한국지능시스템학회 논문지 2008, Vol. 18, No. 1, pp. 41-47

상호 노드 정보를 이용한 클러스터 기반 퍼지 모델트리

Cluster Based Fuzzy Model Tree Using Node Information

박진일* · 이대종** · 김용삼* · 조영임*** · 전명근+*

Jin-Il Park*, Dae-Jong Lee**, Yong-Sam Kim*, Young-Im Cho*** and Myung-Geun Chun+*

* 충북대학교 전기전자컴퓨터공학부

** 충북대학교 BK21 충북정보기술사업단

*** 수원대학교 IT 대학 컴퓨터학과

요 약

클러스터 기반 퍼지 모델트리에서 훈련 데이터의 과잉 적응은 검증데이터의 성능을 저하시키는 문제점을 가지고 있다. 이러한 문제점을 해결하기 위한 방법으로 본 논문에서는 상호 노드간의 정보를 고려하는 방법을 제안하고자 한다. 제안된 방법은 우선 입력과 출력변수의 속성을 고려한 퍼지 클러스터링에 의해 중심벡터를 계산한 후, 중심벡터들과 입력 속성간의 소속도를 이용하여 구간 분할된 영역별로 각각의 선형모델을 구축한다. 예측 단계에서는 입력된 데이터가 잎노드에 도달하기까지 경유하게 되는 노드들의 중심벡터들과 입력 데이터간의 거리값에 따른 소속도를 계산한 후 최종적으로 각 노드의 선형모델들과 계산된 소속도를 이용하여 출력값을 예측하게 된다. 제안된 방법의 우수성을 보이기 위해 다양한 벤치마크 데이터를 대상을 실험한 결과, 기존의 클러스터 기반 퍼지 모델트리보다 향상된 성능을 보임을 알 수 있었다.

키워드 : 데이터 모델링, 데이터 예측, 퍼지 클러스터링, 퍼지 모델트리

Abstract

Cluster based fuzzy model tree has certain drawbacks to decrease performance of testing data when over-fitting of training data exists. To reduce the sensitivity of performance due to over-fitting problem, we proposed a modified cluster based fuzzy model tree with node information. To construct model tree, cluster centers are calculated by fuzzy clustering method using all input and output attributes in advance. And then, linear models are constructed at internal nodes with fuzzy membership values between centers and input attributes. In the prediction step, membership values are calculated by using fuzzy distance between input attributes and all centers that passing the nodes from root to leaf nodes. Finally, data prediction is performed by the weighted average method with the linear models and fuzzy membership values. 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 cluster based fuzzy model tree.

Key Words : Data Modeling, Data Prediction, Fuzzy Clustering, Fuzzy Model Tree

1. 서 론

결정트리는 의사결정과정을 도표화하여 관심대상 집단을 몇 개의 소집단으로 분류하거나 예측하는 매우 효과적인 데이터 마이닝의 한 기법으로, 모형의 구축과정을 일종의 트리 형태로 표현한다. 결정트리의 분류인 모델트리 기반의 다양한 알고리즘이 제안되고 있으며[1], 주된 기법으로 M5[2], RETIS[3], M5'[4], RegTree[5] 및 HTL[6] 등이 있다. 일반적으로 예측문제에서는 연속적인 입력 변수 및 출력값을 갖는 데이터들이 대부분을 차지한다. 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.

클러스터 기반 퍼지 모델트리(C-FMT:Cluster-based Fuzzy Model Tree)는 다중 입력변수들 중 중요한 특성을 갖는 변수를 선정한 후 분리기준인 SDR(Standard Deviation Reduction)값을 이용하여 입력공간을 분할하는 모델트리방식과 다르게 모든 입력속성들을 고려하여 분리기준을 판정하는 방법이다. 이 방법은 모든 입력속성을 고려하여 퍼지 클러스터(FCM:Fuzzy C-Means)에 의해 계산된 중

접수일자 : 2007년 10월 25일
완료일자 : 2008년 1월 15일
감사의 글 : 본 연구는 정보통신부 및 정보통신연구진흥원의 IT신성장동력핵심기술개발사업의 일환으로 수행하였음. [2007-S-020-01, 프라이버시 보호형 바이오인식 시스템 개발]
+ : 교신저자

심벡터를 설정한 후, 각각의 중심벡터들과 입력속성간의 소속도를 이용하여 내부 노드를 형성하고, 형성된 내부노드에서 각각의 선형모델을 구축한다. 노드의 분리기준으로서 부모노드(parent node)에서 구축된 모델에서 계산된 오차값이 자식노드(child node)에서 계산된 오차값보다 클 경우에 분기가 이루어진다. 최종 단계에서는 임의의 입력데이터와 잎노드에서 계산된 클러스터 중심값과 비교하여 소속도가 높은 클러스터에 속한 선형모델을 선택하여 출력값을 예측한다[7,8].

이러한 트리 기반의 알고리즘에서 모델트리의 생성과정에서 훈련 데이터에 과잉 적응하게 되는 경우 이 트리는 잡음들에 의한 좋은 일반화 성능을 보일 것으로 기대될 수 없다. 역으로, 트리의 분할이 너무 일찍 중단되면 훈련 데이터에 대한 오차가 충분히 낮지 않으며 따라서 전체적인 성능의 저하를 초래할 수 있다. 이러한 트리 분할 정지 시점의 문제점을 해결하기 위하여 데이터의 부분 집합을 사용하여 훈련하는 교차확인이나, 최선의 후보 분할이 이루어지도록 엔트로피를 고려하는 방법 등이 연구되어지고 있다. 본 논문에서는 클러스터 기반 퍼지 모델트리에서 이러한 문제점를 해결하기 위한 방법을 제안하고자 한다.

클러스터 기반 퍼지 모델트리는 최종 예측 단계에서 잎노드의 선형모델만을 고려하는 지역적 정보에 국한되는 단점을 가지고 있다. 제안된 방법에서는 입력된 데이터가 모델트리의 상위노드에서 잎노드에 도달하는 노드간의 상호 정보를 이용함으로써 상위 노드의 전역적인 특성과 하위 노드의 지역적 특성을 모두 고려하는 방법을 제안하고자 한다. 제안된 방법에서는 예측 단계에서 입력된 데이터가 잎노드에 도달하기까지 경유하게 되는 노드들의 중심벡터들과 입력 데이터간의 거리값에 따른 소속도를 계산한 후 최종적으로 각 노드들의 선형모델들과 계산된 소속도를 이용하여 출력값을 예측하게 된다. 본 논문의 구성은 2장에서 제안된 상호 노드 정보를 이용한 클러스터 기반 퍼지 모델트리 기법에 대하여 설명한다. 3장에서는 제안한 알고리즘과 관련한 실험 및 고찰을 설명하고, 마지막으로 4장에서 결론을 맺는다.

2. 상호 노드 정보를 이용한 클러스터 기반 퍼지 모델트리

2.1. 클러스터 기반 퍼지 모델트리

클러스터 기반 퍼지 모델트리는 단일 입력속성만을 고려하지 않고 모든 입력속성을 동시에 고려하여 모델을 생성하는 퍼지 클러스터 기반 모델트리 기법으로 n 개의 입력데이터에 대해 FCM를 이용하여 $c(<n)$ 개의 중심벡터를 계산한다. 일반적으로 FCM알고리즘은 입력변수만을 고려하지만, 클러스터 기반 퍼지 모델트리 알고리즘은 출력값을 포함하여 데이터의 특성이 반영되도록 입력과 출력을 포함한 중심벡터를 구한다[8]. 따라서 FCM에 의해 입력패턴에 대하여 계산된 q차원을 갖는 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.

클러스터 기반 퍼지 모델트리를 이용하여 데이터 모델을 구하는 과정을 단계별로 설명하면 다음과 같다.

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

[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 (2)로부터 구한 오차값 E_b 값이 S_1 이상일 때 다음 단계를 실행하고, 그렇지 않을 경우 분기를 정지한다.

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

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

[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 소속도 $u_i(k)$ 가 높은 클러스터로 하위노드 X_i 의 입출력 클러스터를 형성한다.

$$\begin{cases} X_i = \{x(k) \mid u_i(x(k)) > u_j(x(k))\}, \ all \ i \neq j \\ Y_i = \{y(k) \mid (x(k)) \in X_i\} \end{cases} \tag{3}$$

[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 개수중 하나라도 설정된 개수 (S_2) 이하이면 분기를 정지하고 상위노드를 말단의 잎노드로 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 오차값 E_f 를 구한다.

$$\hat{y_i}(k) = a_1^i \cdot x_1^i(k) + ,\cdots, + a_q^i \cdot x_q^i(k) + a_{q+1}^i \quad \text{for } k = 1,2,\cdots,n_i \tag{4}$$

$$E_f = \sqrt{\left\{\sum_{i=1}^{c}\sum_{j=1}^{n_i} (\hat{y_i}(j) - y_i(j))^2\right\} / \left\{\sum_{i=1}^{c} n_i\right\}} \tag{5}$$

여기서, E_f 은 모든 클러스터에 해당되는 데이터들을 이용하여 예측된 출력값과 실제 출력값과의 오차값을 나타낸다. 분기전 상위노드에서 식 (2)를 이용하여 계산된 오차값 E_b 와 분기 후 모든 하위노드에서 계산된 오차값 E_c 간의 차

$\delta = E_b - E_f$ 를 계산한 후, δ 값이 증가하거나 아주 적은 값을 갖는 임계값 (S_3) 이하의 값을 가질 경우 분기과정을 정지한다. 즉, δ 가 증가한다는 의미는 분기를 하였음에도 불구하고 오차값이 증가함을 의미하고 또한 δ 가 임계값 이하로 감소하지 않는다는 의미는 분기를 했음에도 오차 측면에서는 큰 효과가 없음을 의미한다.

[단계 6] 표 1의 분기조건을 만족하는 하위 노드를 대상으로 분기를 시작하며, 그 과정은 [단계 1]~[단계 5] 과정을 반복한다. 단, 트리의 깊이가 설정된 값 (S_4) 를 초과할 경우 분기는 정지한다. 그림 1에서는 모델트리의 생성 및 추론과정을 나타냈다. 그림 1(a)에서 보는 바와 같이 분리기준에 의해 2차원의 입력공간이 부분공간으로 분할되었다. 여기서, 각각의 분할 모델은 선형회귀모델 ($y = a_1x_1 + a_2x_2 + a_3$) 을 갖는다. 그림 1(b)에서는 새로운 입력속성에 대한 추론과정을 나타냈다. 새로운 입력속성은 뿌리노드에서 시작하여 노드의 중심값에 따른 소속도에 기반하여 말단의 잎노드까지 탐색을 한다. 그림 1(b)에서 점선이 새로운 속성이 뿌리노드에서 잎노드까지의 추적경로(path)를 나타낸다. 최종적으로 입력속성에 대한 출력값은 잎노드에서 n_{22} 개의 훈련데이터에 의해 미리 계산된 선형계수값을 이용하여 예측한다.

2.2 상호 노드 정보를 이용한 클러스터 기반 퍼지 모델트리

클러스터 기반 퍼지 모델트리에서는 새로운 입력속성에 대하여 훈련 데이터에 의해 생성된 모델트리의 말단 잎노드의 선형모델만을 고려한다. 따라서, 잎노드에 속하는 지역적 데이터의 특성에만 의존하게 되어 잡음 등에 의한 좋지 않은 성능을 얻을 수 있으며, 또한 훈련 데이터의 과잉 적응으로 인하여 전체적인 성능의 저하를 초래 할 수 있다.

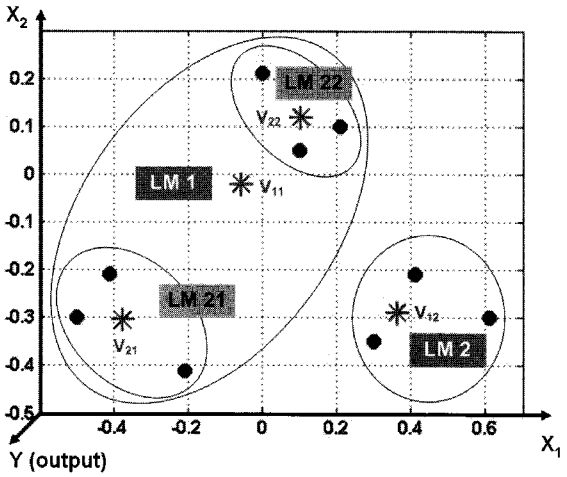
제안된 알고리즘에서는 모델트리 구축에서 FCM 알고리즘에 의한 노드의 중심벡터인 $V = [v(1), v(2), \cdots, v(c)] \in R^{q \times c}$ 을 고려한다. 기존의 방법에서는 c 개의 클러스터 중심값 중에서 소속도가 높은 클러스터의 하위노드에서의 선형모델만을 이용하였다. 그림 2에서는 제안된 방법의 추론 과정을 나타냈다. 그림 1(b)에서는 말단 잎노드의 선형모델만을 고려하는 반면 제안된 방법에서는 그림 2에서 보는 바와 같이 상위 노드의 선형모델까지 고려하여 최종 예측을 하는 구조로 되어있다.

새로운 입력 데이터 $X_i = \left[x_1^i, x_2^i, \cdots, x_q^i\right]$ 가 훈련 데이터에 의해 구축된 모델트리의 루트노드에서 잎노드까지 경유하는 p 개의 중심벡터가 $V_j \in R^{1 \times q}$, $j = 1, 2, \cdots, p$ 이고, 각각의 노드에서의 선형모델을 다음과 같이 정의할 때

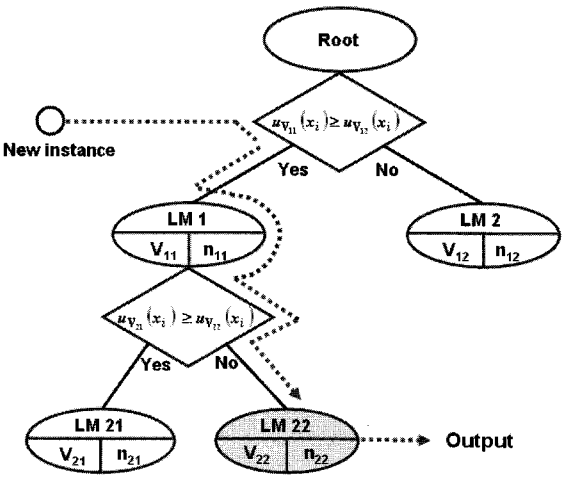
$$\begin{aligned} \hat{y}_i(1) &= a_1^1 \cdot x_1^i + , \cdots , + a_q^1 \cdot x_q^i + a_{q+1}^1 \\ \hat{y}_i(2) &= a_1^2 \cdot x_1^i + , \cdots , + a_q^2 \cdot x_q^i + a_{q+1}^2 \\ &\vdots \qquad\quad \vdots \qquad\quad \vdots \\ \hat{y}_i(p) &= a_1^p \cdot x_1^i + , \cdots , + a_q^p \cdot x_q^i + a_{q+1}^p \end{aligned} \tag{6}$$

중심벡터 V_j 와 입력 데이터 X_i 와의 거리값 d_{ij} 에 따른 소속도를 아래의 식 (7)을 이용하여 계산한다.

$$\mu_{ij} = \frac{1}{\sum_{k=1}^{p} \left(\frac{d_{ij}}{d_{ik}} \right)^{2/(m-1)}} \tag{7}$$



(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

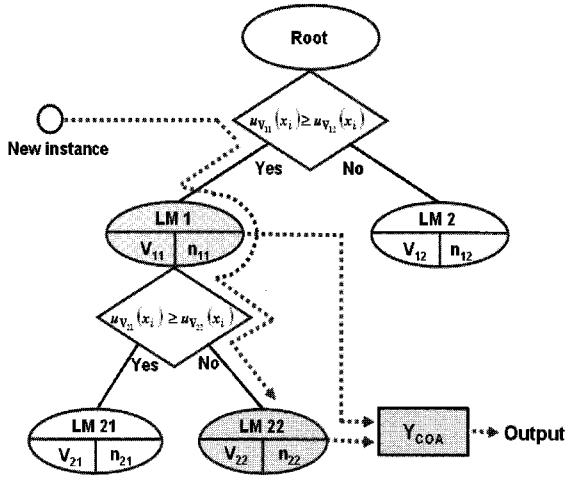


그림 2. 제안된 방법의 추론 과정
Fig. 2. Inducing process of the proposed method

여기서, m 은 퍼지화 정도를 나타내는 퍼지 수로써 일반적으로 2를 이용한다. 또한, d_{ik} 는 q 차원을 갖는 입력 데이터 X_i 와 k 번째 대표 중심값 V_k 의 유클리디안 거리값을 의미한다.

최종 출력 Z_i 는 비퍼지화방법으로 일반적으로 사용되고 있는 무게 중심법을 이용하여 다음과 같이 얻어진다.

$$Z_i = \frac{\sum_{j=1}^{p} \hat{y}_i(j) \cdot u_{ij}}{\sum_{j=1}^{p} u_{ij}} \tag{8}$$

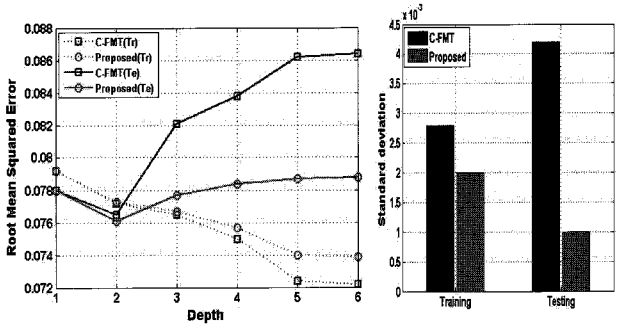
3. 실험 및 결과

본 논문에서 제안된 방법의 타당성을 평가하기 위하여 표 2에서 보인 벤치마크 데이터를 이용하였다[10]. 실험에서는 입력과 출력데이터를 각각 [0, 1]로 정규화 하였으며, 모델트리의 깊이 변화에 따른 특성을 비교하기 위해서 훈련과 검증 데이터가 많은 데이터들을 대상으로 하여 트리의 깊이를 1에서 6까지 변화시키며 실험하였다. 표 3에서는 클러스터 기반 퍼지 모델트리에 의해 생성된 Abalone 데이터의 구조를 나타내었다. 생성된 모델트리는 각 노드에서의 중심값들과 구분 분류된 데이터들을 이용한 선형 모델들로 이루어진다. 본 논문에서는 기존의 방법에서는 말단 잎노드에서의 선형 모델만을 고려하는 반면 제안된 방법에서는 말단 잎노드에 도달하기까지의 상위 노드의 선형 모델들도 함께 고려하는 방법을 제안하였다.

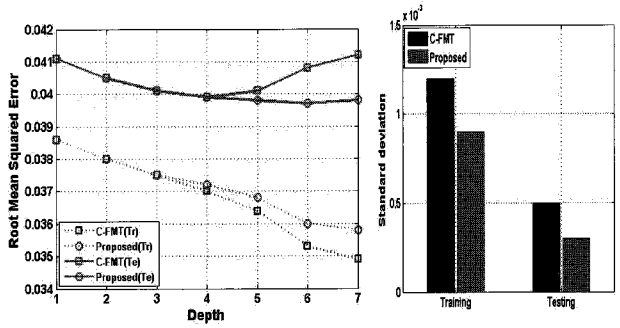
표 2. 벤치마크 데이터의 사양
Table 2. Specification of benchmark data sets

Data sets	#Observations		#Attributes	
	Training	Test	Continuous	Nominal
Abalone	2000	2177	7	1
Delta ailerons	3000	4129	6	0
Delta elevators	4000	5517	6	0
Computer activity	4000	4192	8	0

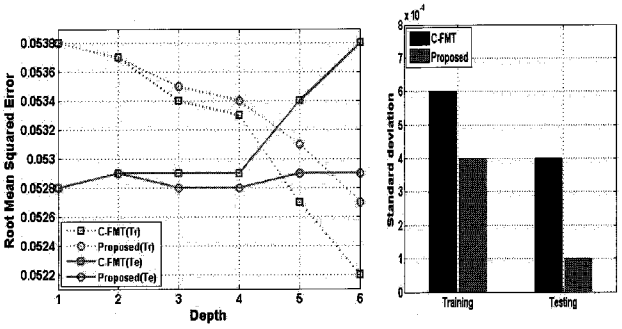
그림 3에서는 각각의 벤치마크 데이터에서 모델트리의 깊이 변화에 따른 특성들을 보여주고 있다. 아울러 표 4에서는 여러 가지 비교척도를 이용하여 두 방법을 비교 분석하였다. 그림 3에서 알 수 있듯이 일반적으로 모델트리에서 트리의 깊이가 증가 할수록 훈련 데이터의 지역적 특성만을 고려하므로 오차는 점차적으로 감소하는 반면, 어느 정도 깊이 이상에서는 과도한 훈련 데이터의 적응 등으로 인한 과잉적응의 문제가 발생함을 확인 할 수 있다. 특히 Abalone 데이터의 경우 검증 데이터에 대한 RMSE(Root Mean Squared Error)를 고려할 때 기존 방법의 경우 트리의 깊이가 1에서 6까지 변화하면서 표준편차가 ±0.0042로 크게 변화하므로 분할 정지 시점을 잘못 선정할 경우 훈련 데이터에 대한 오차는 충분히 낮아지지만, 전체 성능에 미치는 영향이 매우 크다는 것을 알 수 있다.



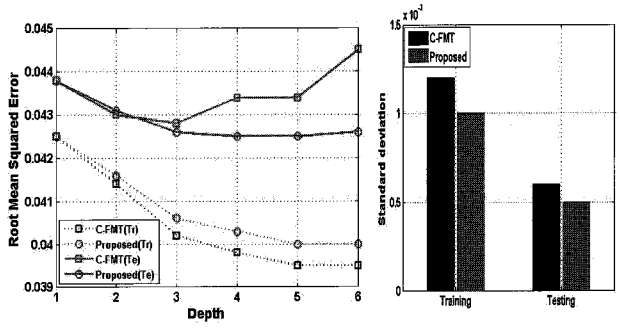
(a) Abalone



(b) Delta ailerons



(c) Delta elevators



(d) Computer activity

그림 3. 모델 트리의 깊이 변화에 따른 특성 비교
Fig. 3. Comparison of quality follows in depth change of model tree

그러나 상호 노드간의 정보를 고려하는 제안된 방법에서는 표준편차가 ±0.0010으로 기존의 방법에 비하여 검증 데이터에 대하여 일관성 있는 성능을 가지고 있음을 확인할 수 있었다. 또한 다른 벤치마크 데이터들에서도 동일한 성능을 얻을 수 있었다. 표 5에서는 트리의 깊이를 변화시키면서 가장 우수한 성능을 보일 때의 두 방법을 비교하였다. 제안된

표 3. Abalone 데이터의 클러스터 기반 퍼지 모델트리 구조
Table 3. Cluster-based fuzzy model tree for Abalone dataset

Depth	Model No.	Center	Linear equation
0	Root	-	$\hat{y}=-0.11\cdot x_1+0.34\cdot x_2+0.74\cdot x_3+1.24\cdot x_4-1.22\cdot x_5-0.36\cdot x_6+0.11\cdot x_7+0.07$
1	LM 1	[0.727,0.719,0.147,0.425, 0.351,0.344,0.338]	$\hat{y}=-0.45\cdot x_1+0.27\cdot x_2+0.23\cdot x_3+1.26\cdot x_4-1.05\cdot x_5-0.32\cdot x_6+0.18\cdot x_7+0.35$
	LM 2	[0.460,0.441,0.094,0.138, 0.112,0.111,0.115]	$\hat{y}=-0.05\cdot x_1+0.17\cdot x_2+1.00\cdot x_3+1.39\cdot x_4-1.67\cdot x_5-0.22\cdot x_6+0.50\cdot x_7+0.07$
2	LM 11	[0.685,0.674,0.137,0.349, 0.288,0.282,0.282]	$\hat{y}=-0.29\cdot x_1+0.05\cdot x_2+0.27\cdot x_3+1.75\cdot x_4-1.38\cdot x_5-0.51\cdot x_6+0.27\cdot x_7+0.34$
	LM 12	[0.799,0.794,0.163,0.555, 0.462,0.450,0.433]	$\hat{y}=-0.61\cdot x_1+0.59\cdot x_2+0.05\cdot x_3+0.94\cdot x_4-0.80\cdot x_5-0.22\cdot x_6+0.15\cdot x_7+0.28$
	LM 21	[0.551,0.534,0.111,0.194, 0.157,0.158,0.161]	$\hat{y}=-0.27\cdot x_1+0.04\cdot x_2+0.82\cdot x_3+2.52\cdot x_4-2.20\cdot x_5-0.32\cdot x_6+0.18\cdot x_7+0.21$
	LM 22	[0.341,0.319,0.073,0.065, 0.052,0.052,0.055]	$\hat{y}=0.03\cdot x_1+0.34\cdot x_2+0.57\cdot x_3-1.23\cdot x_4-0.53\cdot x_5+0.38\cdot x_6+1.63\cdot x_7+0.03$

표 4. 벤치마크 데이터에 대한 실험 결과 (Mean)
Table 4. Experimental results for benchmark dataset (Mean)

Data sets	Correlation coefficient		Mean absolute error		Root mean squared error		Relative absolute error(%)		Root relative squared error (%)	
	C-FMT	Proposed method	C-FMT	Proposed method	C-FMT	Proposed method	C-FMT	Proposed method	C-FMT	Proposed method
Abalone	0.6966 ±0.0251	0.7227 ±0.0065	0.0563 ±0.0012	0.0550 ±0.0003	0.0822 ±0.0042	0.0780 ±0.0010	68.1580 ±1.5210	66.5369 ±0.4082	73.4242 ±3.7076	69.6622 ±0.8832
Delta ailerons	0.8341 ±0.0049	0.8389 ±0.0024	0.0285 ±0.0004	0.0282 ±0.0002	0.0404 ±0.0005	0.0400 ±0.0003	49.9685 ±0.6861	49.3513 ±0.4286	55.3686 ±0.6657	54.7380 ±0.3936
Delta elevators	0.7963 ±0.0033	0.7988 ±0.0002	0.0404 ±0.0004	0.0401 ±0.0001	0.0531 ±0.0004	0.0529 ±0.0001	54.9670 ±0.5385	54.5756 ±0.0575	60.5607 ±0.4500	60.2139 ±0.0248
Computer activity	0.9486 ±0.0014	0.9500 ±0.0013	0.0269 ±0.0004	0.0265 ±0.0005	0.0435 ±0.0006	0.0429 ±0.0005	25.7010 ±0.3391	25.3530 ±0.4951	31.7054 ±0.4261	31.2359 ±0.3880

표 5. 벤치마크 데이터에 대한 실험 결과 (Best)
Table 5. Experimental results for benchmark dataset (Best)

Data sets	Correlation coefficient		Mean absolute error		Root mean squared error		Relative absolute error(%)		Root relative squared error (%)	
	C-FMT	Proposed method	C-FMT	Proposed method	C-FMT	Proposed method	C-FMT	Proposed method	C-FMT	Proposed method
Abalone	0.6725	0.7178	0.0548	0.0546	0.0765	0.0761	66.3682	66.0441	68.3908	68.0402
Delta ailerons	0.8266	0.8287	0.0281	0.0280	0.0399	0.0397	49.2453	49.0526	54.7047	54.4377
Delta elevators	0.7904	0.7966	0.0401	0.0400	0.0528	0.0528	54.5298	54.5126	60.2030	60.1740
Computer activity	0.9462	0.9462	0.0265	0.0262	0.0428	0.0425	25.3410	25.0433	31.2222	30.9585

표 6. 벤치마크 데이터에 대한 실험 결과
Table 6. Experimental results for benchmark dataset

Data sets	BP[9]		M5P		Fuzzy cluster based Model Tree					
					Root node		C-FMT		Proposed method	
	Training	Testing	Training	Testing	Training	Testing	Training	Testing	Training	Testing
Abalone	0.0715	0.0779	0.0774	0.0764	0.0811	0.0781	0.0722	0.0765	0.0773	0.0761
Delta ailerons	0.0340	0.0411	0.0367	0.0402	0.0386	0.0411	0.0370	0.0399	0.0360	0.0397
Delta elevators	0.0506	0.0545	0.0536	0.0528	0.0544	0.0531	0.0538	0.0528	0.0534	0.0528
Computer activity	0.1565	0.1742	0.1728	0.1566	0.0434	0.0445	0.0402	0.0428	0.0400	0.0425

방법이 기존의 방법에 비하여 큰 성능향상을 얻지는 못하였지만, 실험 결과를 토대로 분석한 결과 제안된 방법이 트리의 깊이 변화에 대한 오차의 일관성을 유지함은 물론 최종 성능에서도 기존의 방법에 비하여 향상된 성능을 가지고 있음을 확인 할 수 있다.

마지막으로 데이터 마이닝에서 널리 사용되는 역전파(BP:Back-Propagation)알고리즘과 모델트리에서 가장 진보된 알고리즘인 M5P과 비교하였다.

M5P는 데이터 마이닝에 널리 쓰이는 WEKA 프로그램에서 실행하였다[11]. 표 6에서는 각각의 알고리즘으로 실험한 결과를 나타내었다. 모델트리의 경우 루트노드에서의 오차값과 분할 후 잎노드에서의 오차값을 별도로 기입하였다. 표에서 보는 바와 같이 모든 데이터에 대하여 인공신경망에 의한 회귀방법보다 트리구조의 M5P과 퍼지 클러스터 기법을 이용한 방법이 우수한 것으로 나타났다. 분석 결과 기존의 모델트리 방식에 비해 제안된 상호 노드 정보를 이용하는 방법이 가장 우수한 결과를 보임을 확인할 수 있었다.

4. 결 론

본 논문에서는 기존의 클러스터 기반 퍼지 모델트리에서 트리의 깊이에 따른 훈련 및 검증데이터의 일관성 문제점을 해결하기 위해 상호 노드간의 정보를 고려하는 방법을 제안하였다. 제안된 방법은 먼저 입력과 출력변수의 속성을 고려한 퍼지 클러스터링에 의해 중심벡터를 계산한 후, 중심벡터들과 입력 속성간의 소속도를 이용하여 구간 분할된 영역별로 각각의 선형모델을 구축한다. 예측 단계에서는 입력된 데이터가 잎노드에 도달하는 노드간의 중심벡터와 입력 데이터간의 거리값에 따른 소속도를 계산한 후 최종적으로 무게 중심법을 이용하여 출력값을 예측하게 된다. 제안된 방법의 우수성을 보이기 위해 다양한 벤치마크 데이터를 대상으로 실험한 결과, 기존의 클러스터 기반 퍼지 모델트리보다 향상된 성능을 보임을 확인하였다. 본 논문에서 제안된 방법은 클러스터 기반 퍼지 모델트리를 대상으로 하였으며, 향후 과제로 다양한 트리 기반의 알고리즘에 적용하는 방안에 대하여 연구할 계획이다.

참 고 문 헌

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

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

[3] Karalic A, "Linear regression in regression tree leaves", *in Proceedings of ISSEK'92*, Bled, Solvenia, 1992.

[4] 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.

[5] 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.

[6] 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.

[7] Witold Pedrycz, "C-Fuzzy Decision Trees", *IEEE Trans. on System, Man, and Cybernetics*, Part C, Vol. 35, No. 4, pp. 498-511, 2005.

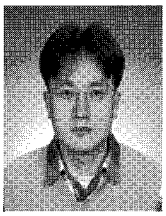
[8] 이대종, 박진일, 전명근외, "클러스터 기반 퍼지 모델트리를 이용한 데이터 모델링", *한국퍼지 및 지능시스템학회 논문지*, Vol. 16, No. 5, pp. 608-615, 2006.

[9] Guang-Bin Huang and et al, "Extreme learning machine:Theory and applications", *Neurocomputing*, Vol. 70, pp. 489-501, 2006.

[10] http://www.sgi.com/tech/mlc/db

[11] http://www.cs.waikato.ac.nz/~ml/weka/index.html

저 자 소 개

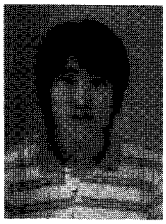


박진일(Jin Il Park)
2001년 : 한밭대학교 제어계측공학과(학사)
2003년 : 한밭대학교 제어계측공학과 (공학석사)
2005년~현재 : 충북대학교 제어계측공학과 박사과정

관심분야 : 지능시스템, 다중생체인식, 퍼지이론
E-mail : moralskr@yahoo.co.kr

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 충북정보기술단 초빙조교수

관심분야 : 데이터마이닝, 얼굴인식, 다중생체인식
E-mail : djmidori@empal.com



김용삼 (Yong Sam Kim)
2006년 : 충북대학교 전기전자공학과(학사)
2006년～현재 : 충북대학교 제어계측공학과 석사과정

관심분야 : 다중생체인식, 신호 처리, 임베디드 프로그래밍
E-mail : yszzang99@paran.com

조영임 (Young Im Cho)
1987년 : 고려대학교 생물학과(학사)
1988년 : 고려대학교 컴퓨터과학과(학사)
1990년 : 고려대학교 일반대학원 컴퓨터학과(이학석사)
1994년 : 고려대학교 일반대학원 컴퓨터학과(이학박사)
1995년～1996년 : 삼성전자 멀티미디어연구소 선임연구원
1999년～2000년 : University of Massachusetts Postdoc.
1996년～2005년 : 평택대학교 컴퓨터과학과 교수
2005년～현재 : 수원대학교 IT대학 컴퓨터학과 교수

관심분야 : 인공지능, Biometrics, 에이전트, 뉴로-퍼지
E-mail : ycho@suwon.ac.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 전기전자컴퓨터공학부 교수

관심분야 : Biometrics, 패턴인식, 데이터마이닝
E-mail : mgchun@chungbuk.ac.kr