의 단위구간들을 순서대로 $B_j^{(i)}\,(j \in 1, ..., K)$라 두자. 그러면 변수 X_i의 j번째 단위구간인 $B_j^{(i)}$에 속할 때 다른 변수들이 갖는 확률밀도함수를 수식 (1.1)과 같이 표현할 수 있다. 편의상 수식 (1.1)을 $p(-X_i \mid B_j^{(i)})$라 표현하고, 분포 간의 유사함을 측정하는 거리 척도를 D, 적절한 threshold δ가 주어졌다고 하자. 그러면 변수 X_i의 두 단위구간 $B_x^{(i)}$와 $B_y^{(i)}$가 유사하다는 것은 수식 (1.2)를 만족하는 것으로 정의할 수 있다. 그러나 수식 (1.2)에서 다변수 분포 간의 유사도 척도인 D를 정의하는 것은 어려운 문제이며, 연속형 변수와 범주형 변수가 혼재되어 있을 경우 이는 더욱 어려워진다.

$$p(X_1, X_2, ..., X_{i-1}, X_{i+1}, ..., X_N \mid X_i \in B_j^{(i)}) \qquad (1.1)$$

$$B_x^{(i)} \approx B_y^{(i)},\ \text{if}\ \ D(p(-X_i \mid B_x^{(i)}), p(-X_i \mid B_y^{(i)})) < \delta,\ (x,\ y \in 1, ..., K) \quad (1.2)$$

기존의 비지도학습 기반의 다변수적 이산화 방법들은 구간 간의 유사도 측정을 위해 연관패턴분석[1,16], PCA[16]와 같은 기법들을 활용하거나, 구간 간 유사도 척도를 개발하여 사용하였다[18]. 그러나 이들의 한계점은 연속형 변수와 범주형 변수 간의 상호작용을 제대로 고려하지 못한다는 것이다. UCPD[16]는 이산화 하는 변수와 다른 변수들 간의 상호작용을 고려함에 있어, 연속형 변수와의 상호작용과 범주형 변수와의 상호작용을 순차적으로 고려한다. 이는 4장의 실험결과에서 확인할 수 있듯이, 연속형 변수와 범주형 변수 사이에 존재하는 상호작용을 제대로 반영할 수 없는 원인이 된다. IPD[18]의 경우 구간 간의 유사도 척도를 정의하고, 이를 활용하여 이산화를 수행한다. 그러나 IPD에서 사용되는 유사도 척도는 범주형 변수와의 상호작용은 고려할 수 없다는 한계가 있다.

제조, 의료, 보험, 마케팅 등 거의 모든 도메인의 데이터가 연속형 변수와 범주형 변수가 혼재된 혼합형 데이터(Mixed type data)라는 사실을 고려할 때, 이산화 방법이 연속형 변수와 범주형 변수 간의 상호작용을 보존하지 못한다면 곤란할 것이다.

제안하는 이산화 방법은 혼합형 데이터에서 변수 간의 상호작용을 보존하기 위해 임베딩(embedding) 기법을 활용한다. 제안방법은 [그림 1]과 같이, 세 단계로 이루어진다.

첫 번째로, 각 연속형 변수를 여러 개의 단위구간으로 이산화 한다. 그리고 모든 변수에 대해 dummy coding을 수행한다. 이를 통해 연속형 변수의 각 단위구간, 그리고 범주형 변수의 각 범주에 대한 dummy 변수들을 생성한다.

두 번째로, 각 변수의 단위구간 $B_j^{(i)}$ 별로, $p(-X_i \mid B_j^{(i)})$에 대한 정보를 표현

하는 임베딩을 학습한다. 임베딩을 위한 모델은 Word2Vec[17]과 유사한 형태로, 한 쌍의 embedding layer와 decoding layer를 가지는 신경망 모델이다.

모델은 $n-1$개 변수의 값으로 나머지 1개 변수의 값을 예측한다. 이러한 과정을 통해 학습된 embedding layer의 각 열벡터는 연속형 변수의 단위구간 혹은 범주형 변수의 범주에 대한 임베딩으로, 다른 변수와의 상호작용 정보를 표현한다. 즉, 연속형 변수의 단위구간 $B_j^{(i)}$에 대한 임베딩은 $p(-X_i \mid B_j^{(i)})$ 정보를 표현하게 된다.

임베딩 학습 과정에서 모든 변수는 그 유형에 관계없이 동일한 공간상의 임베딩으로 다루어지며, 이 임베딩 간의 연산을 통해 모든 변수 간의 상호작용이 동시에 고려된다. 따라서 학습된 연속형 변수 단위구간의 임베딩은, 다른 연속형 변수와의 상호작용은 물론 범주형 변수와의 상호작용 정보도 표현할 수 있다.

나아가, 각 단위구간이 동일한 공간상의 임베딩으로 표현되었기 때문에 $D(p(-X_i \mid B_x^{(i)}),\ p(-X_i \mid B_y^{(i)}))$를 단위구간의 임베딩 간 cosine 거리를 통해 측정할 수 있다. 즉, 혼합형 데이터에서 연속형 변수의 두 단위구간 간의 유사함을 일반적인 거리 척도를 통해 측정할 수 있는 것이다.

마지막으로, 단위구간 간 유사도를 바탕으로 클러스터링을 수행하고, 동일한 클러스터에 속한 단위구간들을 병합하여 하나의 구간으로 만듦으로써 이산화가 완료된다.

제안방법의 성능 평가는 각 변수 별로 적절한 이산화 결과를 알고 있는 데이터에 대해, 제안방법이 적절한 결과를 도출할 수 있는지 평가하는 방식으로 수행한다. 적절한 결과는 다음 장의 [그림 2]와 같이 각 구간의 패턴이 동질적이면서, 동시에 동질적인 구간이 분할되지 않도록 이산화 하는 것이다. 이에 대해서는 4장에서 자세히 다룬다.

본 연구의 기여는 두 가지이다. 첫 번째로, 임베딩 기법을 활용한 새로운 방식의 비지도학습 기반의 다변수적 이산화 방법을 제안하였다. 제안

방법은 연속형 변수의 이산화를 임베딩을 통해 수행한 최초의 접근이다. 두 번째로, 제안방법은 변수 간의 상호작용을 보존할 수 있으며, 특히 기존방법과 달리 연속형 변수와 범주형 변수 간의 상호작용도 보존할 수 있다. 현실에 존재하는 많은 데이터가 혼합형 데이터임을 고려하면, 제안방법이 유용하게 활용될 수 있을 것이다.

본 논문의 구조는 다음과 같다. 2장에서는 관련연구를 다룬다. 3장에서는 제안방법에 대해 설명하고, 4장에서는 실험과 그 결과를 제시한다. 5장에서는 결론에 대해 다루고, 6장은 참고문헌으로 구성되어 있다.

2. 관련연구

2.1. 이산화 방법의 분류

이산화는 일반적으로 정보손실을 발생시킨다. 이산화 방법들은 데이터의 패턴을 최대한 보존하면서, 데이터를 간결히 표현하는 방법을 찾는 것을 목표로 한다. 이산화 방법을 분류하는 기준은 다양한데, 본 연구에서 제안하는 방법은 비지도학습 기반의 다변수적 이산화 방법으로 분류할 수 있다. 이 절에서는 지도학습/비지도학습 기반 이산화 방법과, 단변수적/다변수적 이산화 방법에 대해 설명한다.

2.1.1. 지도학습/비지도학습 기반 방법

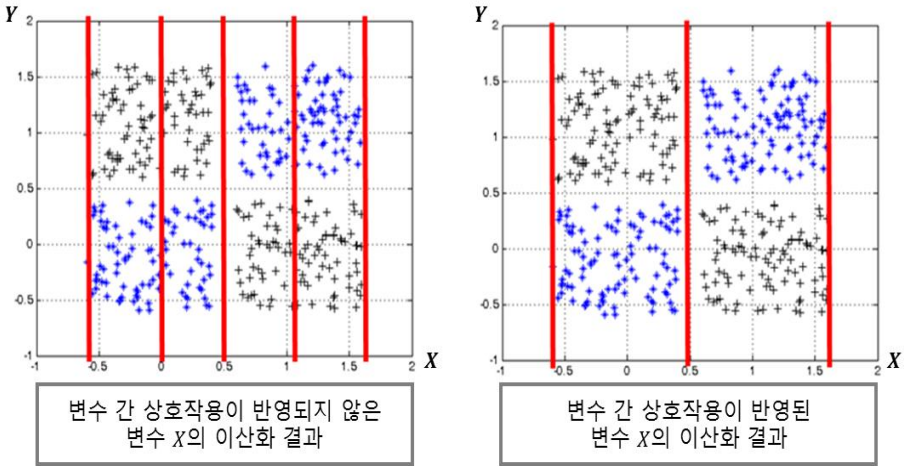
이산화 방법은 클래스 변수를 사용하는지 여부에 따라 지도학습/비지도학습 기반 방법으로 분류할 수 있다. 지도학습 기반 방법은 entropy[4, 6]나 chi-square[11] 등으로 측정되는 각 구간의 클래스 불순도를 최소화하는 방식으로 이산화를 수행한다.

반면 클래스 변수를 활용하지 않는 비지도학습 기반의 이산화 방법은 통계치[23, 25]나 변수 간의 상관관계[14]나 클러스터링[3] 등을 활용한다. 지도학습 기반 방법은 비지도학습 기반 방법에 비해, 이산화 수행 후 분류 모델링에서 더 좋은 성능을 보이는 경향이 있다[4]. 그러나 지도학습 기반 이산화 방법은 클래스 변수가 없는 상황에서는 사용할 수 없기 때문에, 그 활용범위에 제약이 있다.

제안방법은 비지도학습 기반의 이산화 방법으로, 변수 간의 상호작용 정보를 활용하여 이산화를 수행한다. 제안방법은 지도학습을 위한 전처리로 사용될 수 있음은 물론, 클래스 변수가 없는 상황에서도 연관패턴 분석이나 이상탐지와 같은 비지도학습을 위한 전처리로서도 사용될 수 있다.

2.1.2. 단변수적/다변수적 방법

이산화 방법을 분류하는 또 다른 기준은 단변수적인 방법과 다변수적인 방법으로 나누는 것이다. 단변수적 방법은 모든 변수를 독립적으로 이산화하고, 다변수적 방법은 변수 간의 상호작용을 고려하여 이산화를 수행한다. 대표적인 단변수적 방법으로는 EqualWidth, EqualFreq[23]와 같은 방법들이 있다. EqualWidth는 해당 변수의 최소값과 최대값 사이의 구간을 등 간격으로 이산화 하는 방법이며, EqualFreq는 하나의 구간에 동일한 빈도의 데이터가 포함되도록 이산화 하는 방법이다.



[그림 2] 단변수적 이산화 방법의 한계

단변수적 방법들의 한계점은, 이산화 과정에서 변수 간의 상호작용 패턴이 손실될 수 있다는 것이다. 예를 들어 [그림 2]와 같은 형태의 데이터를 고려해보자. 데이터는 그래프에서 가로축으로 표현되는 연속형 변수 X, 세로축으로 표현되는 연속형 변수 Y, 그리고 각 데이터 포인트의 모양(+, •)로 표현되는 범주형 변수 Z를 가진다. 가로축의 변수 X를 이산화한다고 하였을 때, 변수 X와 Z와의 상호작용을 고려하면 [그림 2]의 오른쪽과 같은 형태로 이산화 되는 것이 적절할 것이다. 그러나 변수 간의 상호작용을 고려하지 않는 EqualWidth 방법을 사용하면, [그림 2]

의 왼쪽과 같이 변수 간 상호작용이 반영되지 않은 이산화 결과를 도출할 수 있다. 이러한 한계를 극복하기 위해 다변수적 방법들이 연구되었으며, 제안방법도 다변수적 방법에 해당한다.

2.2. 비지도학습 기반의 다변수적 이산화 방법

제안방법을 포함한 비지도학습 기반의 다변수적 이산화 방법들의 공통적인 접근 방법은, 다른 변수들과의 상호작용 패턴이 유사한 영역이 하나의 구간에 속하도록 이산화 하는 것이다.

비지도학습 기반의 다변수적 이산화 방법은 분류, 예측과 같은 지도학습은 물론, 연관패턴분석이나 이상 탐지와 같은 비지도학습에도 활용 가능하며, 이산화 과정에서 정보손실이 적다는 점에서 유용하다. 그러나 그 유용성에 비해 개발된 방법은 많지 않은데, 현재까지 개발된 방법으로는 MVD[1], UCPD[16], IPD[18]가 존재한다.

MVD는 연속형 변수를 여러 개의 단위구간들로 이산화 한 후, 유사한 인접구간을 재귀적으로 병합하는 방식으로 이산화를 수행한다. 인접구간간의 유사함은 연관패턴분석을 통해 판단된다. 각 변수를 transaction의 item이라고 간주하고, 인접한 두 구간에 속하는 데이터에 대해 frequent itemset mining을 수행한다. 만약 모든 item의 support가 두 구간에서 큰 차이가 없으면 둘은 유사한 구간으로, 하나라도 support의 차이가 크면 유사하지 않은 구간으로 판단한다.

예를 들어 어떤 데이터의 연속형 변수 X_1의 두 단위구간 G_1, G_2 간에 유사함을 판단하는 과정은 다음과 같다. 먼저 변수 X_1의 값이 G_1, G_2에 속하는 데이터에 대해, X_1 변수 이외에 다른 변수들의 frequent itemset을 찾는다. 이 item 중 하나가 $X_2 = A$라고 할 때, X_1과 X_2 변수에 대해 contingency table을 그리면 [그림 3]과 같다. 그러면 X_1이 G_1에 속할 때 $X_2 = A$의 support와, G_2에 속할 때 $X_2 = A$의 support 차이를 구할 수 있다. Frequent itemset의 모든 item들에 대해, 이와 같은 방식으로 support 차이를 구할 수 있다. 이 중 하나라도 support 차이가 기준값을 넘는 경우, 두 구간을 유사하지 않은 구간으로 판단한다. 반대로 모든 item에서 support의 차이가 기준값을 넘지 않으면, 두 구간을 병합한다.

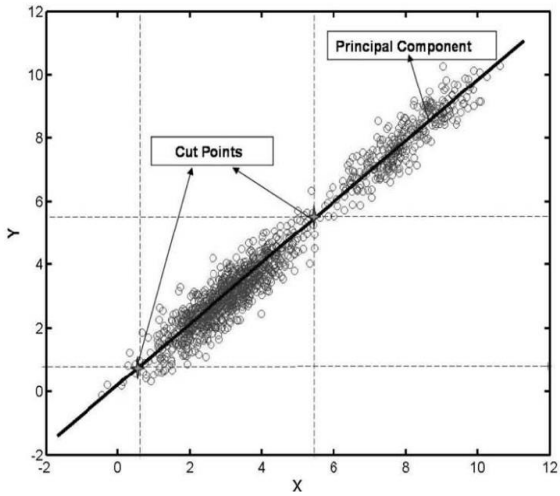
MVD의 한계점은 worst case에 계산 량이 변수 개수에 지수적으로 증가한다는 것인데[1], 이는 각 변수의 모든 인접구간 쌍에 대해, 앞서 설명한 바와 같이 frequent itemset mining을 수행해야하기 때문이다.

	$X_1 = G_1$	$X_1 = G_2$
$X_2 = A$	20	25
$X_2 \neq A$	80	75
$support(X_2 = A, G)$	0.2	0.25

[그림 3] MVD의 이산화 방법

UCPD는 PCA를 활용하여 이산화를 수행하는 방법이다. [그림 4]와 같이 데이터의 주성분을 구한 뒤, 주성분 축을 데이터의 밀도를 바탕으로 이산화 한다. 그리고 그 결과를 원 변수의 축에 전사하여 이산화를 수행한다. 이 방법은 PCA에 기반 하기 때문에, MVD보다 고차원의 데이터를 다루기 용이하다는 장점이 있다.

연속형 변수와 범주형 변수가 모두 존재할 경우, 다음과 같이 이산화 방법이 변경된다, 우선 연속형 변수들에 대해 PCA를 수행한다. 다음으로, PCA를 통해 얻은 주성분 축을 EqualFreq 방법을 사용하여 여러 개의 단위구간으로 이산화 한다. 그 후, 각 단위구간들의 유사도를 수식(2.1)과 같이 계산하여 유사한 단위구간들을 병합한다.



[그림 4] UCPD의 이산화 방법

수식(2.1)에서 A, B는 각각 주성분의 특정 단위구간에 속하는 데이터 d_1, d_2에서 범주형 변수들의 frequent itemset을 의미한다. x는 A와 B에 속하는 item이며, $s_{d_1}(x)$는 d_1에서 item x의 support를 의미한다. α는 scaling parameter이다. 주성분의 단위구간들에 대한 병합이 끝나면, 연속형 변수만 존재하는 경우와 마찬가지로 주성분의 이산화 결과를 원 변수 축에 전사한다.

$$\mathrm{sim}(d_1,d_2) = \frac{\sum_{x \in A \cap B} \max\left\{0, 1-\alpha\left|s_{d_1}(x)-s_{d_2}(x)\right|\right\}}{\| A \cup B \|} \qquad (2.1)$$

그러나 UCPD에는 2가지 문제점이 있다. 첫 번째는, PCA에 기반한 방법이기 때문에 변수 간 선형적인 상관관계만을 반영할 수 있다는 점이다 [18]. 변수 간에 비선형적인 관계가 존재할 경우, UCPD는 이를 포착할 수 없다는 한계가 있다.

두 번째는, 연속형 변수와 범주형 변수를 동시에 처리하지 못한다는 것이다. UCPD는 혼합형 데이터에서 연속형 변수 간의 상관관계를 먼저

고려한 후 범주형 변수를 고려하기 때문에, 연속형 변수와 범주형 변수 간의 상호작용을 보존하지 못할 수 있다.

IPD는 두 단위구간 간의 유사도 측정을 위해, Interaction distance(ID)라는 척도를 정의하여 사용한다. ID는 두 단위구간에 대해 다른 변수들이 가지는 다변수 분포의 누적확률분포 간 거리를 통해 두 구간의 유사도를 측정한다. 해당 척도는 empirical data에 대해 closed form으로 계산 가능하고, 데이터의 분포를 가정할 필요가 없다는 장점이 있다. ID로 측정되는 단위구간 간의 유사도를 바탕으로, IPD는 Minimum description length에 기반 한 목적식을 최적화하는 방식으로 이산화를 수행한다.

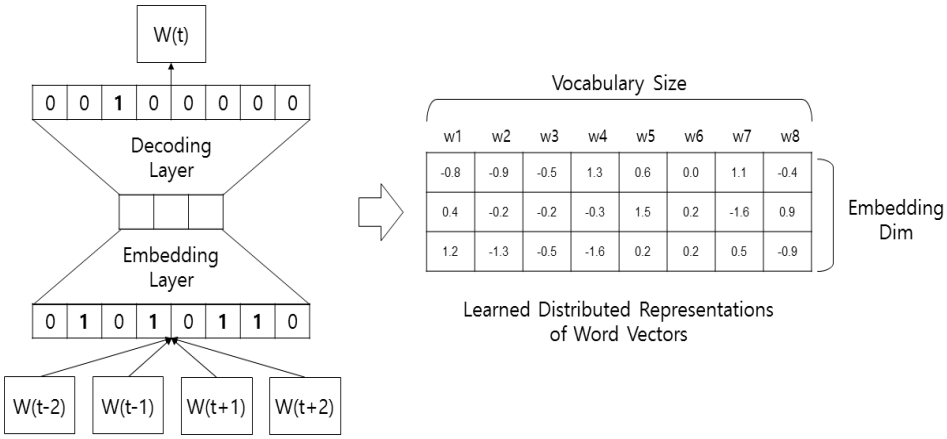
하지만 IPD는 혼합형 데이터에 대해 한계를 가지는데, ID가 범주형 변수를 고려하지 못하기 때문이다. 만약 이산화 하고자 하는 변수와 다른 범주형 변수 간에 상호작용이 존재할 경우, IPD는 이를 반영하지 못한다.

2.3. Word2Vec

제안방법은 구간 간 유사도 계산의 어려움을 임베딩 기법을 통해 해결한다. 임베딩이란 정보를 표현하는 방법에 대해 학습하는 것이다. 제안방법에서는 임베딩을 통해 이산화하고자 하는 변수의 각 단위구간을 하나의 임베딩으로 표현한다. 단위구간에 대한 임베딩의 학습방법은, 딥러닝 기반의 자연어 처리 분야에서 많이 활용되고 있는 Word2Vec[17] 방법과 유사하다.

Word2Vec은 텍스트 데이터를 바탕으로 각 단어의 임베딩을 학습하는 방법으로, 어떤 두 단어가 사용되는 맥락이 비슷하면 그 의미 또한 유사하다는 'distributional hypothesis'[8]에 기반 한다.

Word2Vec 모델은 유사한 맥락에서 사용된 단어들의 임베딩이 유사하도록 학습을 수행한다. 예를 들어, 비슷한 맥락에서 사용되는 두 단어인 'hotel'과 'motel'의 임베딩 간 거리(cosine 거리)는, 그렇지 않은 두 단어인 'hotel'과 'sheep'의 임베딩 간 거리보다 가깝게 학습된다.



[그림 5] CBOW 모델

구체적인 모델 학습 방법은 다음과 같다. 모델은 단어가 글의 맥락(문장, 문서, 또는 특정 window size 크기만큼의 단어 열) 안에서 어떤 단

어들과 함께 등장했는지의 정보를 바탕으로 단어의 임베딩을 학습한다. [그림 5]의 왼쪽은 Word2Vec 학습을 위한 모델 중 하나인 Continous Bag of Words(CBOW) 모델의 구조이다. 모델은 1개의 hidden layer을 가지며 input layer에서 hidden layer로 연결되는 embedding layer, hidden layer에서 output layer로 연결되는 decoding layer를 가진다. hidden layer에서 activation function은 사용하지 않으며, output layer에서는 softmax를 통해 맥락 가운데의 단어를 예측한다.

이를 수식으로 표현하면 다음과 같다. V를 vocabulary의 size, k를 맥락의 window size, d를 단어 임베딩의 차원이라고 하자. 각 단어 $x(\in R^{V\times 1}, x^T 1 = 1)$는 V차원의 one-hot encoding으로 표현된다. t번째 단어를 중심으로 한 맥락 $[x_{t-k}, ..., x_{t-1}, x_t, x_{t+1}, ..., x_{t+k}]$가 있다고 하자. 그러면 모델의 input은 x_t를 제외한, 나머지 $2k$개 주변 단어가 되고, 모델의 target은 x_t가 된다.

모델의 embedding layer를 $E(\in R^{d\times V})$라고 하면, 은닉층 $h(\in R^{d\times 1})$는 수식 (2.2)와 같이 주어진다. Decoding layer를 $D(\in R^{V\times d})$라 하고, j번째 단어에 대응되는 decoding layer의 열벡터를 v_{o_j}라 하자. 그러면 입력 h에 대해 j번째 단어를 맥락의 가운데 단어로 예측할 확률은 수식 (2.3)과 같이 주어진다. 모델은 output layer의 출력 벡터와 target 벡터 간의 cross entropy loss를 통해 학습된다.

$$h = \frac{1}{2k} E(x_{t-k} + x_{t-(k-1)} + ... + x_{t-1} + x_{t+1} + ... + x_{t+(k-1)} + x_{t+k}) \quad (2.2)$$

$$y_j = \frac{\exp(h^T v_{o_j})}{\sum_j \exp(h^T v_{o_j})} \quad (2.3)$$

학습된 CBOW 모델 embedding layer의 각 열벡터는 [그림 5]의 오른쪽과 같이, 그에 대응되는 단어의 임베딩이 된다. 제안방법에서 단위구간의 임베딩을 학습하는 방법은 CBOW 모델이 단어의 임베딩을 학습하는 방식과 유사하다. 이에 대해서는 3장에서 자세히 다룬다.

3. 제안방법

본 연구에서 제안하는 이산화 방법의 핵심은 연속형 변수 단위구간의 임베딩을 학습하고, 임베딩 간의 유사도를 바탕으로 단위구간을 병합하는 것이다. 단위구간의 임베딩을 위한 모델구조와 그 학습 방법은 Word2Vec과 유사하다. Word2Vec에서 단어의 임베딩은 해당 단어 주변에 어떤 단어들이 존재하는지에 대한 정보를 바탕으로 학습된다. 이와 유사하게 연속형 변수 단위구간의 임베딩은 해당 변수가 그 단위구간의 값을 가질 때 다른 변수들이 갖는 값을 바탕으로 학습할 수 있다.

Notations	
m: #Instances	x: training input ($\in \{0,1\}^N$, $x^T 1_N = n-1$)
n: #Variables ($n = n_{CON} + n_{CAT}$)	y: training target ($\in \{0,1\}^N$, $y^T 1_N = 1$)
n_{CON}: #Continuous variables	W_E: Embedding layer weight ($\in R^{d \times N}$)
n_{CAT}: #Categorical variables	W_D: Decoding layer weight ($\in R^{N \times d}$)
$v_{CON}^{(i)}$: i-th continuous variable	b_D: Decoding layer bias ($\in R^N$)
($i = 1, \ldots, n_{CON}$)	$B_k^{(i)}$: k-th Basic interval of $v_{CON}^{(i)}$
$v_{CAT}^{(j)}$: j-th categorical variable	($k = 1, \ldots, K$)
($j = 1, \ldots, n_{CAT}$)	$M^{(i)}$: Pairwise distance matrix($\in R^{K \times K}$)
$\lvert v_{CAT}^{(j)} \rvert$: #Category in $v_{CAT}^{(j)}$	$W_E[B_k^{(i)}]$: Embedding of $B_k^{(i)}$ ($\in R^d$)
N: Dimension of dummy coded data	

[그림 6] 제안방법의 notation

Pseudo Code of Proposed method

Model Input

- D: Original data ($\in R^{m \times n}$)
- K: #Basic intervals
- n_epoch: #Epochs to train model
- d: Embedding dimension

Model Output

- Intervals(Discretization result)

def discretize(D**):**

(Step 1) Discretize continuous variables into basic intervals

for i **in** $[1, \ldots, n_{CON}]$:

Discretize $v_{CON}^{(i)}$ into K basic intervals

Make D^{dummy} ($\in \{0,1\}^{m \times N}$) with D

(Step 2) Learn embeddings of basic intervals

for epoch **in range**(1, n_epoch):

for instance **in** D^{dummy}:

Generate training inputs, targets $(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})$ from instance

Predict y with $\hat{y} = softmax(W_D W_E x + b_D)$

Compute Loss and Update Model Parameters

(Step 3) Merge similar basic intervals

for i **in** $[1, \ldots, n_{CON}]$:

Compute pairwise distance matrix $M^{(i)}$ for $W_E[B_1^{(i)}], \ldots, W_E[B_K^{(i)}]$

Cluster and merge basic intervals for $v_{CON}^{(i)}$

return Intervals

[그림 7] 제안방법의 pseudo code

제안방법에서는 Word2Vec 프레임워크로 단위구간의 임베딩을 학습한다. 정형 데이터의 instance를 하나의 맥락으로, instance의 각 변수를 단어와 같이 다룬다. Word2Vec에서 맥락의 가운데에 있는 단어를 주변의 단어들로 예측하듯이, 제안방법에서는 한 변수의 값(단위구간 또는 범주)을 나머지 변수들의 값으로 예측하는 모델을 통해 단위구간의 임베딩을 학습한다.

제안방법의 pseudo code는 [그림 7]과 같으며, 크게 3단계로 구분할 수 있다. 1단계는 각 연속형 변수를 단위구간들로 이산화 하는 것이다. 2단계는 각 단위구간의 임베딩을 학습하는 것이다. 3단계는 유사한 단위구간들을 병합하는 것이다. 이후의 내용을 통해 각 단계에 대한 세부적인 내용을 설명한다. 제안방법에 대해 설명하는 과정에서 사용하는 notation들은 [그림 6]과 같다.

3.1. 연속형 변수의 단위구간 생성

각 연속형 변수에 대해 EqualFreq 방법을 적용하여 K개의 단위구간으로 이산화 한다. K는 20개에서 30개 정도로 설정한다.[1] 다음으로 모든 변수에 dummy coding을 수행하여, 각 변수를 one-hot encoding으로 표현한다.

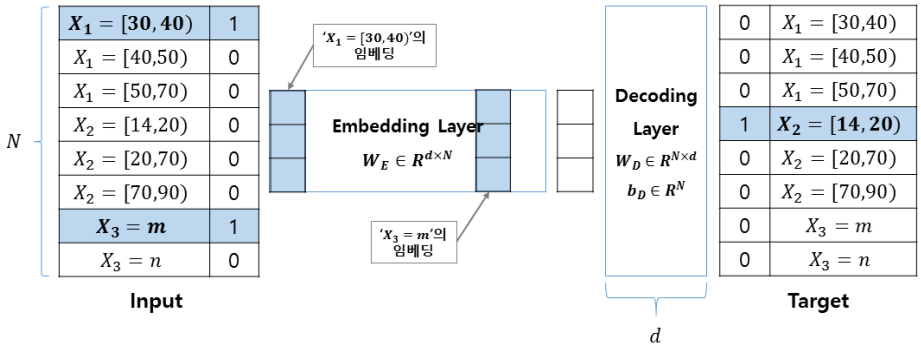
데이터 $D(\in R^{m\times n}, m: Num\,Instances, n: Num\,Variables)$의 i번째 연속형 변수를 $V_{CON}^{(i)}(i=1,...,n_{CON})$, j번째 범주형 변수를 $V_{CAT}^{(j)}$ $(j=1,...,n_{CAT})$, $\left|V_{CAT}^{(j)}\right|$를 j번째 범주형 변수가 갖는 범주의 개수라 하면, 각 변수에 dummy coding을 수행한 데이터 $D^{dummy}(\in\{0,1\}^{m\times N}$, $D^{dummy}1_N = n1_N)$의 차원 N은 $K\times n_{CON}+\sum_{j}^{n_{CAT}}\left|V_{CAT}^{(j)}\right|$으로 주어진다(모든 연속형 변수는 K개의 단위구간으로 이산화 되었다고 가정하자).

일반적으로 변수를 dummy coding 할 시, 다중공선성을 고려하여 변수 별로 1개의 dummy 변수(연속형 변수의 단위구간, 또는 범주형 변수의 범주)를 제거한다. 그러나 본 연구에서는 각 연속형 변수의 단위구간에 대한 임베딩들을 학습하는 것이 목적이므로, 모든 dummy 변수를 사용한다.

1) 변수 중에서 분포가 굉장히 skew되어 있거나 중복된 값이 많은 경우, 단위구간을 K개로 만들 수 없을 수 있다. 이러한 경우에는 가능한 K에 근접한 개수로 변수를 이산화 한다.

3.2. 단위구간의 임베딩 학습

단위구간의 임베딩 모델은 [그림 8]과 같은 형태이다. 모델은 $n-1$개 변수의 값으로 나머지 1개 변수의 값을 예측하는 방식으로 학습된다.[2)]



[그림 8] 단위구간의 임베딩 모델

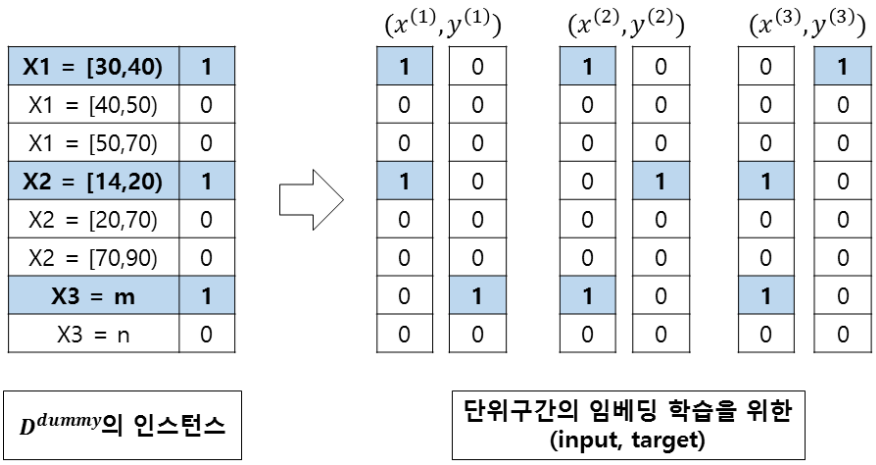
먼저 D^{dummy}의 각 instance에 대해, 총 n개의 모델 input과 target 쌍을 생성한다. 예를 들어 [그림 8]과 같이, 3개의 변수를 갖는 데이터 D로부터 생성된 D^{dummy}의 instance I를 고려하자. I로부터 모델 학습을 위한 input x와 target y의 쌍 3개를 [그림 9]와 같이 생성할 수 있다 ($x \in \{0,1\}^N$, $x^T 1_N = n-1$, $y \in \{0,1\}^N$, $y^T 1 = 1$, $I = x + y$).

모델의 input x에 대한 예측은 $\hat{y} = softmax(W_D W_E x + b_D)$로 주어진다. 생성한 n개의 input에 대해 예측을 수행하고, 예측의 loss를 바탕으로 임베딩 모델의 파라미터를 업데이트한다. Loss함수로는 cross entropy loss를 사용한다.

모델의 파라미터 최적화는 Adagrad[5]를 통해 수행한다. Adagrad는 각 파라미터에 대해 별도의 학습률(learning rate)을 사용하며, 파라미터가 업데이트 되는 빈도가 증가할수록 그 학습률이 작아지는 방향으로 조

2) 실제 구현상에서 모델의 학습은 mini batch단위로 수행되지만, 논의의 단순화를 위해 instance 1개에 대한 학습과정에 대해 설명한다.

정된다. 이는 다음 단계에서 수행되는 단위구간 임베딩 의 클러스터링 결과가, 일정 학습 epoch 이후에서 일관성 있게 생성되도록 하는 효과가 있다.



[그림 9] 단위구간의 임베딩 학습을 위한 input, target 생성

매 파라미터 업데이트 후에는 [24]에서와 같이, 각 단위구간 임베딩의 L2 norm을 1로 scaling한다. 이는 단위구간의 임베딩 간 내적과 cosine 거리를 일치시켜, 모델이 임베딩 간의 cosine 거리를 최적화하도록 만드는 효과가 있다.

3.3. 유사한 단위구간의 병합

학습된 단위구간의 임베딩들을 이용하여, 각 변수 별로 유사한 단위구간들을 병합한다. MVD와 UCPD는 인접구간 간 유사도의 threshold를 파라미터로 지정하여 사용한다. 그러나 제안방법에서 단위구간의 임베딩 간 유사도에 threshold를 지정하는 것은 간단하지 않다. 단위구간의 개수, 단위구간의 임베딩 차원, 데이터의 특성에 따라 적합한 threshold는 매번 달라질 수 있기 때문이다.

제안방법에서는 유사도의 threshold를 직접적으로 지정하는 대신, 단위구간 간의 pairwise distance를 바탕으로 구간들을 클러스터링 한다. 그 후, 클러스터링 결과 하나의 클러스터에 포함된 단위구간들을 하나의 구간으로 병합한다. 각 과정에 대한 구체적인 내용은 아래와 같다.

3.3.1. 단위구간의 임베딩 간 pairwise distance matrix

$V_{CON}^{(i)}$의 단위구간들 간 pairwise distance matrix를 $M^{(i)}(\in R^{K\times K})$라 하자. $M^{(i)}$는 다음과 같은 과정으로 계산된다. 먼저, Embedding layer weight $W_E(\in R^{d\times N})$의 열벡터 중 $B_k^{(i)}$($V_{CON}^{(i)}$의 k번째 단위구간)에 대응되는 벡터(단위구간의 임베딩)를 $W_E\left[B_k^{(i)}\right]$라 하자.

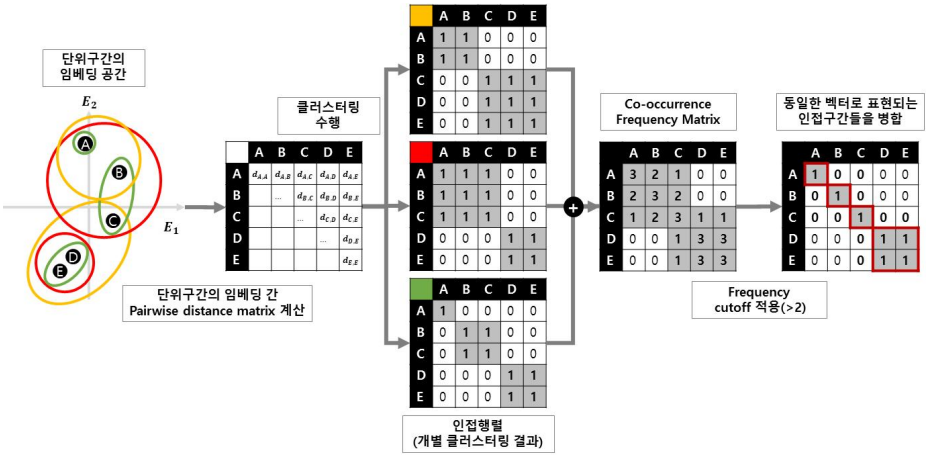
$W_E\left[B_k^{(i)}\right](k=1,...,K)$들에 대해 pairwise distance를 계산하기 전, 클러스터링 품질을 향상시키기 위해 $W_E\left[B_k^{(i)}\right]$들에 대해 Kernel PCA(cosine kernel)를 적용하여 noise를 제거한다. 각 임베딩은 de-noising 과정에서 2차원으로 축소되며, 이를 각각 $w_1^{(i)},...,w_K^{(i)}$라 하자. 그러면 pairwise distance matrix $M^{(i)}$의 각 원소 (a,b)는 수식 (3.2)와 같이 계산된다.

$$M_{a,b}^{(i)}=\cos(w_a^{(i)},w_b^{(i)}),\ where\ a,b\in 1,...,K \tag{3.2}$$

3.3.2. 단위구간의 클러스터링 및 병합

단위구간의 pairwise distance matrix를 바탕으로, 단위구간을 클러스터링 한다. 이를 위해 앙상블 클러스터링을 활용한다. 앙상블 클러스터링을 활용하는 이유는, 각 변수 별로 적절한 구간의 개수가 상이하기 때문이다. 변수의 개수가 많은 경우, 각 변수 별로 클러스터의 개수를 개별적으로 조정하는 것은 쉽지 않다. 앙상블 클러스터링을 활용하면, 각 변수를 얼마나 세밀하게 이산화 할 것인지를 하나의 파라미터를 통해 전체적으로 조절할 수 있다. 또한 앙상블 클러스터링은 클러스터링 결과의 안정성을 높이는데도 도움이 된다.

앙상블 클러스터링의 절차는 다음과 같다. 먼저, K-means 알고리즘의 클러스터 개수를 2개에서 $K-1$개(K: 단위구간의 총 개수)로 변경해가며 단위구간들에 대한 클러스터링 결과를 얻는다. 이 때, 인접하지 않은 구간들이 하나의 클러스터에 속하게 된 경우에는 후처리를 통해 클러스터를 분리한다.



[그림 10] 단위구간의 클러스터링 및 병합

그 다음, [그림 10]과 같이 개별 클러스터링 결과들에 대해, 각 단위구간의 쌍이 같은 클러스터에 속했는지의 정보를 바탕으로 인접행렬을 만

들 수 있다. 이 인접행렬들을 합하여 co-occurrence frequency matrix를 만든다. 그 다음, 이 matrix의 각 원소에 대해 frequency cutoff를 적용하여 binary matrix를 만든다. 이 binary matrix에서 동일한 벡터로 표현되는 단위구간들이 하나의 클러스터가 된다. 각 변수 별로 이와 같은 과정을 수행하여 단위구간들의 클러스터를 얻는다. 마지막으로, 클러스터 별로 그에 속한 단위구간들을 하나의 구간으로 병합함으로써 이산화가 완료된다.

4. 실험

4.1. 실험설계

제안방법은 비지도학습 기반의 다변수적 이산화 방법으로, 이산화 과정에서 변수 간의 상호작용을 보존할 수 있다. 또한 클래스 변수를 필요로 하지 않기 때문에, 비지도학습은 물론 지도학습을 위한 전처리 과정으로도 활용될 수 있다. 제안방법의 성능을 검증하기 위한 실험설계는 다음과 같다.

우선 제안방법이 이산화 과정에서 변수 간의 상호작용을 보존할 수 있으며, 특히 연속형 변수와 범주형 변수 간의 상호작용을 보존할 수 있음을 검증하였다. 이를 위해, 변수 간의 상호작용 패턴을 시각적으로 확인할 수 있는 합성데이터들에 대해 다음의 2가지 실험을 수행하였다. (1)연속형 변수 간에 상호작용이 존재하는 합성데이터에 대해 제안방법이 변수 간의 상호작용을 보존할 수 있는지 검증하였다. 그 다음으로, (2)연속형 변수와 범주형 변수 간에 상호작용이 존재하는 합성데이터에 대한 실험을 통해 연속형 변수와 범주형 변수 간의 상호작용을 보존할 수 있는지 검증하였다.

또한, 제안방법이 지도학습을 위한 전처리로도 활용될 수 있음을 검증하였다. 이를 위해 (3)클래스 레이블이 존재하는 혼합형의 실제데이터들을 제안방법을 통해 전처리 한 후, 분류모델을 학습시키고 그 성능을 평가하는 실험을 수행하였다.

제안방법과의 성능 비교 대상으로 설정한 기존방법은 MVD, UCPD, IPD를 사용하였다. 제안방법에서 단위구간의 임베딩 학습모델은 PyTorch[19]를 통해 구현하였다. 기존 이산화 방법인 MVD와 UCPD는 KEEL[22]의 구현을 활용하였으며, IPD는 해당 논문 저자들의 구현[3]을 활

3) https://github.molgen.mpg.de/EDA/ipd

용하였다.

제안방법의 hyper-parameter는 모든 실험에서 단위구간의 개수는 20개, 임베딩의 차원은 32, 임베딩 학습의 epoch수는 30, 클러스터링 시 co-occurrence frequency의 cutoff는 2로 설정하였다.

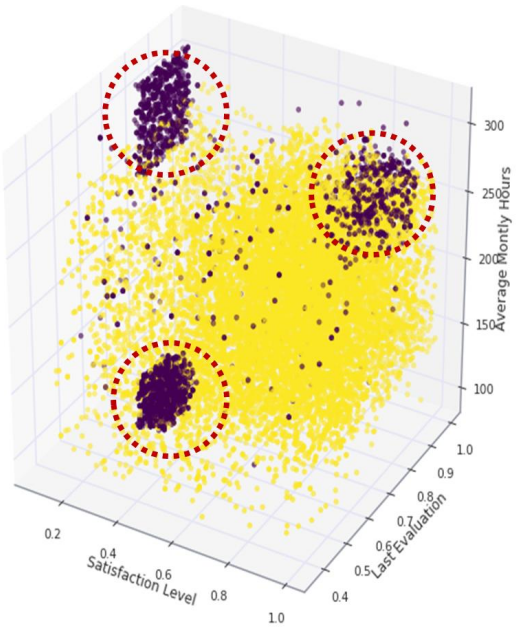
기존방법의 hyper-parameter 설정은, 변수 별로 생성 가능한 최대의 구간 개수를 모든 이산화 방법에서 동일하게 20개로 정한 것 이외에는 기본 설정 값을 사용하였다.

실험에서 사용하는 분류 모델들인 의사결정나무(DT), 로지스틱회귀(LR), 랜덤포레스트(RF)는 Python 기계학습 라이브러리인 Scikit-learn[20]의 구현을 활용하였다. DT의 경우 최대 깊이는 4, LR의 L2 penalty는 1.0, RF의 tree 개수는 10개로 지정하였다.

4.2. 실험데이터

4.2.1. 연속형 변수 간 상호작용이 존재하는 합성데이터

제안방법이 변수 간의 상호작용을 보존할 수 있는지 검증하기 위해, 우선 연속형 변수 간에 상호작용이 존재하는 합성데이터를 고려하자. 실험을 위한 데이터로서 Kaggle에 공개되어 있는 HR 데이터를 사용하였다[4]. HR 데이터는 인공적으로 만들어진 데이터로, 14999개의 instance와 5개의 연속형 변수, 4개의 범주형 변수를 가진다. 클래스 변수는 2개의 클래스를 가지며, 소수 클래스의 비율은 23.8%이다.



[그림 11] HR 데이터의 3차원 산점도

[그림 11]은 HR 데이터의 연속형 변수 중 'last-evaluation', 'satisfaction-level', 'average-montly-hours'들을 각 축으로 하여, 데이터의 분포를 시각화 한 것이다. 진한 색의 점들은 클래스 1, 옅은 색의

4) https://www.kaggle.com/ludobenistant/hr-analytics-1

점들은 클래스 0을 의미한다.

HR 데이터의 3차원 산점도를 살펴보면, 데이터가 밀집된 영역(점선으로 표시된)들을 관찰할 수 있다. 이는 세 변수 간에 상호작용이 존재함을 의미한다. 변수 간의 상호작용을 보존할 수 있는 이산화 방법을 사용하면, 이러한 영역들을 compact하게 cover 할 수 있는 hyper-rectangle을 각 변수 구간의 조합을 통해 만들 수 있을 것이다.

한편 HR 데이터의 경우, 데이터가 밀집된 영역은 주로 클래스 1이 분포하는 영역이다. 따라서 이산화가 적절히 수행된다면, 각 변수 구간의 조합으로 만들 수 있는 hyper-rectangle의 내부는 (1)데이터 포인트의 밀도가 균일하면서, (2)클래스의 불순도가 낮아야 할 것이다.

실험에서는 이산화 결과의 적절성에 대한 판단이 용이하도록, HR 데이터의 전체 입력변수 중에서 앞서 언급한 3개 변수만을 활용하였다. 실험을 통해 제안방법을 포함한 이산화 방법들이 3개 변수 간의 상호작용을 보존할 수 있는지, 각 변수에 대한 이산화 결과를 시각화하여 평가하였다. 또한 각 방법 별로, 이산화 결과로 얻은 데이터에 분류모델을 학습시킨 후 정확도를 비교하였다.

바람직한 이산화 결과는, 이산화 수행 이후에 학습된 분류모델의 성능이 높으면서, 각 변수를 가급적 적은 개수의 구간으로 분할하여야 한다.

4.2.2. 연속형 변수와 범주형 변수 간 상호작용이 존재하는 합성데이터

제안방법이 연속형 변수와 범주형 변수 간의 상호작용을 보존할 수 있음을 검증하기 위해, 다음과 같은 합성데이터를 고려하자. 데이터는 40,000개 instance와 4개의 변수 X_1, ..., X_4를 갖는다. X_1, X_2는 연속형 변수로, 각각 균등분포 $U(0,4)$를 따른다. X_3, X_4는 범주형 변수로, X_1, X_2와 수식(4.1-4)와 같은 관계를 가진다. 이 데이터를 X_1, X_2를 두 축으로 갖는 그래프 상에 표현하면 [그림 12]와 같다. 변수 X_1과 X_2에 대해

이산화를 수행한다고 했을 때, 변수 간의 상호작용 패턴이 동질적인 구간을 형성하기 위해서는 $X_1 = 3$, $X_2 = 3$에서 각 변수의 구간이 분할되어야 할 것이다.

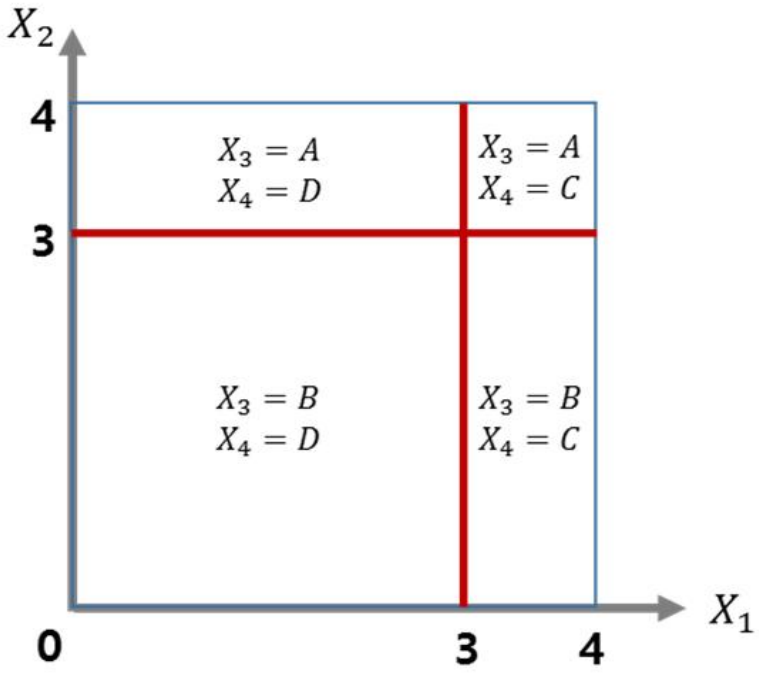
이 합성데이터의 경우, 연속형 변수와 범주형 변수 간의 상호작용을 고려하지 못하면 X_1과 X_2 변수를 제대로 이산화 할 수 없다. 이러한 데이터에 제안방법을 적용하여, 제안방법이 혼합형 데이터에서 변수 간의 상호작용을 보존할 수 있는지 검증하였다,

$$X_3 = A, X_4 = C, \text{ if } X_1 \in [3, 4) \text{ and } X_2 \in [3, 4) \tag{4.1}$$

$$X_3 = A, X_4 = D, \text{ if } X_1 \in [0, 3) \text{ and } X_2 \in [3, 4) \tag{4.2}$$

$$X_3 = B, X_4 = C, \text{ if } X_1 \in [3, 4) \text{ and } X_2 \in [0, 3) \tag{4.3}$$

$$X_3 = B, X_4 = D, \text{ if } X_1 \in [0, 3) \text{ and } X_2 \in [0, 3) \tag{4.4}$$



[그림 12] 연속형 변수와 범주형 변수 간 상호작용이 존재하는 합성데이터

4.2.3. 클래스 레이블이 존재하는 혼합형의 실제데이터

제안하는 이산화 방법이 지도학습을 위한 전처리로도 적합한지 검증하기 위해, UCI repository에 존재하는 데이터 중 클래스 레이블이 존재하는 데이터 4개를 사용하였다. 각 데이터는 혼합형 변수로 구성되어 있으며 각각의 instance 수, 범주형 및 연속형 변수 개수, 클래스 레이블 개수는 [표 1]과 같다. 제안방법을 포함한 각 이산화 방법 별로 [표 1]의 데이터들에 대해 이산화 방법을 적용한 후, 분류모델을 학습시켜 그 성능을 비교하였다.

[표 1] 클래스 레이블이 존재하는 혼합형의 실제데이터

Dataset	#Instance	#Categorical Variable	#Continuous Variable	#Class
Credit1	654	9	6	2
Credit2	1,000	13	7	2
Bank Marketing	41,188	10	10	2
Adult	48,842	8	6	4

4.3. 실험 결과

4.3.1. 연속형 변수 간 상호작용이 존재하는 합성데이터

[그림 13]은 이산화 방법 별 HR 데이터의 이산화 결과로, 위에서부터 MVD, UCPD, IPD 그리고 제안방법을 통해 얻은 결과이다. 각 산점도는 3개의 변수 'last-evaluation', 'satisfaction-level', 'average-montly-hours' 중 2개 변수의 분포를 시각화 한 것이고, 붉은 선은 각 변수에 대한 이산화 결과, 즉 변수 구간의 분할지점을 의미한다.

두 변수 구간의 조합으로 만들어지는 사각형 영역을 하나의 grid라고 하자. 적절한 이산화 결과에서는, 각 grid 안에서의 데이터 패턴은 동질적이어야 하며, 동질적인 패턴을 가진 영역은 하나의 grid에 포함되어야 한다.

MVD의 경우 'satisfaction-level' 변수에 대해, 이질적인 패턴을 가진 구간을 분할하지 못한 것을 알 수 있다. 반면 'last-evaluation'과 'average-montly-hours' 변수에 대한 이산화 한 결과(우측 산점도)를 보면, 동질적인 패턴을 가진 영역(좌측 하단과 우측 상단의 보라색 영역)을 여러 개의 grid로 분할한 것을 확인할 수 있다.

UCPD는 3개 변수 모두 전체 영역의 중간쯤에서 구간을 분할함을 알 수 있다. 이는 UCPD가 PCA에 기반 하기에, 변수 간의 비선형적인 관계를 포착하지 못해 발생하는 문제로 생각된다.

IPD의 경우에는 각 grid가 동질적인 패턴을 보이지만, 동질적인 패턴을 가진 영역이 서로 다른 grid로 분할된 것을 확인할 수 있다. 데이터에 존재하는 패턴에 비해, 변수를 너무 많은 구간으로 이산화 하는 경향이 있음을 알 수 있다. 반면, 제안방법을 통해 형성된 grid들은 동질적인 패턴(각 grid에서 데이터의 밀도가 균일)을 가지면서도, grid의 개수도 적절함을 확인할 수 있다.

한편 HR 데이터의 경우 클래스 1에 해당하는 데이터들이 밀집되어 있

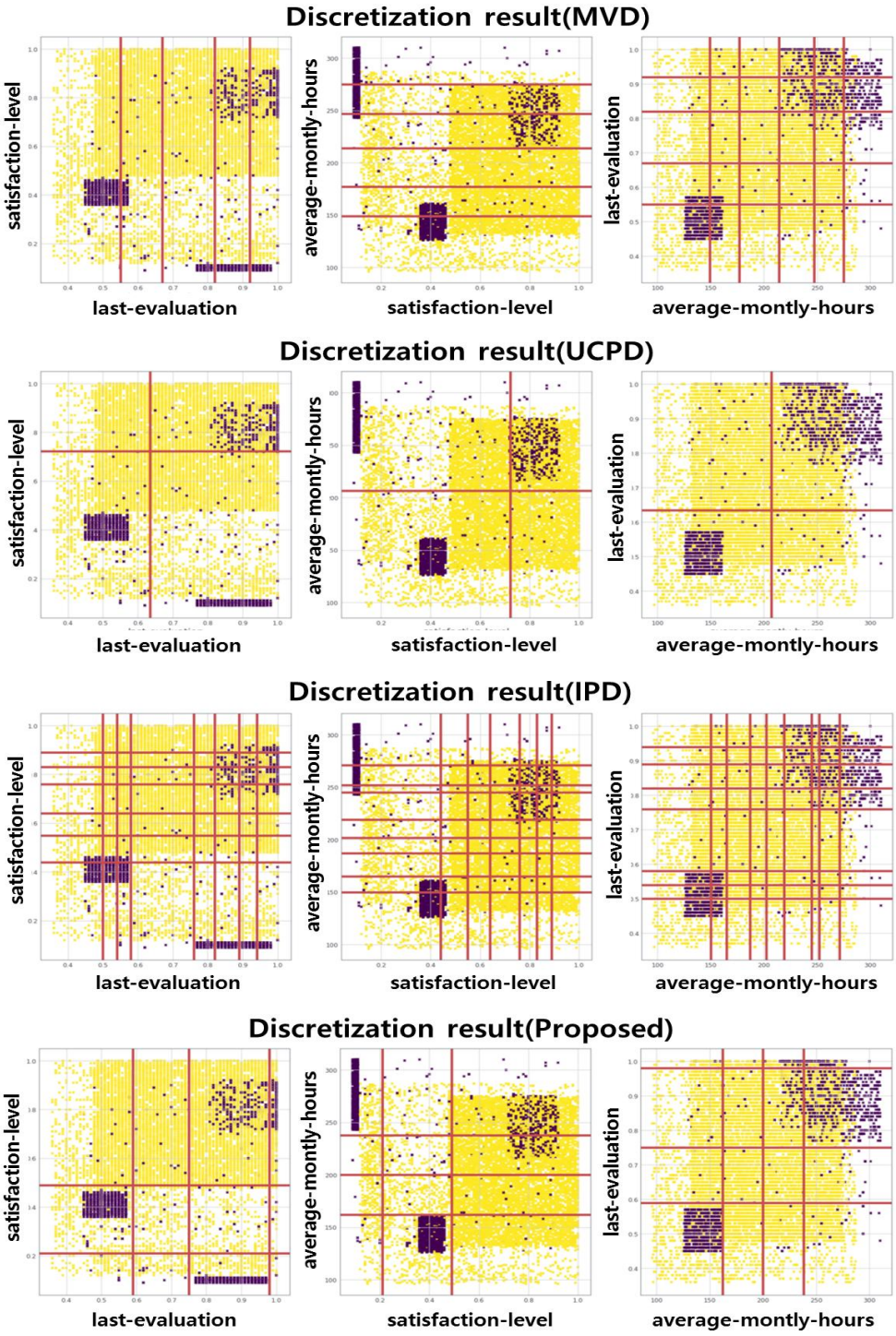
기 때문에, (비지도학습 기반의 이산화 방법일지라도)이산화 품질의 차이는 분류성능의 차이로 이어진다. [표 2]는 이산화 방법과 분류모델의 조합 별로 분류성능을 나타낸 것이다. 분류 성능은 10-fold-CV를 통해 계산되었다.

[표 2] 이산화 방법 별 전처리 이후의 분류 성능(HR 데이터)

Method	DT	LR	RF
MVD	81.4	76.2	81.5
UCPD	88.7(*)	80.4	89.5
IPD	87.1	87.0	90.4
Proposed	88.3	87.3(*)	90.6(*)
Undiscretized	92.0	77.1	96.9

__ : 분류성능이 가장 좋은 경우
(*): 이산화 방법 중에서 분류성능이 가장 좋은 경우

전반적인 성능을 보았을 때, 제안방법이 기존방법들과 비교해 더 좋은 성능을 보임을 확인할 수 있다. 기존방법 중에서는 IPD가 가장 좋은 성능을 보이며, 제안방법과 근소한 성능 차이를 보인다. 그러나 IPD는 [그림 13]에서 확인할 수 있듯이, 과다하게 많은 구간을 생성함을 알 수 있다. 두 방법에서 분류성능이 동일하다면, 데이터를 간결하게 표현할 수 있는 제안방법의 결과가 더 적절할 것이다. 이를 통해 제안방법이 타 방법에 비해 변수 간의 상호작용을 잘 보존하며, 이산화 과정에서 정보손실이 가장 적음을 알 수 있다.



[그림 13] 이산화 방법 별 HR 데이터의 이산화 결과

4.3.2. 연속형 변수와 범주형 변수 간 상호작용이 존재하는 합성데이터

연속형 변수와 범주형 변수 간에 상호작용이 존재하는 합성데이터에 대한 이산화 방법의 성능 평가 실험에서는, 이산화 방법이 각 변수 별로 ground truth의 이산화 결과를 얻을 수 있는지 평가하였다.

이를 위해 제안방법과 UCPD, EqualFreq와 이산화 결과를 비교하였다. 기존의 비지도학습 기반 다변수적 이산화 방법 중에서 연속형 변수와 범주형 변수 간의 상호작용을 고려할 수 있는 방법은 UCPD가 유일하기에, 제안방법과의 비교대상으로 설정하였다.

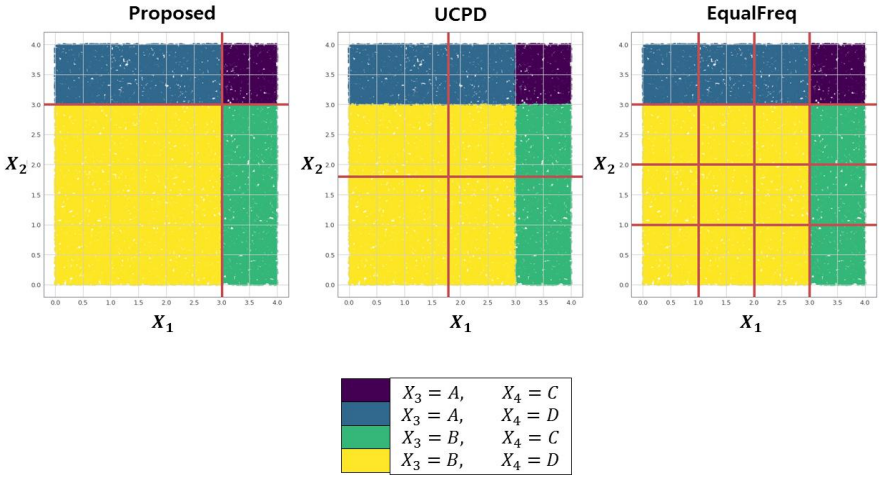
다른 비지도학습 기반 다변수적 이산화 방법인 MVD와 IPD의 경우, 각 변수의 구간을 EqualFreq 방법을 통해 단위구간으로 이산화 한 후에 변수 간의 상호작용을 고려하여 병합해 나간다. 그러나 이 데이터의 경우에는, 연속형 변수만을 고려해서는 단위구간 간의 유사함을 판단하기 어렵다.

이와 같은 이유로 MVD, IPD를 비교대상으로 고려하는 대신, 비지도학습 기반의 단변수적 방법을 사용했을 때 ground truth 분할 지점을 분할하기 위해 필요한 최소의 구간 개수를 참고할 수 있도록 EqualFreq를 비교대상 중 하나로 설정하였다. EqualFreq에서는 변수 별로 ground truth 분할 지점을 분할할 수 있는 최소의 구간 개수인 4개를 사용하였다.

이산화 방법 별로 얻은 각 변수의 분할지점은 [그림 14]와 같다. [그림 14]를 보면, UCPD는 ground truth 분할지점을 선택하지 못했음을 알 수 있다. 오히려 각 변수 영역의 가운데 부근을 분할 지점으로 선택하였다. 이러한 문제의 근본적인 원인은, UCPD가 모든 변수 간의 상호작용을 동시에 고려하지 못하기 때문이다. UCPD는 연속형 변수와 범주형 변수가 모두 존재하는 상황에서, 연속형 변수만을 사용하여 PCA를 수행한 이후에 범주형 변수를 고려한다. 그러나 이 합성데이터의 경우 2개의 연

속형 변수 간에 선형적인 상관관계가 존재하지 않는다. 선형관계가 없는 2개 변수에 대해 주성분을 얻고, 그 주성분 축 상에서 범주형 변수들과의 관계를 고려하는 접근은 잘못된 이산화 결과를 도출하게 할 수 있다.

또한 UCPD는 주성분 축을 단위구간으로 분할한 뒤 수식(2.1)을 활용하여 유사한 단위구간들을 병합하는데, 그 결과물이 유사도 척도의 hyper-parameter(scaling factor, 유사도의 threshold)에 민감하다는 문제가 있다. 이러한 복합적인 문제들로 인해, UCPD에 대해 추가적인 hyper-parameter tuning을 시도하였지만 정확한 이산화 결과를 얻는 데에는 실패하였다([그림 14]의 결과는, 가능한 최대구간 개수를 20으로 조정한 것을 제외하면 KEEL의 기본 설정 값을 바탕으로 얻은 결과이다).

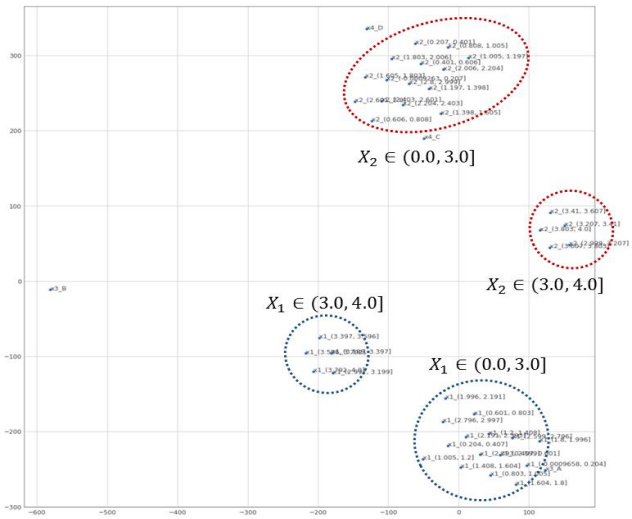


[그림 14] 이산화 방법 별 합성데이터에 대한 이산화 결과

EqualFreq의 경우 ground truth 분할 지점을 분할하기 위해서는 불필요하게 많은 수의 구간을 생성하여야 한다. 반면 제안방법은 ground truth 분할지점인 $X_1 = 3$, $X_2 = 3$에서 각 변수의 구간을 이산화 하고, 그 이외의 구간은 분할하지 않음을 확인할 수 있다.

[그림 15]는 제안방법에서 학습된 단위구간의 임베딩들을 t-SNE[15]를 이용하여 2차원으로 시각화 한 것이다. 시각화 결과를 보면, X_1 변수의

단위구간 20개 중 15개가 $(0.0, 3.0]$ 범위의 클러스터로, 5개가 $(3.0, 4.0]$ 범위의 클러스터로 군집화되는 것을 알 수 있으며, X_2 변수 또한 유사한 양상으로 군집화 되는 것을 확인할 수 있다. 이를 통해 단위구간의 임베딩 간 거리가 변수 간의 상호작용 패턴의 유사함을 반영한다는 것을 알 수 있다. 또한 제안방법이 연속형 변수와 범주형 변수 간의 상호작용을 보존할 수 있음을 확인할 수 있다.



[그림 15] 학습된 단위구간 임베딩의 2차원 시각화 결과

4.3.3. 클래스 레이블이 존재하는 혼합형의 실제데이터

클래스 레이블이 존재하는 혼합형의 실제데이터에 대해, 이산화 방법별로 전처리 이후의 분류 성능을 비교한 것은 [표 3]과 같다. 각 분류모델 별로 가장 성능이 좋은 경우를 밑줄로 표시하였다. 그리고 이산화 방법 중에서(Undiscretized 제외) 가장 성능이 좋았던 경우를 (*)로 표시하였다. [표 4]는 [표 3]의 결과를 종합한 것으로, 이산화 방법 별로 각 데이터, 모델의 조합($4 \times 3 = 12$)에서 해당 방법이 다른 방법과 비교하여 정확도가 가장 높았던 빈도이다.

[표 3] 이산화 방법 별 전처리 이후의 분류 성능(UCI 데이터)

Dataset	Method	DT	LR	RF
Adult	MVD	54.8	55.4	50.8(*)
	UCPD	54.8	55.8	49.7
	IPD	55.5	56.3	49.4
	Undiscretized	56.3	53.8	51.5
	Proposed	56.1(*)	57.1(*)	50.0
Credit1	MVD	81.5	84.9	83.0
	UCPD	81.8	84.4	84.3
	IPD	82.4	84.0	85.2
	Undiscretized	82.3	84.9	85.3
	Proposed	85.7(*)	85.5(*)	85.8(*)
Credit2	MVD	69.9	74.0	73.7(*)
	UCPD	71.9(*)	75.0(*)	72.5
	IPD	71.4	74.7	73.2
	Undiscretized	71.1	74.9	74.1
	Proposed	70.4	74.1	73.6
Bank Marketing	MVD	70.2	78.8	57.3
	UCPD	71.7	80.0	66.6
	IPD	76.6(*)	81.6(*)	66.2
	Undiscretized	76.6	83.8	68.4
	Proposed	75.5	76.8	74.2(*)

__ : 분류성능이 가장 좋은 경우

(*): 이산화 방법 중에서 분류성능이 가장 좋은 경우

[표 4] 이산화 방법 간 성능 비교
(가장 정확도가 높았던 빈도)

Method	이산화 수행하지 않는 경우 포함하여 비교	이산화 방법 간 비교
MVD	0	2
UCPD	2	2
IPD	2	2
Proposed	5	6
Undiscretized	5	-

이산화 방법 간에 정확도를 각 데이터와 모델의 조합 별로 비교했을 때, 제안방법의 정확도가 가장 높았던 빈도가 6번으로 가장 많음을 알 수 있다. IPD, UCPD, MVD는 모두 정확도가 가장 높았던 빈도는 2번씩으로 동일하다.

이산화를 수행하지 않은 경우에서의 정확도를 포함하여 비교한 결과는 [표 4]의 오른쪽 열과 같다. 제안방법은 이산화를 수행하지 않은 경우와 비교하여도 비슷한 수준의 성능을 보임을 확인할 수 있다. 이와 같은 실험결과를 통해, 제안방법이 분류와 같은 지도학습을 위한 전처리로도 유용하게 사용될 수 있음을 확인하였다.

5. 결론

이산화는 기계학습 모델의 예측결과에 대한 이해가 요구되는 도메인에서, 모델의 예측성능 및 예측결과에 대한 해석력과 직결된 중요한 전처리 과정이다.

이산화 과정에서 발생하는 정보손실을 최소화하기 위한 방법으로서, 비지도학습 기반의 다변수적 이산화 방법들이 연구되었다. 이 방법들은 변수 간의 상호작용을 보존하는데 초점을 두고 있다. 그러나 기존방법들은 연속형 변수와 범주형 변수 간의 상호작용을 보존하지 못하는 한계가 있다.

본 연구에서는 임베딩을 활용한 새로운 비지도학습 기반의 다변수적 이산화 방법을 제안하였다. 합성데이터와 실제데이터에 대한 실험을 통해 제안방법이 변수 간의 상호작용을 보존할 수 있으며, 특히 기존방법들과 달리 연속형 변수와 범주형 변수 간의 상호작용 또한 효과적으로 보존할 수 있음을 검증하였다.

제안방법의 한계점은 이산화 결과의 일관성이다. 단위구간의 임베딩은 신경망 모델을 통해 수행된다. 따라서 신경망 모델의 특성 상 weight의 초기 값과 학습률(learning rate)에 따라 다른 이산화 결과를 얻을 수 있다. 이산화 결과의 일관성을 높일 수 있는 방법에 대한 추후 연구가 필요할 것이다. 또한 임베딩의 차원 크기나 단위구간의 개수를 데이터의 instance 개수나 변수 개수에 따라 어떻게 설정해야 하는지, 대략의 가이드라인을 제시하는 것도 유용할 것이다.

제조, 의료, 보험, 마케팅 등 대부분의 도메인에서 다루어지는 데이터는 혼합형 데이터이다. 제안방법은 혼합형 데이터를 다루는데 적합하기에, 다양한 도메인에서의 기계학습에 활용될 수 있을 것이다. 특히, 기계학습 모델의 예측 결과를 바탕으로 의사결정을 내리는 도메인에서, 제안방법은 유용한 전처리 방법으로 사용될 수 있을 것이라 기대한다.

참고문헌

[1] Bay, Stephen D. "Multivariate discretization for set mining." Knowledge and Information Systems 3.4 (2001): 491-512.

[2] Caruana, R., Lou, Y., Gehrke, J., Koch, P., Sturm, M., & Elhadad, N. (2015, August). Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 1721-1730). ACM.

[3] Chmielewski, M. R., & Grzymala-Busse, J. W. (1996). Global discretization of continuous attributes as preprocessing for machine learning. International journal of approximate reasoning, 15(4), 319-331.

[4] Dougherty, J., Kohavi, R., & Sahami, M. (1995, July). Supervised and unsupervised discretization of continuous features. In Machine learning: proceedings of the twelfth international conference (Vol. 12, pp. 194-202).

[5] Duchi, J., Hazan, E., & Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. Journal of Machine Learning Research, 12(Jul), 2121-2159.

[6] Fayyad, U., & Irani, K. (1993). Multi-interval discretization of continuous-valued attributes for classification learning.

[7] Garcia, S., Luengo, J., Sáez, J. A., Lopez, V., & Herrera, F. (2013). A survey of discretization techniques: Taxonomy and empirical analysis in supervised learning. IEEE Transactions on Knowledge and Data Engineering, 25(4), 734-750.

[8] Harris, Z. S. (1954). Distributional structure. Word, 10(2-3), 146-162.

[9] Hourani, A., Chou, C. A., & Khanmohammadi, S. (2014, January). An Improvement on Association Rule Based Classification of Medical Data. In IIE Annual Conference. Proceedings (p. 3371). Institute of Industrial and Systems Engineers (IISE).

[10] Huang, G. M., Huang, K. Y., Lee, T. Y., & Weng, J. T. Y. (2015). An interpretable rule-based diagnostic classification of diabetic nephropathy among type 2 diabetes patients. BMC bioinformatics, 16(1), S5.

[11] Kerber, R. (1992, July). Chimerge: Discretization of numeric attributes. In Proceedings of the tenth national conference on Artificial intelligence (pp. 123-128). Aaai Press.

[12] Lakkaraju, H., Bach, S. H., & Leskovec, J. (2016, August). Interpretable decision sets: A joint framework for description and prediction. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 1675-1684). ACM.

[13] Liu, H., Hussain, F., Tan, C. L., & Dash, M. (2002). Discretization: An enabling technique. Data mining and knowledge discovery, 6(4), 393-423.

[14] Lud, M. C., & Widmer, G. (2000). Relative unsupervised discretization for association rule mining. Principles of data mining and knowledge discovery, 148-158.

[15] Maaten, L. V. D., & Hinton, G. (2008). Visualizing data using t-SNE. Journal of Machine Learning Research, 9(Nov), 2579-2605.

[16] Mehta, S., Parthasarathy, S., & Yang, H. (2005). Toward unsupervised correlation preserving discretization. IEEE Transactions on Knowledge and Data Engineering, 17(9), 1174-1185.

[17] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems (pp. 3111-3119).

[18] Nguyen, H. V., Müller, E., Vreeken, J., & Böhm, K. (2014). Unsupervised interaction-preserving discretization of multivariate data. Data Mining and Knowledge Discovery, 28(5-6), 1366-1397.

[19] Paszke, A., Gross, S., Chintala, S., & Chanan, G. (2017). PyTorch.

[20] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion,

B., Grisel, O., ... & Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12(Oct), 2825-2830.

[21] Ribeiro, M. T., Singh, S., & Guestrin, C. (2016, August). Why should i trust you?: Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 1135-1144). ACM.

[22] Triguero, I., González, S., Moyano, J. M., García, S., Alcalá-Fdez, J., Luengo, J., ... & Herrera, F. (2017). KEEL 3.0: an open source software for multi-stage analysis in data mining. International Journal of Computational Intelligence Systems, 10(1), 1238-1249.

[23] Wong, A. K., & Chiu, D. K. (1987). Synthesizing statistical knowledge from incomplete mixed-mode data. IEEE Transactions on Pattern Analysis and Machine Intelligence, (6).

[24] Xing, C., Wang, D., Liu, C., & Lin, Y. (2015). Normalized Word Embedding and Orthogonal Transform for Bilingual Word Translation. In HLT-NAACL (pp. 1006-1011).

[25] Yang, Y., & Webb, G. I. (2009). Discretization for naive-Bayes learning: managing discretization bias and variance. Machine learning, 74(1), 39-74

Abstract

Multivariate Discretization through Interval-wise Embedding and Clustering of Continuous Variables

Taewook Kim
Department of Industrial Engineering
The Graduate School
Seoul National University

Discretization is a data preprocessing method that converts continuous variables into discrete variables. The purpose of discretization is for interpretable machine learning. In domains such as medical and insurance where machine learning models are used for decision making, interpretability of the model is critical. In these domains, interpretable models such as associative classifier and decision tree are widely used, and they require continuous inputs to be discretized. The result of discretization has a great influence on prediction performance and interpretability of the model. Therefore, sophisticated discretization method is the crucial step for interpretable machine learning. In this study, I proposed an unsupervised multivariate discretization method which is based on embedding technique. Unsupervised multivariate discretization methods do not require class variable and preserve interactions among variables. They can be used as a preprocessing method for unsupervised learning such as association mining or anomaly detection. Also, they can be used for supervised learning such

as classification and prediction. However, existing methods have limitations that they are not able to preserve the interaction between continuous variables and categorical variables. Given that the data from almost any domains, such as manufacturing, healthcare, and insurance, is mixed type, it may be problematic if discretization method is unable to handle mixed data. The proposed method solves this problem through embedding technique. First, each continuous variable is discretized into basic intervals. Then, the embedding of each basic interval is learned by exploiting the interaction among variables. Finally, clustering is performed on the learned embeddings to merge similar basic intervals. Experimental results show that the proposed method can preserve the interactions among variables. Furthermore, it is possible to preserve the interactions between continuous variables and categorical variables, unlike the existing methods. Considering that the data analyzed in almost any domains are mixed type, I expect that the proposed method can be used in various domains where the interpretable model is required.

Keywords: multivariate discretization, embedding, mixed data
Student Number: 2015-22856