퍼지 및 지능시스템학회 논문지 2006, Vol. 16, No. 5, pp. 608-615

클러스터 기반 퍼지 모델트리를 이용한 데이터 모델링

Data Modeling using Cluster Based Fuzzy Model Tree

이대종*, 박진일**, 박상영***, 정남정***, 전명근+***

Dae-Jong Lee, Jin-Il Park, Sang-Young Park, Nahm-Chung Jung, Meung-Geun Chun

* 충북대학교 BK21 충북정보기술사업단

** 충북대학교 전기전자컴퓨터공학부

*** 한국 수자원공사 수자원연구원

요 약

본 논문에서는 퍼지 클러스터 기법을 이용하여 구간 분할된 퍼지 모델트리의 제안과 이를 이용한 데이터 모델링 기법을 다룬다. 제안된 방법은 먼저 입력과 출력변수의 속성을 고려한 퍼지 클러스터링에 의해 중심벡터를 계산한 후, 중심벡터들과 입력속성간의 소속도를 이용하여 구간 분할된 영역별로 각각의 선형모델을 구축한다. 노드의 확장은 부모노드(parent node)에서 만들어진 모델에서 계산된 오차값과 자식노드(child node)에서 계산된 오차값을 비교하여 이루어진다. 출력값 예측 단계에서는 입력된 데이터와 잎노드에서 계산된 클러스터 중심값과 비교하여 소속도가 높은 선형모델을 선택하여 데이터에 대한 출력값을 예측하게 된다. 제안된 방법의 우수성을 보이기 위해 다양한 데이터를 대상으로 실험한 결과, 기존의 모델트리방식 및 뉴럴 네트워크 기반의 신경회로망 보다 향상된 성능을 보임을 알 수 있었다.

키워드 : 모델트리, 퍼지 클러스터, 데이터 예측, M5P

Abstract

This paper proposes a fuzzy model tree consisting of local linear models using fuzzy cluster for data modeling. First, cluster centers are calculated by fuzzy clustering method using all input and output attributes. And then, linear models are constructed at internal nodes with fuzzy membership values between centers and input attributes. The expansion of internal node is determined by comparing errors calculated in parent node with ones in child node, respectively. As a final step, data prediction is performed with a linear model having the highest fuzzy membership value between input attributes and cluster centers in leaf nodes. To show the effectiveness of the proposed method, we have applied our method to various dataset. Under various experiments, our proposed method shows better performance than conventional model tree and artificial neural networks.

Key Words : Model tree, Fuzzy clustering, data prediction, M5P

1. 서 론

"데이터의 홍수, 정보의 빈곤" 시대를 살고 있는 현대인에게 정보의 중요성이 점차 인식되면서 대량의 데이터 속에 내재된 의미 있는 상관관계, 패턴 경향 등을 찾아내는 일련의 프로세스인 데이터 마이닝(Data mining) 기법이 최근 부각되고 있다. 또한, 대량으로 증가하고 있는 데이터 중에서 쓸모 있고 정제된 정보를 발견하는 데이터 마이닝이 중요한 정보추출의 도구로 활용됨으로써 현재 데이터 베이스 분류 관련 분야뿐만 아니라 전 학문 전반에 걸쳐 그 필요성이 점차 증가하고 있다 [1]. 이러한 데이터 마이닝은 방대한 양의 데이터 속에서 통계적 기법, 수학적 기법 및 패턴인식을 찾아내는 일련의 과정으로, 데이터에 근거한 의사 결정 과정이라 할 수 있다. 이러한 데이터 마이닝을 위해 다양한 알고리즘이 제안되고 있다. 주된 알고리즘으로는 학습을 통한 최적화 기법으로 예측, 군집, 분류 등에 이용되고 있는 신경망 기법과 높은 적중률을 보이면서 해석 또한 용이한 결정트리(DT; Decision Tree), 모델트리(MT; Model Tree) 기법 등이 대표적이며, 이 밖에도 결과에 대해 수학적 설명이 가능한 통계적 기법으로 회귀분석, 군집분석, 시계열 분석 등이 있다.

인공 신경망은 인간 두뇌의 신경세포를 모방한 개념으로, 의사 결정트리와 마찬가지로 과거에 수집된 데이터로부터 반복적인 학습과정을 거쳐 데이터에 내재되어 있는 패턴을 찾아내는 모델링 기법이다. 인공 신경망 모델은 통계학적 모형과는 달리 그 자체의 귀납적 특성으로 인해 모형을 도출하기 위한 이론 수립과정을 생략할 수 있으며, 통계학적 모형에서 요구되는 엄격한 가정에 제한을 받지 않는다 [2]. 그러나 인공신경망은 트리구조의 결정법칙에 비해 분석의 결과를 쉽게 이해하기 어렵고 이로부터 유용한 정보를 얻는데 문제점을 지니고 있다. 물론 데이터 마이닝에서 해석의 용이함이 언제나 예측모형의 중요한 특성이 되는 것은 아니다. 더 많은 해석의 용이함을 가지고 있으면서도 예측에 덜 효과적인 모형보다는 매우 정확한 예측을 하는 신경망이 더 선호되는 경우가 많다. 그럼에도 불구하고 변수의 중요성과 그것들 간의 상호작용을 이해하는 것은 향후 데이터 수집 작업을 향상시

+ : 교신저자

접수일자 : 2006년 9월 7일

완료일자 : 2006년 9월 28일

킬 수 도 있기 때문에 신경회로망의 해석적 어려움은 실제적인 단점이 될 수 있다.

결정트리는 의사결정과정을 도표화하여 관심대상 집단을 몇 개의 소집단으로 분류하거나 예측하는 매우 효과적인 데이터 마이닝의 한 기법으로, 모형의 구축과정을 일종의 트리 형태로 표현한다. 결정트리를 형성하는 알고리즘으로 널리 사용되는 것으로 CHAID [3], CART [4], C4.5 [5] 등을 이용하는데 이들은 분리기준과 정지규칙 그리고 가지치기 등에서 서로 다른 형성과정을 가지고 있다. 이러한 결정트리방식은 말단의 잎노드로부터 결과를 도출할 수 있을 뿐만 아니라 연결된 각 내부노드로부터 결과에 대한 과정을 추적할 수 있어 해석이 용이하고 구현이 비교적 간단한 장점을 지니고 있다. 그러나 분석용 자료에만 의존하기 때문에 새로운 자료의 예측에서는 불안정할 가능성이 높고, 이진분류 알고리즘을 적용한 경우 분리 가지수가 많아지는 단점이 있다. 특히, 예측문제에서 목표변수가 연속형인 경우 결정트리에서는 회귀트리(Regreesion tree)를 이용하는데, 이 방법은 목표변수의 평균에 기초해서 분리가 일어나므로 필연적으로 정보의 손실이 발생하여 신경회로망 등에 비해 예측력이 다소 떨어지는 단점을 지니고 있다.

일반적으로 예측문제에서는 연속적인 입력 변수 및 출력값을 갖는 데이터들이 대부분을 차지한다. 결정트리의 하나의 분류인 회귀트리는 말단 노드에 위치한 잎노드에 속한 연속적인 출력값의 평균값을 계산함으로 예측력의 저하를 초래한다. 이러한 문제점을 해결하기 위해 모델트리 기반의 다양한 알고리즘이 제안되고 있으며[6], 주된 기법으로 M5[7], RETIS[8], M5'[9], RegTree[10] 및 HTL[11,12] 등이 있다.

모델트리는 말단의 잎노드에 속한 출력값의 평균값을 계산하는 회귀트리와 달리 연속적인 입력값과 출력값을 이용하여 예측 오차값이 최소화되는 계수값을 계산한 후, 계산된 계수값을 이용하여 출력값을 예측한다. 이러한 모델트리도 회귀트리와 같이 데이터를 반복적으로 분리하여 트리구조를 생성하는 상-하 추론 모델트리(TIMIT:Top-down Induction of Model Tree) 형식을 갖는다. 이러한 트리구조의 추론방식은 몇 가지 문제점 즉, 다중 입력변수 중에서 뿌리노드의 기준이 되는 첫 번째 주요 입력속성을 선정하는 문제, 입노드들의 결정 그리고, 잎노드에서 모델의 선택 등이 선행되어야 한다. 특히, 다중입력 속성 중에서 하나의 선택된 입력속성에 의존하여 트리의 구조를 확장시키는 것은 몇 가지 문제점을 초래할 수 있다. 물론, 단일입력 속성을 선택함으로써 트리구조 자체가 간단하고 명료해 보이지만, 단일 속성만 선택함으로써 다중 속성을 선택한 경우보다 트리 가지의 수가 구조적으로 증가할 우려가 높다. 또한, 어떤 노드에서 주어진 분리기준에 의해 분할되지 않는 입력속성이 존재하는 경우 두 개 또는 그 이상의 입력속성을 동시에 고려하는 것이 분류문제에서 효과적인 것으로 보고되고 있다[13].

본 논문에서는 퍼지 클러스터 기법을 이용하여 단일 입력속성만을 고려하지 않고 모든 입력속성을 고려하여 분리기준을 판정하는 클러스터 기반 퍼지 모델트리(c-fuzzy model tree)를 제안한다. 제안된 방법은 모든 입력속성을 고려하여 퍼지 클러스터에 의해 계산된 중심벡터를 설정한 후, 각각의 중심벡터들과 입력속성간의 소속도를 이용하여 내부 노드를 형성하고, 형성된 내부노드에서 각각의 선형모델을 구축한다. 노드의 분리기준으로서 부모노드(parent node)에서 구축된 모델에서 계산된 에러값이 자식노드(child node)에서 계산된 에러값보다 클 경우에 분기가 이루어진다. 최종 단계에서는 임의의 입력데이터와 잎노드에서 계산된 클러스터 중심값과 비교하여 소속도가 높은 클러스터에 속한 선형모델을 선택하여 출력값을 예측한다. 논문의 구성은 2장에서 제안된 클러스터 기반 퍼지 모델트리에 대하여 설명한다. 3장에서는 제안한 알고리즘과 관련한 실험 및 고찰을 설명하고, 마지막으로 4장에서 결론을 맺는다.

2. 클러스터 기반 퍼지 모델트리

2.1 모델트리

모델트리는 회귀트리와 구조적으로 동일하지만 회귀트리는 말단의 잎노드에 속한 연속적인 출력값의 평균값을 계산함으로써 예측력의 저하를 초래한다. 이러한 문제점을 해결하기 위해 모델트리 기반의 다양한 알고리즘이 제안되고 있으며, 주된 차이점으로는 각각의 잎노드에 속한 평균값을 취하는 회귀트리와 달리 연속적인 입력과 출력값을 대상으로 예측값과 실제 출력값과의 에러가 최소화되는 선형모델을 생성하고, 생성된 선형모델을 이용하여 출력값을 예측한다 [7].

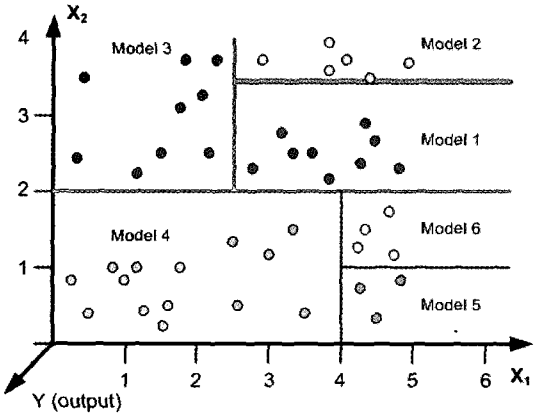
모델트리를 추론하기 위해 M5 알고리즘이 이용된다. M5 모델트리 알고리즘은 식 (1)에서 보인 바와 같이 해당되는 내부마디에 존재하는 입출력 데이터의 표준 편차와 상위노드와 하위노드와의 감소율에 기인한 SDR(Standard Deviation Reduction)을 분리기준으로 사용하고 있다.

$$SDR = sd(T) - \sum_i \frac{|T_i|}{|T|} \times sd(T_i) \qquad (1)$$

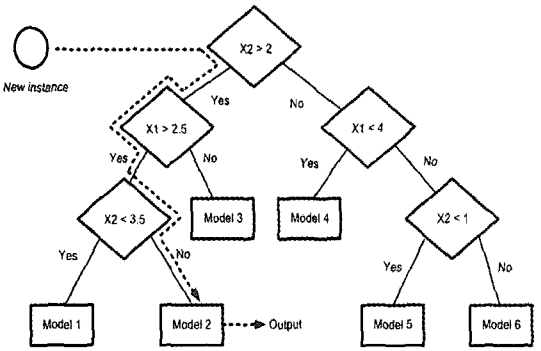
여기에서 T는 도달한 마디의 예제들의 집합이고, $T_1, T_2 \cdots$ 들은 선택된 속성에 따라 분리된 마디로 부터의 결과 집합들이다.

모델트리는 입력속성들 중에서 하나의 속성만을 대상으로 SDR을 계산한 후, SDR이 최대가 되는 수치값을 기준점으로 하여 입력공간을 분할한다. 동일한 방법으로 설정된 조건을 만족할 때까지 노드의 분기는 지속된다. 그러나 지나치게 많은 마디를 가지는 트리구조는 새로운 자료를 적용할 때 예측 오차가 매우 커지는 경향이 있다. 따라서 모델트리구조가 형성된 후, 주어진 트리구조에서 적절하지 않은 마디를 제거하여 적당한 크기의 구조를 갖도록 가지치기(Prunning) 과정을 수행하게 된다. 마지막 단계에서는 가지치기 과정을 수행한 각각의 말단에 위치한 잎노드에서 계산된 선형 모델들 사이에서 필연적으로 발생할 수 있는 비연속적인 값들을 보상해주는 평활화(smoothing)과정이 수행된다. 이런 모든 과정을 거친 후, 임의의 입력데이터에 대하여 루트노드로부터 말단의 잎노드까지 경로를 탐색한 후, 잎노드에서 계산된 선형계수값을 이용하여 출력값을 예측한다.

그림 1에서는 모델트리의 생성 및 추론과정을 나타냈다. 구조를 나타냈다. 그림 1(a) 에서 보는 바와 같이 분리기준에 의해 2차원의 입력공간이 6개의 모델을 갖는 부분공간으로 분할되었다. 여기서, 각각의 분할모델은 선형회귀모델($y = a_0 + a_1 x_1 + a_2 x_2$)을 갖는다. 그림 1(b)에서는 새로운 입력속성에 대한 추론과정을 나타냈다. 새로운 입력속성은 뿌리노드의 조건 ($X_2 > 2$)으로부터 시작하여 내부노드를 거쳐 말단의 잎노드인 Model 2를 탐색하였다. 그림 1(b)에서 점선이 새로운 속성이 뿌리노드에서 입노드까지의 추적경로(path)를 나타낸다. 최종적으로 입력속성에 대한 출력값은 Model 2에서 미리 계산된 선형계수값을 이용하여 예측한다.



(a) 모델트리의 생성



(b) 모델트리의 추론

그림 1. 모델트리의 생성 및 추론 과정
Fig. 1. Building and inducing process of model trees
(a) Building model trees (b) Inducing model trees

2.2 클러스터 기반 퍼지 모델트리

2.2.1. 퍼지 클러스터링

기존의 모델트리방식에서는 다중 입력변수들 중 첫 번째로 중요한 특성을 갖는 변수를 선정한 후 분리기준인 SDR 값을 이용하여 입력공간의 분할이 이루어진다. 상위 루트에서 결정된 분기점을 대상으로 또 다른 입력속성을 이용하여 다시 분기점을 계산하고 입력공간의 분할을 나타내는 내부노드를 구축한다. 이러한 과정은 정지규칙을 만족할 때 까지 반복적으로 수행되며, 최종적으로 정지규칙에 만족하는 말단의 잎노드를 얻으며, 이를 이용하여 트리구조의 모델을 형성한다. 본 논문에서는 퍼지 클러스터 기법을 이용하여 단일 입력속성만을 고려하지 않고 모든 입력속성을 동시에 고려하여 모델을 형성하는 퍼지 클러스터 기반 모델트리를 제안한다. 제안된 방법은 n개의 입력데이터를 대해 퍼지 c-Means 클러스터링 기법을 이용하여 $c(<n)$ 개의 중심벡터를 계산한다. 여기서, 중심벡터는 한 개의 입력속성만이 아닌 모든 입력속성을 갖는다. 계산된 중심벡터들과 입력데이터간의 퍼지 소속도를 계산하고 소속도가 높은 데이터로 입력데이터의 분할이 이루어지는 내부노드를 형성하고 각각의 내부노드에서 선형모델을 생성한다. 노드의 분리기준으로서 부모노드에서 구축된 모델에서 계산된 오차값이 자식노드들에서 계산된 오차값보다 클 경우 분할이 이루어진다.

본 논문에서 데이터 분할을 위해 사용된 퍼지 c-Means 알고리즘을 단계별로 간략히 설명하면 다음과 같다 [14].

[단계 1] 취득하고자 하는 대표점의 수 즉, 클러스터의 수 $c(2 \le c \le n)$를 정하고, 초기 분할행렬 $U^{(0)}$를 초기화한다. 분할행렬에 할당될 소속정도의 값 μ_{ik}는 다음 식을 만족한다.

$$\mu_{ik} = \mu_{Ai}(x_k) \in [0,1] \tag{2}$$

[단계 2] 각 단계에서 데이터 x와 초기 분할행렬값 u를 이용하여 대표 특징의 중심 $v_{ij^{(r)}}$을 계산한다.

$$v_{ij}^{(r)} = \frac{\sum_{k=1}^{n} \mu_{ik}^{m} \cdot x_{kj}}{\sum_{k=1}^{n} \mu_{ik}^{m}} \tag{3}$$

[단계 3] 단계 2에서 계산된 대표 특징들의 중심값과 데이터 x와의 거리값 d에 의하여 분할 행렬 $U^{(r)}$을 다음과 같이 갱신한다.

$$\mu_{ik}^{(r+1)} = \left[\sum_{j=1}^{c} \left(\frac{d_{ik}^{(r)}}{d_{jk}^{(r)}} \right)^{2/(m-1)} \right]^{-1} \tag{4}$$

여기서, m은 퍼지화 정도를 나타내는 퍼지 수로써 일반적으로 2를 이용한다. 또한, d_{jk}는 p차원을 갖는 j번째 데이터 x_j와 k번째 대표 중심값 v_k와의 유클리디안 거리값을 의미한다.

$$d_{jk} = d(x_j, c_k) = \left[\sum_{i=1}^{p} (x_{ji} - v_{ki})^2 \right]^{-1} \tag{5}$$

[단계 4] 다음과 같이 목적함수를 계산한 후, 만약 $\| J(U^r, v^r) - J(U^{(r-1)}, v^{(r-1)}) \| \le \epsilon_l$ 이면 알고리즘을 종료하고 그렇지 않으면 [단계 2]로 가서 반복 수행한다.

$$J(U, c_1, c_2, \cdots, c_c) = \sum_{i=1}^{c} J_i = \sum_{i=1}^{c} \sum_{j=1}^{n} u_{ij}^{m} d_{ij}^{2} \tag{6}$$

2.2.2. 클러스터기반 퍼지 모델트리

앞서 설명된 FCM 알고리즘은 n개의 입력 패턴 $X=[x(1), x(2), \cdots, x(n)] \in R^{q \times n}$을 c개의 중심벡터인 $\mathbf{V}=[\mathrm{v}(1), \mathrm{v}(2), \cdots, \mathrm{v}(\mathrm{c})] \in \mathrm{R}^{q \times c}$ 클러스터로 군집화 시켜주는 기법이다. k번째 입력패턴 $x(k)$는$[x_1(k) x_2(k), \cdots, \cdots, x_q(k)]^T \in R^q$이고, 출력은 $y(k) \in R^1$인 경우 입출력 데이터 z_k와 모든 데이터를 나타내는 Z는 다음과 같이 표현된다.

$$z(k) = [x_1(k) x_2(k), \cdots, x_q(k), y_k]^T \in R^{(q+1)} \tag{7}$$

$$Z = [z(1), z(2), \cdots, z(n)] \in R^{(q+1) \times n} \tag{8}$$

일반적으로 FCM 알고리즘은 입력변수 X 만을 고려하지만, 제안된 모델트리 알고리즘은 출력값을 포함하여 데이터의 특성이 반영되도록 입력과 출력을 포함한 Z를 이용하여 중심벡터를 구하였다 [13]. 따라서 FCM에 의해 입력패턴에 대하여 계산된 i번째 중심벡터 $v(i) = [v_1(i), v_2(i), \cdots, v_q(i)]$와 출력패턴에 대하여 계산된 i번째 중심벡터 $w_i = v_{(q+1)}(i)$를 얻을 수 있다. 중심벡터를 구한 후 하위 노드로 분기할 것인지의 판정은 다음 네 가지의 조건을 고려한다.

표 1. 분기조건
Table 1. Split criterion

- 분기 전 예측 오차값이 설정된 값 (S_1)이상일 때
- 분기 후 모든 클러스터에 포함되는 데이터의 개수가 설정된 값 (S_2)이상일 때
- 분기 전과 분기 후의 오차값 향상이 설정된 값 (S_3)이상 일 때
- 분기된 트리의 깊이 (depth)가 설정된 값 (S_4)이하 일 때

앞서 기술된 표기 방법에 따라 클러스터 기반 퍼지 모델 트리를 이용하여 데이터 모델을 구하는 과정을 단계별로 설명하면 다음과 같다.

[단계 1] 표 1에 언급된 분기조건에 적용되는 값 S_1, S_2, S_3, S_4 을 설정한다.

[단계 2] 모델트리의 특정 노드에 존재하는 $h\,(h \geq S_2)$ 개의 입출력 데이터 $\{X, Y\} \in R^{q \times h}$에 대하여 최소자승(LSE:Least Square Error)법을 이용하여 선형계수값을 구한 후, 실제 출력값과 예측값과의 오차값을 다음과 같이 산출한다. 식 (10)으로부터 구한 오차값 E_b 값이 S_1 이상일 때 다음 단계를 실행하고 그렇지 않을 경우 분기를 정지한다.

$$\hat{y}(k) = a_1 \cdot x_1(k) + a_2 x_2(k) + \cdots + a_q x_q(k) + a_{q+1} \quad (9)$$
$$\text{for} \quad k = 1,2,\cdots,h$$

$$E_b = \sqrt{\sum_{k=1}^{h} (\hat{y}(k) - y(k))^2 / h} \quad (10)$$

[단계 3] FCM 알고리즘을 이용하여 [단계 1]의 노드에 존재하는 입출력 데이터를 이용하여 c 개의 클러스터 중심값을 산출한 후, 다음과 같이 입력값을 c 개의 중심값 중에서 소속도가 높은 클러스터로 하위노드 X_i의 입출력 클러스터를 형성한다.

$$\begin{cases} X_i = \{x(k) \mid u_i(x(k)) > u_j(x(k))\}, \ all \ i \neq j \\ Y_i = \{y(k) \mid (x(k)) \in X_i\} \end{cases} \quad (11)$$

여기서, U_i는 아래와 같이 상위노드에 있는 데이터와 i번째 하위노드의 중심벡터에 대해 식 (4)로부터 계산되는 소속값을 나타낸다.

$$U_i = [\, u_i(x(1)), u_i(x(2)), \cdots, u_i(x(h)) \,] \quad (12)$$

[단계 4] 각각의 하위노드인 $X_1, X_2, \cdots, X_c$에 존재하는 데이터의 개수 $n_1, n_2, \cdots, n_c$를 계산한 후, 각각의 데이터의 개수 중 하나라도 설정된 개수(S_2) 이하이면 분기를 정지하고 상위노드를 말단의 잎노드(leaf node)로 간주한다. 그렇지 않을 경우 [단계 5]를 실행한다.

[단계 5] 하위노드 중 클러스터 i 에 해당하는 입출력 데이터 $\{X_i, Y_i\}$만을 이용하여 [단계 1]에서 계산된 방법과 마찬가지로 실제 출력값과 예측값과의 오차값을 각각 산출한 후, 하위노드에 존재하는 모든 데이터를 이용하여 에러값 E_f 를 구한다.

$$\hat{y}_i(k) = b_{i1} \cdot x_{i1}(k) + b_{i2} x_{i2}(k) + \cdots + b_{iq} x_{iq}(k) + b_{iq_{+1}} \quad (13)$$
$$\text{for } k = 1,2,\cdots,n_i$$

$$E_f = \sqrt{\left(\sum_{i=1}^{c} \sum_{j=1}^{n_i} (\hat{y}_i(j) - y_i(j))^2\right) / \left\{\sum_{i=1}^{c} n_i\right\}} \quad (14)$$

여기서, E_f은 모든 클러스터에 해당되는 데이터들을 이용하여 예측된 출력값과 실제 출력값과의 오차값을 나타낸다.

분기전 상위노드에서 식 (10)에서 계산된 오차값 E_b 와 분기 후 모든 하위노드에서 계산된 에러값 E_c간의 차 $\delta = E_b - E_f$를 계산한 후 δ값이 증가하거나 아주 적은 값을 갖는 임계값 (S_3) 이하의 값을 가질 경우 분기과정을 정지한다. 즉, δ가 증가한다는 의미는 분기를 하였음에도 불구하고 오차값이 증가함을 의미하고 또한 δ가 임계값 이하로 감소하지 않는다는 의미는 분기를 했음에도 오차 측면에서는 큰 효과가 없음을 의미한다.

[단계 6] 표 1의 분기조건을 만족하는 하위노드를 대상으로 분기를 시작하며, 그 과정은 **[단계 1]~[단계 5]** 과정을 반복한다. 단, 트리의 깊이(depth)가 설정된 값 (S_4)를 초과할 경우 분기는 정지한다.

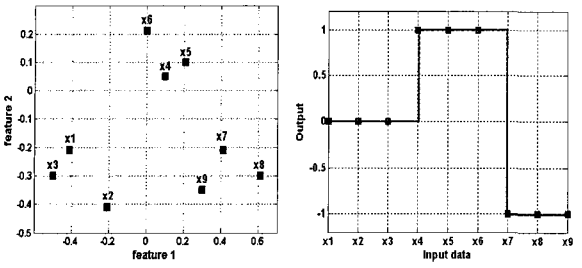
3. 실험 및 결과

3.1 예제를 통한 제안방법의 타당성 검토

(1) 2입력-1출력을 갖는 데이터에 적용

제안된 퍼지 클러스터 기반 모델 트리 알고리즘을 설명하기 위해 그림 2와 같이 2입력-1출력을 갖는 9개의 데이터를 고려해 볼 수 있다.

각각의 계산과정별로 살펴보면 다음과 같다. 우선, 모든 입출력 데이터를 이용하여 루트노드에서 선형 계수값을 산출한 결과, $\hat{y}_{root}(k) = -1.063\, x_1(k) + 3.2166 x_2(k) + 0.576$의 관계식을 얻었으며, 이 값을 이용하여 에러값 $E_{root} = 0.324$를 얻었다. 다음 단계로, 클러스터링 기반 모델트리를 형성하기 위해 분할하고자 하는 클러스터의 수를 2개로 설정한 후, FCM에 의해 클러스터의 중심벡터 $v_{11} = [-0.059, -0.021]$과 $v_{12} = [0.362, -0.289]$를 계산하였다. 계산된 클러스터와 입력데이터간의 퍼지 소속도를 식 (4)에 의해 계산한 결과 표 2와 같다. 표 2에서 소속도가 높은 클러스터로 데이터를 분할 하여 내부노드를 형성하였고, 해당 클러스터를 대상으로 선형 계수값을 구하였다. 계산된 계수값을 이용하여 에러값을 구한 결과 클러스터 1에 대한 오차값 $E_{11} = 0.0758$, $E_{12} = 0.0$를 얻었으며, 분기된 모든 내부노드에 있는 데이터의 오차값을 구한 결과 $E_1 = 0.0619$을 얻었다. 따라서 분기 전의 오차값 E_{root}와 분기 후 에러값 E_1 의 차 $\delta = 0.0139$를 얻어 데이터를 클러스터별로 분할함으로써 오차율이 감소됨을 확인 할 수 있다.



(a) 입력 (b) 출력

그림 2. 2입력-1출력 데이터
Fig. 2 dimensional input and 1 dimensional output data, (a) input (b) output

표 2. 분기 후에 계산된 퍼지 소속도
Table 2. Fuzzy membership after partitioning

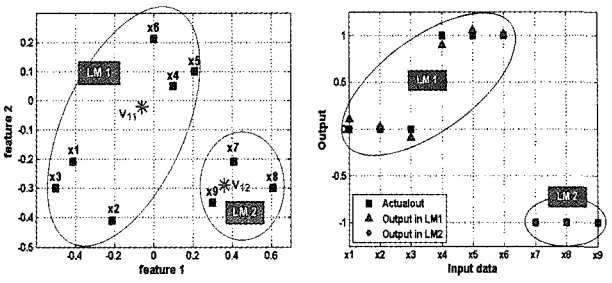
	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
v_{11}	0.7909	0.6624	0.7315	0.8342	0.6671	0.8695	0.0324	0.1055	0.0309
v_{12}	0.2091	0.3376	0.2685	0.1418	0.1418	0.3329	0.9676	0.8945	0.9691
E_{1j}	E_{11} = 0.0758						E_{12} = 0		
E_1	E_1 = 0.0619								

표 2에서 클러스터 2의 모델(LM2)에서 계산된 $E_{12}=0$이므로 더 이상 분할을 할 필요가 없으나, 클러스터 1의 내부노드에서 계산된 에러값 $E_{11}=0.0758$의 값을 가지므로 클러스터 1에 해당하는 입출력 데이터를 이용하여 동일한 방식에 의해 모델 분할을 시도하였으며 그 결과를 표 3에 나타냈다. 전 과정과 동일한 방법으로 소속도가 높은 클러스터로 데이터를 분할 한 후 해당 클러스터를 대상으로 선형계수값을 구하고, 계산된 계수값을 이용하여 에러값을 구한 결과 클러스터 1에서 분기된 LM21 모델에서 오차값 $E_{21}=0.0$과 LM22 모델에서 $E_{22}=0.0$을 얻었다. 또한, 다음번 깊이의 모든 내부노드에서 계산된 오차값 $E_2=0.0$ 으로 계산되었다. 따라서 첫번째 깊이의 첫번째 내부노드에서 산출된 에러값 E_{11}과 분할 후 에러값 E_2의 차 $\delta=0.0758$를 얻어 분기함으로써 에러율이 감소됨을 확인 할 수 있었다.

표 3. 두 번째 분기 후 잎노드에서 계산된 퍼지 소속도
Table 3. Fuzzy membership calculated from leaf nodes after second partitioning

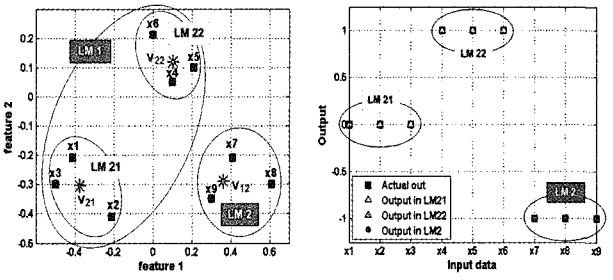
	x_1	x_2	x_3	x_4	x_5	x_6
v_{21}	0.9865	0.9774	0.9554	0.0269	0.0918	0.0281
v_{22}	0.0135	0.0226	0.0446	0.9731	0.9082	0.9719
E_{2j}	E_{21} = 0			E_{22} = 0		
E_2	0					

그림 3 및 4에서는 분할 깊이에 따른 입력공간의 분할과 출력을 나타냈다. 최종적으로 그림 4에서 보는 바와 같이 분할된 클러스터는 출력의 특성에 맞게 세부분으로 구축되었으며, 각각의 영역별로 선형계수값을 구한 후, 이 값만을 이용함으로써 분할하지 않고 사용하는 일반적인 방법에 비해 성능이 우수함을 확인할 수 있었다. 그림 5에서는 퍼지 클러스터에 의해 구축된 모델트리의 구조를 나타냈다.



(a) 입력공간의 분할 (b) 출력

그림 3. 첫 번째 분기 후 입력과 출력
Fig. 3. Input and output at the first partitioning (a) Partitioning of input space (b) output



(a) 입력공간의 분할 (b) 출력

그림 4. 잎노드에서의 입력과 출력
Fig. 4. Input and output at leaf nodes (a) Split of input space (b) output

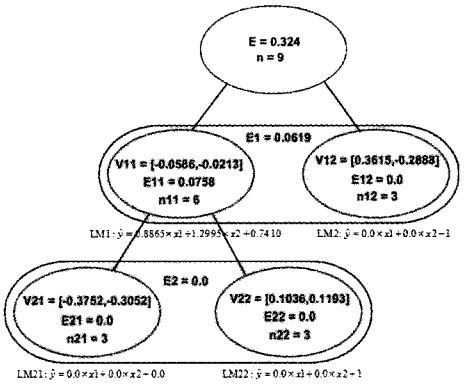
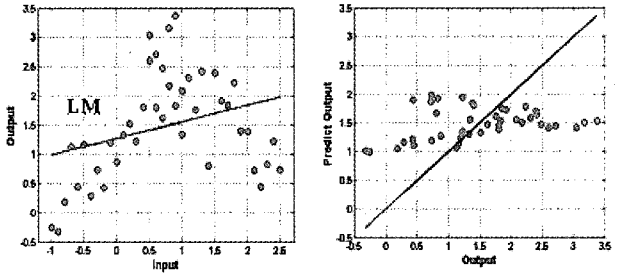


그림 5. c-퍼지 모델트리의 구조
Fig. 5. Structure of c-fuzzy model tree

(2) 단일입력-단일출력을 갖는 데이터에 적용

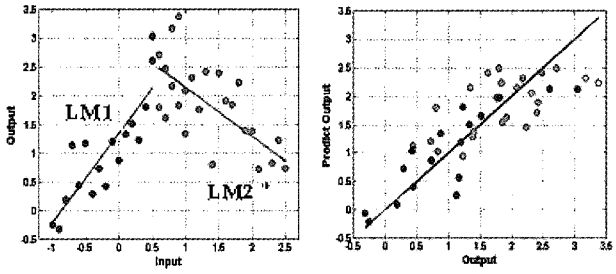
그림 6에서 보인 단일입력-단일출력 데이터를 이용하여 제안된 방법을 적용하였다. 그림 6~8에서는 분할 전, 첫 번째 분할 후, 두 번째 분할 후 계산된 선형모델과 실제값과 예측값을 비교하여 나타냈다. 그림

에서 알 수 있는 바와 같이 분할 노드의 개수가 증가할수록 예측값과 실제 출력값과의 오차율이 감소됨을 확인 할 수 있었다. 그림 9에서는 제안된 방법에 의하여 구축된 모델트리의 구조를 나타냈다. 그림 9에서 보는 바와 같이 분할 전의 데이터를 이용할 경우 에러값이 0.8468이었으나, 클러스터 기반에 의해 하위노드로 분기하였을 경우 에러율이 0.5098로 감소됨을 확인할 수 있다. 하위노드 중 에러율이 0.5452인 LM2에 대해서 다시 한번 분할을 하였을 경우 에러율이 0.5275로 감소하였다.



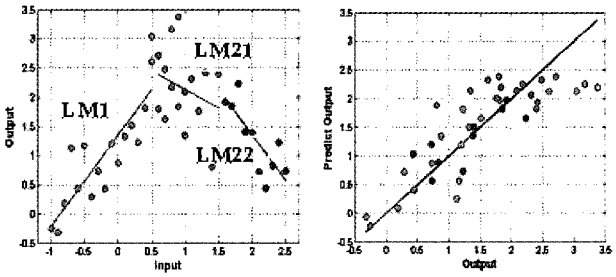
(a) 입출력 (b) 실제값과 예측값

그림 6. 뿌리노드에서의 선형모델
Fig. 6. Linear mode in root node



(a) 입출력 (b) 실제값과 예측값

그림 7. 첫 번째 분할 후 선형모델
Fig. 7. Linear mode after the first expansion



(a) 입출력 (b) 실제값과 예측값

그림 8. 두 번째 분할 후 선형모델
Fig. 8. Linear mode after the second expansion

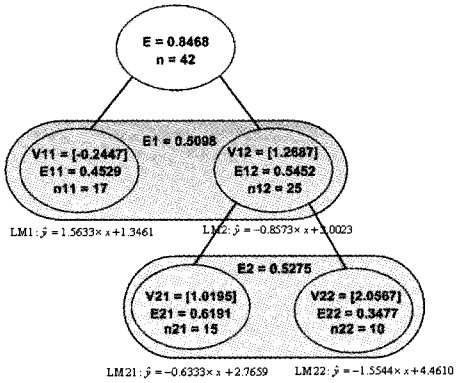


그림 9. c-퍼지 모델트리 구조
Fig. 9. Structure of c-fuzzy model tree

3.2 벤치마크 데이터를 이용한 실험

제안된 퍼지 클러스터 기반 모델트리 알고리즘을 평가하기 위하여 표 4에서 보인 데이터를 이용하였다[15]. 성능 비교를 위해 데이터 마이닝에서 널리 사용되는 역전파(BP: Back-Propagagtion) 알고리즘과 모델트리에서 가장 진보된 알고리즘인 M5P과 비교하였다. 다양한 BP 알고리즘 중 MATLAB에 구현된 Lenvenberg-Marquardt 알고리즘 적용하였고 [16], M5P는 데이터 마이닝에 널리 쓰이는 WEKA 프로그램에서 실행하였다[17].

입력과 출력데이터를 각각 [0, 1]로 정규화 한 후 실험한 결과 표 5와 같은 결과를 얻었다. 제안된 방법의 경우 루트노드에서의 에러값과 분할 후 잎 노드에서의 에러값을 별도로 기입하였다. 표에서 보는 바와 같이 모든 데이터에 대하여 퍼지 클러스터기법을 이용하여 분기를 함으로써 에러값의 감소를 나타냈다. 실험결과, 표 5에서 보는 바와 같이 모든 데이터에 대하여 인공신경망에 의한 회귀방법보다 트리구조의 M5P과 제안된 방법이 우수한 것으로 나타났다. M5P과 제안된 방법을 비교하면, "Servo" 데이터만을 제외하고 모든 데이터에서 제안된 클러스터 기반 퍼지 모델트리 방식이 우수한 결과를 보였다. 표 6에서는 여러 가지 비교척도를 이용하여 두 방법을 비교분석하였다. 분석 결과 기존의 모델트리 방식에 비해 제안된 클러스터기반 퍼지 모델트리 방식이 우수한 결과를 보임을 확인할 수 있었다.

4. 결 론

본 논문에서는 퍼지 클러스터에 의한 산출된 중심값과 퍼지소속 정보를 기준으로 모델을 분기한 후, 데이터를 모델링하는 새로운 퍼지 클러스터 기반 모델트리 방법을 기술하였다. 제안된 방법은 하나의 입력속성만을 선택하여 분기하는 기존의 모델트리 방식과 달리 모든 입력속성을 고려한 분기조건을 제안함으로써 예측 오차율을 줄 일 수 있었다. 최종 단계에서는 임의의 입력데이터에 대하여 루트노드로부터 하위노드 까지 각각의 클러스터 중심값 간에 계산된 소속도를 이용하여 분기점을 추론하면서 각각의 말단에 위치한 잎노드를 탐색한 후 미리 계산된 선형계수값을 이용하여 출력값을 예측한다. 다양한 데이터를 대상으로 실험한 결과 기존의

표 4. 벤치마크 데이터의 사양
Table 4. Specification of benchmark dataset

Dataset	# Observations		# Attributes		Output properties			
	Training	Test	Continuous	Nominal	Max	Min	Mean	Std
Machine CPU	100	109	6	0	1150	6	105.6	160.8
Abalone	2000	2177	7	1	29	1	9.933	3.224
Delta ailerons	3000	4129	6	0	0.002	-0.002	-7e-6	3.0e-4
Delta elevator	4000	5517	6	0	0.013	-0.014	-1e-4	0.002
Computer activity	4000	4192	8	0	99	0	83.96	18.40
Servo	80	87	0	4	7.101	0.131	1.389	1.559

표 5. 벤치마크 데이터에 대한 실험결과
Table 5. Experimental results for benchmark dataset

Data sets	BP		M5' (MinNumber: 4)		Fuzzy cluster based Model Tree			
					Root node		Leaf node	
	Training	Testing	Training	Testing	Training	Testing	Training	Testing
Machine CPU	0.0030	0.0919	0.0245	0.0441	0.0340	0.0546	0.0152	0.0333
Abalone	0.0715	0.0779	0.0774	0.0764	0.0811	0.0781	0.0769	0.0762
Delta ailerons	0.0340	0.0411	0.0367	0.0402	0.0386	0.0411	0.0374	0.0399
Delta elevators	0.0506	0.0545	0.0536	0.0528	0.0544	0.0531	0.0537	0.0528
Computer activity	0.1565	0.1742	0.1728	0.1566	0.0434	0.0445	0.0410	0.0428
Servo	0.0031	0.2100	0.0945	0.1053	0.1545	0.1659	0.0882	0.1095

표 6. M5'와 제안방법의 성능비교
Table 6. Comparing our method with M5'

Data sets	Correlation coefficient		Mean absolute error		Root mean squared error		Relative absolute error (%)		Root relative squared error (%)	
	M5'	our method	M5'	our method	M5'	our method	M5'	our method	M5'	our method
Machine CPU	0.9371	0.9647	0.0187	0.0106	0.0441	0.0334	26.843	14.757	35.578	26.944
Abalone	0.7331	0.7349	0.0548	0.0549	0.0764	0.0762	66.319	66.520	68.246	68.111
Delta ailerons	0.8360	0.8388	0.0282	0.0282	0.0402	0.0399	49.270	49.293	55.089	54.758
Delta elevators	0.7990	0.7985	0.0401	0.0401	0.0528	0.0528	54.650	54.640	60.182	60.257
Computer activity	0.4529	0.9500	0.0715	0.0265	0.1566	0.0428	68.950	25.337	89.531	31.231
Servo	0.8858	0.8900	0.0630	0.0624	0.1053	0.1095	40.154	37.334	48.094	50.265

모델트리방식, 뉴럴 네트워크 기반의 인식방법 보다 향상된 인식 성능을 보임을 알 수 있었다. 특히, 동일한 구조를 갖는 기존의 모델트리 방식과 제안된 방법을 비교한 결과 단순한 예측 오차율뿐만 아니라 모든 비교척도에서 제안된 클러스터기반 퍼지 모델트리 방식이 우수한 결과를 보임을 확인할 수 있었다.

참 고 문 헌

[1] Jiawei Han, Micheline Kamber, "Data Mining : Concepts and Techniques", MORGAN KAUFMANN, pp550, 2001.

[2] Richard O. Duda, Peter E. Hart, David G. Stork, Pattern Classification, Willy Interscience, 2000

[3] Kass, Chi-squard Automatic Interaction Detection, Magidson and SPSS inc., 1980

[4] Breiman, L., Friedman, J.H., Olshen, R.A. and Stone, C.J., "Classification and regression tree", Belmont CA:Wadsworth, 1984

[5] Quilan, J.R., "C4.5: Programs for machine learning", Morgan Kaufmann, 1993.

[6] Donato Malerbe, and et al, "Stepwise Indcution of Model Trees, LNAI 1275, pp.20-32m 2001,

[7] Quinlan J.R. "Learning with continuous classes" in Proceedings AI'92, Adams & Sterling (Eds.), World Sc -ientific, pp. 343-348, 1992.

[8] Karalic A, "Linear regression in regression tree leaves, in Proceedings of ISSEK'92, Bled,

Solvenia, 1992.
[9] Wang Y., Witten I.H., "Inducing Model Trees for Continuous Classes", in Poster Paper of the 9th European Conference on Machine Learning (ECML 97)., M. van Someren, & G. Widmer (Eds.), Prague, Czech Republic, pp. 128-137, 1997.
[10] Lanubile A., Malerba D, "Induction of regression trees with Regtree", in Book of Short Paper on Classification and Data Analysis", Pescara, Italy, pp.253-260, 1997.
[11] Torgo L, "Kernel Regression Trees", in Poster paper of 9th European Conference on Machine Learning (ECML 97), M. van Someren, & G. Widmer (Eds.), Prague, Czech Republic, pp. 118-127, 1997.
[12] Torgo L, "Functional Models for Regression Tree Leaves", in Proceedings of the Fourteenth International Conference (ICML '97), D. Fisher (Ed.), Nashville, Tennessee, pp.385-393, 1997.
[13] Witold Pedrycz, "C-Fuzzy Decision Trees", IEEE Trans. on System, Man, and Cybernetics, Part C, Vol. 35, No. 4, pp. 498-511, 2005.
[14] J.C. Bezdk, Pattern Recognition with Fuzzy Objective Algorithms, Plenum Press, New York, 1981.
[15] http://www.sgi.com/tech/mlc/db
[16] Guang-Bin Huang and et al, Extreme learning machine:Theory and applications", Avaiable online at www.sciencedirect.com, 2006
[17] http://www.cs.waikato.ac.nz/~ml/weka/index.html

저 자 소 개

이대종(Dae Jong Lee)
1995년 : 충북대학교 전기공학과(학사)
1997년 : 충북대학교 전기공학과(공학석사)
2002년 : 충북대학교 전기공학과 (공학박사)
2004년~2005년 : University of Alberta Postdoc.
2006년~현재 : 충북대학교 BK21 충북정보기술사업단 초빙 조교수
관심분야 : 음성신호처리, 얼굴인식, 다중생체인식
Phone : 043) 261-2388
Fax : 043) 268-2386
E-mail : djmidori@empal.com

박진일(Jin Il Park)
2001년 : 한밭대학교 제어계측공학과(학사)
2003년 : 한밭대학교 제어계측공학과 (공학석사)
2005년~현재 : 충북대 제어계측공학과 박사과정.
관심분야 : 지능시스템, 다중생체인식, 퍼지이론
Phone : 043) 261-2388
Fax : 043) 268-2386
E-mail : moralskr@yahoo.co.kr

박상영(Sang Young Park)
1996년 : 충북대학교 도시공학과(학사)
1999년 : 충북대학교 도시공학과 (공학석사)
2004년 : 충북대학교 도시공학과 (공학박사)
2005년~현재 : 한국수자원공사 수자원연구원 선임연구원
관심분야 : 데이터마이닝, 시계열분석
E-mail : sypark119@kwater.or.kr

정남정(Nahm Chung Jung)
1981년 : 경북대학교 (학사)
1998년 : UNESCO-IHE 위생공학(공학석사)
2003년~현재 : UNESCO-IHE, TU Delft (박사과정)
1985년~현재 : 수자원공사 입사, 상하수도 연구소장
관심분야 : 저수지 수질모델링(물리적 수치모델과 데이터 마이닝)
E-mail : chung@kwater.or.kr

전명근(Myung Geun Chun)
1987년 : 부산대학교 전자공학과(학사)
1989년 : 한국과학기술원 전기 및 전자공학과(공학석사)
1993년 : 한국과학기술원 전기 및 전자공학과(공학박사)
1993년~1996년 : 삼성전자 자동화연구소 선임연구원
2000년~2001년 : University of Alberta 방문교수
1996년~현재 : 충북대학교 전기전자 컴퓨터공학부 교수
관심분야 : Biometrics, 감정인식, 지능시스템
Phone : 043) 261-2388
Fax : 043) 268-2386
E-mail : mgchun@chungbuk.ac.kr