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

조선대학교
CHOSUN UNIVERSITY

[UCI]I804:24011-200000266279

2017년 8월

석사학위논문

TSK 퍼지 규칙과 랜덤 클러스터링을 이용한 ELM 예측기의 설계

조 선 대 학 교 대 학 원

제어계측공학과

염 찬 욱

조선대학교
CHOSUN UNIVERSITY

TSK 퍼지 규칙과 랜덤 클러스터링을 이용한 ELM 예측기의 설계

Design of ELM Predictor using TSK Fuzzy Rule and Random Clustering

2017년 8월 25일

조 선 대 학 교 대 학 원

제어계측공학과

염 찬 욱

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

TSK 퍼지 규칙과 랜덤 클러스터링을 이용한 ELM 예측기의 설계

지도교수 곽 근 창

이 논문을 공학석사학위신청 논문으로 제출함.

2017년 4월

조 선 대 학 교 대 학 원

제어계측공학과

염 찬 욱

dCollection @ chosun

조선대학교
CHOSUN UNIVERSITY

염찬욱의 공학석사학위논문을 인준함

위원장 조선대학교 교수 장 순 석 인

위 원 조선대학교 교수 반 성 범 인

위 원 조선대학교 교수 곽 근 창 인

2017년 5월

조선대학교 대학원

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

목 차

제1장 서론 ····· 1

제2장 ELM 모델 ····· 5

제1절 ELM의 구조 ····· 5

제2절 ELM의 분류기와 예측기 ····· 8

제3절 ELM의 활성 함수 ····· 12

제3장 TSK 기반의 ELM 예측기 ····· 16

제1절 TSK 기반 ELM 모델의 구조 ····· 16

제2절 TSK 기반 ELM 예측기의 설계 ····· 19

제4장 실험 및 결과분석 ····· 22

제1절 함수 데이터 예측 문제 ····· 23

제2절 Boston Housing 데이터 예측 문제 ····· 28

제3절 Auto-MPG 데이터 예측 문제 ····· 34

제5장 결론 ····· 42

참고문헌 ····· 44

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

표 목 차

표 4-1. 컴퓨터 실험 환경 ····· 22
표 4-2. 함수 데이터를 사용한 예측기 성능 결과 ····· 28
표 4-3. Boston Housing 데이터를 사용한 예측기 성능 결과 ····· 34
표 4-4. Auto-MPG 데이터를 사용한 예측기 성능 결과 ····· 40
표 4-5. TSK 기반의 ELM 예측기에서 생성되는 전제부 파라미터 값 ····· 40
표 4-6. TSK 기반의 ELM 예측기에서 생성되는 파라미터 값 ····· 41

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

그림 목차

그림 2.1 Extreme Learning Machine 구조 ····· 5
그림 2.2 Extreme Learning Machine의 학습 과정 ····· 7
그림 2.3 Extreme Learning Machine 분류기 구조 ····· 8
그림 2.4 Extreme Learning Machine 예측기 구조 ····· 10
그림 2.5 시그모이드 활성 함수 출력 ····· 12
그림 2.6 하드 리미트 활성 함수 출력 ····· 13
그림 2.7 ReLU 활성 함수 출력 ····· 14
그림 3.1 TSK 기반 ELM 예측기 구조 ····· 19
그림 4.1 함수 데이터 출력 값 ····· 23
그림 4.2 정규화 과정을 하지 않은 함수 검증 데이터 출력 값 ····· 24
그림 4.3 정규화 과정을 하지 않은 함수 데이터 평균 제곱근 편차 값 ····· 25
그림 4.4 정규화 과정을 한 함수 검증 데이터 출력 값 ····· 26
그림 4.5 정규화 과정을 한 함수 데이터 평균 제곱근 편차 값 ····· 27
그림 4.6 Boston Housing 데이터 출력 값 ····· 29
그림 4.7 정규화 과정을 하지 않은 Boston Housing 검증 데이터 출력 값 ····· 30
그림 4.8 정규화 과정을 하지 않은 Boston Housing 데이터 평균 제곱근 편차 값 ··· 31
그림 4.9 정규화 과정을 한 Boston Housing 검증 데이터 출력 값 ····· 32
그림 4.10 정규화 과정을 한 Boston Housing 데이터 평균 제곱근 편차 값 ····· 33
그림 4.11 Auto-MPG 데이터 출력 값 ····· 35
그림 4.12 정규화 과정을 하지 않은 Auto-MPG 검증 데이터 출력 값 ····· 36
그림 4.13 정규화 과정을 하지 않은 Auto-MPG 데이터 평균 제곱근 편차 값 ····· 37
그림 4.14 정규화 과정을 한 Auto-MPG 검증 데이터 출력 값 ····· 38
그림 4.15 정규화 과정을 한 Auto-MPG 데이터 평균 제곱근 편차 값 ····· 39

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

ABSTRACT

Design of ELM Predictor using TSK Fuzzy Rule and Random Clustering

Yeom, Chan-Uk
Advisor : Prof. Kwak, Keun Chang, Ph. D.
Dept. of Control and Instrumentation Eng.,
Graduate School of Chosun University

In this paper propose an ELM predictor using TSK fuzzy rule and random clustering method to improve the prediction performance of ELM predictor. The TSK-based ELM model consists of a structure that uses a linear function in place of a weight and sets the center of the cluster at random. There is no weight between the input layer and the hidden layer, and the weight between the hidden layer and the output layer is a linear equation. We propose method generates meaningful rules by using the if-then rule. Also, it's improves the performance of the predictor by randomly generating the cluster center through the initial membership matrix. Experiments were compared with existing ELM predictors using function approximation, Boston Housing, and Auto-MPG data. Experimental results showed that the proposed method showed better performance than the conventional ELM predictor.

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

제1장 서론

머신 러닝(Machine Learning)은 인공지능의 한 분야로서, 데이터라는 형태로 얻어지는 경험(Experience)으로부터 특정한 목표 작업(Task)에 대한 성능(Performance)을 향상 시키는 일련의 과정으로 정의된다. 머신러닝은 컴퓨터가 대량의 데이터로부터 귀납적 추론을 통해서 스스로 새로운 지식을 도출 가능하게 한다[1]. 이러한 머신 러닝은 크게 지도 학습 또는 교사 학습(Supervised Learning)과 비지도 학습 또는 자율 학습(Unsupervised Learning)으로 나눌 수 있다. 지도 학습은 각각의 객체 별로 속성에 대한 입력 벡터와 목표치가 주어진 학습 데이터들이 존재 할 때, 이 데이터들로부터 하나의 함수를 유추하여 새로운 객체에 대한 목표치를 예측하는 방법이다. 지도 학습은 주로 예측(Prediction) 문제에 활용되며, 목표치의 형태에 따라서 분류(Classification) 문제, 회귀(Regression) 문제 등에 활용된다. 비지도 학습은 데이터를 설명해 줄 수 있는 특성을 발견하기 위한 방법으로 군집화(Clustering) 문제, 밀도 추정(Density Estimation), 그리고 차원 축소(Dimensionality Reduction) 등에 활용된다. 그 밖에도 지도 학습과 비지도 학습의 특성을 결합한 반지도 학습(Semi-supervised Learning)[2]과 현재 상태에서 어떠한 행동을 취하는 것이 최적인지를 학습하는 강화 학습(Reinforcement Learning)[3]이 있다.

머신 러닝의 지도 학습의 방법인 ELM(Extreme Learning Machine) 모델은 Huang 등에 의하여 제안된 방법으로, 방사 기저 함수(RBF, Radial Basis Function)를 가지는 단일층 전향 신경망을 기반으로 하는 구조를 가지고 있다. ELM 모델은 여러 분야에서 많은 연구가 진행 되었다.

이론적인 연구로는 딥 러닝(Deep Learning)을 기반으로 한 커널 ELM(KELM, Kernel Extreme Learning Machine)[4] 모델, ELM 모델과 같은 배치 학습(Batch Learning) 방법이 발표된 이후, 데이터들이 빅데이터화 되면서 데이터 집합의 크기가 거대해졌다. 이러한 경우 은닉층 출력 행렬의 크기 또한 커지기 때문에 계산 시간과 알고리즘을 개발하는데 많은 노력이 필요하다. 앞에서 언급된 문제들을 해결하기 위해 학습을 위한 데이터를 덩어리 단위 (Chunk-by-Chunk) 혹은 하나씩 (One-by-One) 처리하고, 새로운 데이터가 입력될 때 마다 다시 학습 할 필요가 없는 순차적 ELM(OS-ELM, Online Sequential-Extreme Learning Machine)[5] 모델,

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

불균형적인 클래스 분포를 가지고 있는 데이터를 처리하기 위해서 데이터를 평형 데이터로 일반화 시키는 가중치 ELM(WELM, Weighted Extreme Learning Machine)[6] 모델, 네트워크에 숨겨진 노드 매개 변수가 이전의 학습에서 얻은 값으로 매개 변수 값을 생성하는 자기 적응형 진화 ELM(SAE-ELM, Self-Adaptive Evolutionary Extreme Learning Machine)[7] 모델 등이 연구 되어왔다.

ELM 분류기에 대한 연구로는 Huang[8]이 제안한 분류기로, 일반적인 신경회로망에서 사용되고 있는 역전파(Back-Propagation) 알고리즘은 학습 하는 시간이 길다는 단점이 있다. 이와 같은 단점을 보완하기 위해서 입력층과 은닉층 사이의 가중치의 학습을 생략하는 방법이다. Niki Martienel[9]은 컨볼루션 신경회로망(Convolution Neural Networks)과 ELM 분류기를 결합한 성능이 컨볼루션 신경회모랑과 서포트 벡터 머신(SVM, Support Vector Machine)을 결합한 분류기보다 높은 분류 성능을 보였다. Le AN[10]는 ELM 분류기를 기반으로 원본 고해상도의 이미지의 고주파 구성 요소들을 대상 값으로 사용하고, 저해상도 이미지의 구성 요소를 ELM 분류기에 학습 한 뒤, 저해상도 이미지가 주어지면 학습을 통해서 고해상도 이미지가 생성되는 연구를 하였다. Huang[11]은 입력을 각각의 데이터가 아닌 지역(Field)을 사용하는 국부 지역 수신 ELM(ELM-LRF, Extreme Learning Machine Local Receptive Fields) 모델을 연구하였고, 컨볼루션 신경회로망보다 높은 분류 성능을 보였다. Alexandros losifidis[12]는 동작 표현은 퍼지 벡터 양자화를 하고, 동작 분류에서는 피드 포워드 신경망(Feedforward Neural Network)을 통해 시각에 독립적인 수행하는 최소 클래스 분산 ELM(Minimum Class Variance Extreme Learning Machine) 모델을 제안한다. Zhige Xie[13]는 학습 3D 형상을 분할하여 각각의 세그먼트들을 ELM 분류기를 통해 학습하고, 검증 단계에선 앞에서 얻어진 세그먼트들과 검증 3D 형상의 세그먼트들 사이의 활성 윤곽을 구해 최종적인 세그먼트들을 구하는 연구를 하였다. Yujun Zeng[14]는 기존의 교통 표지판 인식 성능을 향상시키기 위해서 색 공간을 이용한 커널 ELM(Deep Perceptual-Kernel Extreme Learning Machine)을 연구하였고, 기존의 방법들 보다 높은 인식 성능을 보였다. Kun Han[15]은 음성 감정 인식에서 음성 데이터로부터 높은 수준의 특징을 추출하기 위해 딥 신경회로망(DNN, Deep Neural Network)를 이용하여 음성 세그먼트에 대한 감정 상태 확률 분포를 생성한다. 그런 다음 세그먼트 수준의 확률 분포로부터 발화 수준의 특징을 구성하고, 이런 발화 수준 특징을 ELM을 통해 음성을 분류하는 음성 감정 인식 분류기를 연구하였다.

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

최근에는 예측 문제에 대한 연구도 활발하게 진행되고 있다. 국내에서 지방 소도시의 단기 물 수요 예측의 적용성을 높이기 위한 ELM 모델 연구[16], ELM을 사용하여, 기상요소를 분석하고 일별 태양광 발전량을 예측하는 모델 연구[17], 그리고 ELM을 이용한 일별 최대 전력 수요 예측 알고리즘 개발[18], 특수일 최대 전력수요 예측 모델 개발[19], 정수장 운영 효율을 올리기 위한 ELM 기반의 물 수요 예측기 연구[20] 등이 있다. 국외에서는 광전자 신호를 예측하는 ELM 모델[21], 환자의 신체 상태를 나타내는 데이터를 이용하여 병을 진단하는 ELM 모델[22-23], 그리고 OS-ELM을 이용한 독일의 네카강(Neckar River)의 수위 예측 모델 연구[24]를 진행했다. Xia Chen[25]은 전기 가격 예측에 사용되던 인공 신경회로망(ANN, Artificial Neural Network)의 역전파 알고리즘 등에 의해 학습 속도가 느리고, 국부 최소해(Local Minimum)에 빠질 수 있는 단점을 보완하기 위해 피드 포워드 신경회로망 기반의 ELM 예측기를 사용하여 학습 시간을 단축하였고, 좋은 성능을 보였다. Beatriz Saavedra-Moreno[26]는 풍력 발전소의 운영에 사용되는 측정 상관 예측 방법(MCP, Measure-Correlate-Predict Methods)을 그룹 데이터 처리 방법(GMDH, Group Method of Data Handling)과 ELM를 통해 예측하는 연구를 하였다. Hui-Xin Tian[27]은 자체 적응으로 임계 값을 수정하는 변형된 에이다부스트(AdaBoost)와 ELM을 결합한 앙상블 ELM(Ensemble Extreme Learning Machine) 예측기를 개발하여 강철의 온도를 예측하는 연구를 하였다. Qi Yu[28]는 ELM 예측기의 성능을 일반화시키고 은닉층의 노드 수를 줄여 구조적인 복잡함을 해결하기 위해, 델타-테스트 ELM(DT-ELM, Delta Test-Extreme Learning Machine) 예측기를 개발하였다. Jianzhou Wang[29]는 확률적 단기 풍속 예측을 위한 경험적 웨이브렛 변환(EWT, Empirical Wavelet Transform)과 가우시안 프로세스 회귀(GPR, Gaussian Process Regression)과 ELM을 결합한 예측기를 연구하였다.

위 연구들의 결과들을 보면 ELM 분류기의 분류 성능은 대체적으로 잘 나오는 것을 알 수 있다. 하지만 ELM 예측기의 예측 성능은 일정하지 못했다. 그 이유로 첫 번째 데이터가 저차원이거나, 고차원 또는 선형적이거나 비선형적으로 데이터의 형태에 따라서 예측 성능이 달랐다. 두 번째 입력층과 은닉층 사이에 있는 가중치의 문제로 의미 있는 규칙을 찾지 못한다. 마지막으로 세 번째 성능을 높이기 위해서 은닉층 노드의 수를 크게 해야 한다.

따라서, 본 논문에서는 ELM 예측기의 문제점들을 보완 할 수 있도록 TSK 퍼지 규칙(Takagi-Sugeno-Kang Fuzzy Rule)과 랜덤 클러스터링(Random Clustering)을

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

이용한 ELM 예측기를 설계한다.

TSK 퍼지 규칙과 랜덤 클러스터링을 이용한 ELM 예측기는 기존에 사용된 퍼지 클러스터링(Fuzzy Clustering) 방법의 활성 함수가 아닌 랜덤한 방법에 의해 클러스터 중심을 초기화하는 방법을 사용한다. 이렇게 함으로써 의미 있는 퍼지 규칙을 생성하고, 적은 수의 규칙으로도 좋은 성능을 나타 낼 수 있다. 제안된 알고리즘의 성능을 확인하기 위해 1차 함수, 2차 함수, 다차원 데이터 등의 다양한 예측 문제들을 적용하여 기존의 방법과 성능을 비교하였으며, 본 논문에서 제안하는 알고리즘이 우수한 성능을 보였다.

본 논문의 구성은 다음과 같다. 1장은 서론으로 시작하여, 2장에서는 ELM 모델에 대한 설명과 분류기, 예측기에 대한 설명을 한다. 3장은 제안하는 방법인 TSK 기반의 ELM 예측기에 사용된 TSK 퍼지 모델과 랜덤 클러스터링 방법들에 대한 설명하고, 4장에서는 데이터베이스에 대한 설명과 실험 결과 및 분석을 한다. 마지막으로 5장에서는 결론으로 논문을 마무리 한다.

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

제2장 ELM 모델

제1절 ELM의 구조

ELM 모델은 Huang[8, 30] 등에 의하여 제안된 방법으로, 방사 기저 함수(RBF, Radial Basis Function)를 가지는 단일층 전향 신경망을 기반으로 한다. 구조는 입력층, 은닉층, 그리고 출력층으로 되어 있다. 그림 2.1은 ELM 모델의 구조를 나타낸다[31].

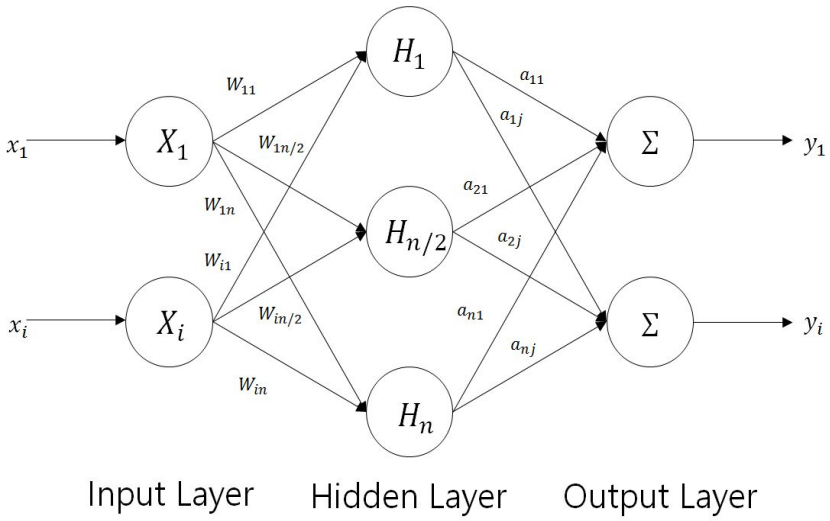


그림 2.1 Extreme Learning Machine 구조

은닉층의 노드 수는 사용자에 의해 결정된다. 입력층과 은닉층, 은닉층과 출력층 사이에는 서로를 연결해 주는 가중치 w, a가 존재하며, 입력 값 x를 출력 값 y로 대응 시켜주는 비선형 함수의 역할을 한다. 은닉층에 사용되는 기저 함수로는 일반적으로 시그모이드(Sigmoid) 함수를 주로 사용한다. 최급경사법(Grdient Descent Method) 기반의 일반적인 신경회로망의 학습은 좋은 성능을 얻기 위해 많은 반복 학습을 필요로 한다. 하지만 적절하지 못한 학습들로 인하여 낮은 적응성을 보이거나 종종 국부 최소해에 수렴하거나 과잉적합(Over Fitting) 되는 단점을

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

가지고 있다. 그러나 ELM 모델의 출력 가중치들은 반복적인 학습 구조 대신에 모어 페로스의 일반화 역행렬(Moore-Penrose Generalized Inverse)에 의해 분석적으로 계산되어짐으로 반복적인 학습을 하지 않는다.

ELM 모델은 일부 파라미터들을 임의의 값으로 고정시키고 한 번의 학습을 통해 모델을 구성하는 신경회로망이다. 학습 속도가 빠르면서 좋은 성능을 보여준다. 식(2.1)은 ELM의 기본적인 구조를 나타낸다. 식(2.1)으로 ELM 모델의 학습 과정을 설명한다[32].

$$H(w_i x_j)a = y_j,\ j = 1, 2, \ldots, m \tag{2.1}$$

식(2.1)에서 w_i와 a는 임의로 할당되는 가중치이며, x_j는 입력 값을 y_j는 출력 값을 나타낸다. 여기서 w_i는 i번째 은닉층 노드와 입력층들의 사이를 연결하는 가중치로 임의의 값으로 초기화된다. a는 은닉층 노드들과 출력층을 연결하는 가중치이다. $w_i x_j$는 서로의 내적을 의미하기 때문에 식(2.1)은 식(2.2), 식(2.3), 식(2.4)로 표현 할 수 있다.

$$Ha = y \tag{2.2}$$

여기서, 식(2.2)에 있는 행렬을 풀어보면 식(2.3), 식(2.4), 식(2,5)과 같이 표현 할 수 있다.

$$\begin{aligned} &H(w_1, w_2, \ldots, w_i, x_1, x_2, \ldots, x_j) \\ &= \begin{bmatrix} H(w_1 x_1) & \ldots & H(w_i x_1) \\ \vdots & \vdots & \vdots \\ H(w_i x_j) & \ldots & H(w_i x_j) \end{bmatrix} \end{aligned} \tag{2.3}$$

$$a = \begin{bmatrix} a_1 \\ \vdots \\ a_N \end{bmatrix}_T \tag{2.4}$$

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

$$y = \begin{bmatrix} y_1 \\ \vdots \\ y_j \end{bmatrix}_{j \times 1} \tag{2.5}$$

식(2.3)의 행렬 H는 전방향 신경회로망에서 은닉층의 출력을 나타낸다. 행렬 H를 최소화 시키는 최소 자승법(LSE, Least Square Estimation)의 해는 a로 식(2.4)과 같다. 식(2.6)은 최소 자승방법을 사용한 것을 보여준다.

$$a = \min_a \| Ha - y \| \tag{2.6}$$

위의 식(2.6)에서 구해진 값에서 행렬 H를 무어-페로스의 일반화된 역행렬로 표현하게 되면 식(2.7)과 같이 표현 할 수 있다.

$$\hat{a} = H^{\dagger} y \tag{2.7}$$

따라서, ELM 모델은 기존의 신경회로망의 학습 방법들과는 다르게 학습을 한 번만 하기 때문에 학습 시간이 빠르고, 학습 오차 값이 작아 최적의 가중치 값을 얻을 수 있다. 그림 2.2은 위에서 설명한 ELM 모델의 학습 과정을 표현한 것이다.

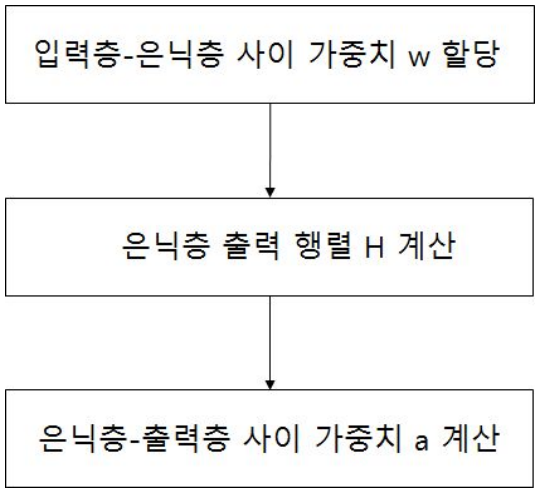


그림 2.2 Extreme Learning Machine의 학습 과정

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

제2절 ELM의 분류기와 예측기

ELM 모델의 종류로는 크게 두 가지로 1장 서론에서 언급했던 ELM 분류기와 ELM 예측기가 있다. ELM 모델의 장점인 학습을 한 번만 진행하기 때문에 학습 속도가 빠르고, 과적합, 국부 최소해와 같은 문제들로 인한 성능 저하의 문제점을 해결했기 때문에 일반적인 분류기와 예측기 보다 높은 성능을 보인다[33]. 2절에서는 ELM 분류기와 ELM 예측기의 구조를 설명하고 학습 방법에 대해서 설명한다.

1. ELM 분류기

ELM 분류기는 ELM 모델의 학습 방법을 이용하여 임의의 데이터 집합이 있을 때, 그안에 있는 각각의 데이터들의 특징들을 파악하여 같거나 비슷한 데이터들을 나누어 준다. 종류로는 국부 지역 수신 ELM 분류기, 최소 클래스 분산 ELM 분류기, 이러한 ELM 분류기들 중에서 은닉층을 하나만 가지는 신경회로망을 단일 은닉층 피드 포워드 네트워크(SLFNs, Single Hidden Layer Feedforward Networks)라고 부른다[34].

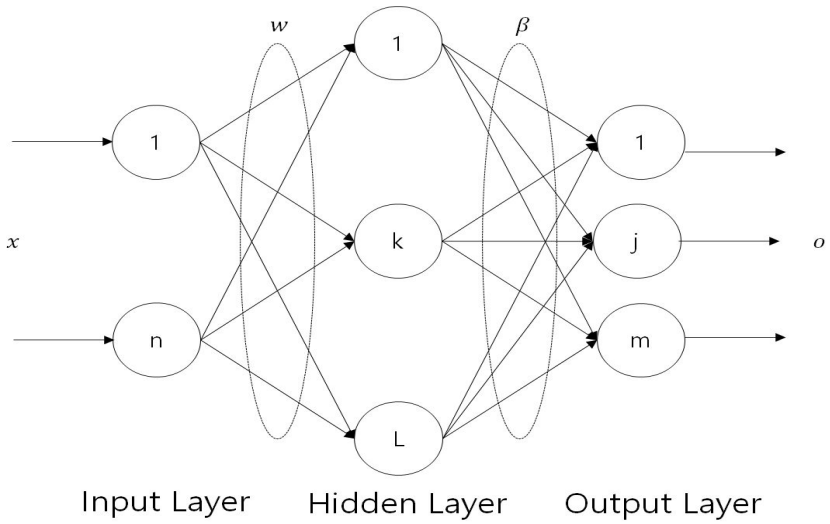


그림 2.3 Extreme Learning Machine 분류기 구조

dCollection @ chosun

조선대학교
CHOSUN UNIVERSITY

그림 2.3은 단일 은닉층 피드 포워드 네트워크의 구조를 나타낸다. ELM 분류기에서 입력 데이터는 $x_i = [x_{i1}, \ldots, x_{is}]^T \in R^s$이고, 출력 데이터는 $t_i = [t_{i1}, \ldots, t_{im}]^T \in R^m$인 데이터 집합 (x_i, t_i)이 있을 때, 은닉층의 노드 수는 Z이고, $g(x)$의 가중치를 가지는 단일 은닉층 피드 포워드 네트워크를 식(2.8)로 나타 낼 수 있다.

$$o_j = \sum_{i=1}^{Z} \beta_i \cdot g_i(x_j) = \sum_{i=1}^{Z} \beta_i \cdot g(w_i \cdot x_j + b_i) \tag{2.8}$$

o_j는 j번째의 출력층 노드의 출력 값을 나타낸다. Z는 은닉층의 노드의 수를 나타낸다. $g_i(x_j)$는 은닉층의 i번째 은닉층 노드의 출력을 나타낸다. w_i는 입력층과 i번째 은닉층 사이에 있는 가중치를 나타낸다. b_i는 i번째 은닉층 노드에 있는 한계(Threshold) 값을 나타낸다[8].

$$\sum_{i=1}^{Z} \beta_i \cdot (w_i \cdot x_j + b_i) = t_j,\ j = 1, \ldots, N \tag{2.9}$$

식(2.9)는 ELM 모델의 구조에 나오는 식(2.1)과 같은 의미를 가지고 있다. 여기서, N은 데이터 샘플들의 수를 나타낸다. 이 과정에서 $w_1, \ldots, w_Z$와 $b_1, \ldots, b_Z$를 임의의 확률 분포 함수를 통해 추출하여 사용된다.

$$\mathrm{H}\beta = T \tag{2.10}$$

식(2.10)은 식(2.9)를 행렬로 나타낸 것이다. 식(2.10)의 풀이 법은 1절 ELM 구조에서 설명한 학습 과정부분에 있는 식(2.3), 식(2.4), 식(2.5)과 같은 방법으로 이루어진다. 이 과정에서 최소 자승방법을 이용하여 목적 함수를 최소화 할 수 있는 β를 결정한다. 최종적으로 식(2.11)과 같이 나타낼 수 있다.

$$\hat{\beta} = H^{+} T \tag{2.11}$$

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

2. ELM 예측기

ELM 예측기는 전력 수요 예측, 태양광 발전량 예측, 단기 물 수요 예측 등의 분야에 사용되고 있는 예측기이다. ELM 예측기는 임의의 데이터 집합이 있을 때, 입력으로 과거 또는 현재까지 수집된 데이터를 넣고 ELM 모델의 학습 방법을 이용하여 각각의 데이터들 사이의 상관관계를 분석하여, 그 다음 데이터를 예측 하는 것이다. ELM 예측기에 대한 연구는 국내보다는 국외에서 활발히 이루어지고 있다. ELM 예측기의 구조는 그림 2.4와 같이 표현 할 수 있다[35-36].

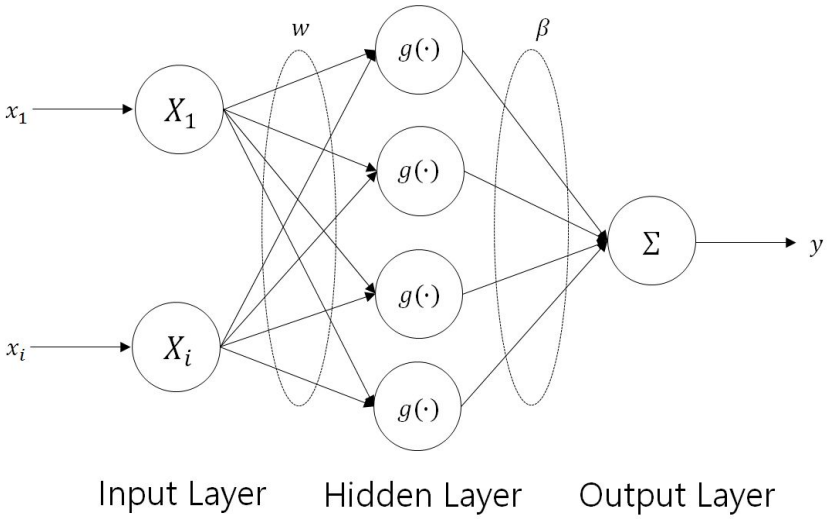


그림 2.4 Extreme Learning Machine 예측기 구조

그림 2.4를 보면 알 수 있듯이 ELM 예측기의 구조에서 입력층과 은닉층까지는 ELM 분류기와 유사한 것을 알 수 있지만, 차이점은 출력층에서 나오는 출력 값이 하나인 것을 볼 수 있다. ELM 예측기의 과정은 ELM 분류기의 과정과 유사하며, 식 (2.12)처럼 간결하게 표현 할 수 있다.

$$g\beta = y \tag{2.12}$$

dCollection @ chosun

조선대학교
CHOSUN UNIVERSITY

식(2.12)를 행렬의 형태로 변환하면, 식(2.13)으로 표현 할 수 있다.

$$H(w_{1,}w_{2,}\ldots,w_{i,}x_{1,}x_{2,}\ldots,x_{j}) = \begin{bmatrix} g(w_1x_1) & g(w_2x_1)\ldots & g(w_ix_1) \\ \vdots & \vdots & \vdots \\ g(w_ix_j) & g(w_ix_i)\ldots & g(w_ix_j) \end{bmatrix} \tag{2.13}$$

행렬 H는 전방 신경회로망에서 은닉층의 출력을 나타내며, 최종적으로 최소자승법에 의하여 출력 가중치 $\widehat{H}$값을 구한다. 식(2.14), 식(2.15)와 같이 표현할 수 있다.

$$\beta = \min_{\beta} \| H\beta - y \| \tag{2.14}$$

$$\widehat{H} = g^{\dagger} y \tag{2.15}$$

식(2.15)에서 $H^{\dagger}$값은 은닉층 H의 모어-페로스의 일반화된 역행렬을 나타낸다. 만약에 $H^{\dagger}H$가 역행렬이 존재하지 않는 비정칙 행렬일 경우, H의 모어-페로스의 일반화된 역행렬은 식(2.16)과 같이 표현된다.

$$H^{\dagger} = (H^{\dagger}H)^{-1}H^{\dagger} \tag{2.16}$$

최종적으로 예측된 값은 식(2.17)로 표현 할 수 있다.

$$\widehat{y} = H\widehat{\beta} \tag{2.17}$$

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

3절 ELM의 활성 함수

ELM 모델의 구조에서 각 층에 있는 노드들의 주요 기능은 입력과 연결 강도의 가중치 합을 구한 다음 활성 함수(Activation Function)에 의해 계산 된 후에, 그 값을 출력으로 내보낸다. 여기서, 활성 함수는 입력층과 은닉층 사이, 은닉층과 출력층 사이에 있으며 일정(Threshold) 값 미만에서는 0을 출력하고, 이상의 값에서는 1을 출력 하는 함수이다. 어떠한 활성 함수를 사용하느냐에 따라 은닉층의 노드에서 나오는 출력 값이 달라지기 때문에 중요한 역할을 한다. 그 중에서 많이 사용되고 있는 시그모이드(Sigmoid) 함수, 하드 리미트(Hard Limit) 함수, 그리고 정류한 선형(ReLU, Rectified Linear Unit) 함수들을 설명한다.

1. 시그모이드 활성 함수

시그모이드 함수는 S 자 형태를 가지는 비선형 곡선 함수로 성장 곡선을 반영할 수 있는 모형에 자주 이용 되는 함수이다. 시그모이드 함수의 형태는 그림 2.5와 같이 나타 낼 수 있고, 식(2.18)로 표현 할 수 있다[37].

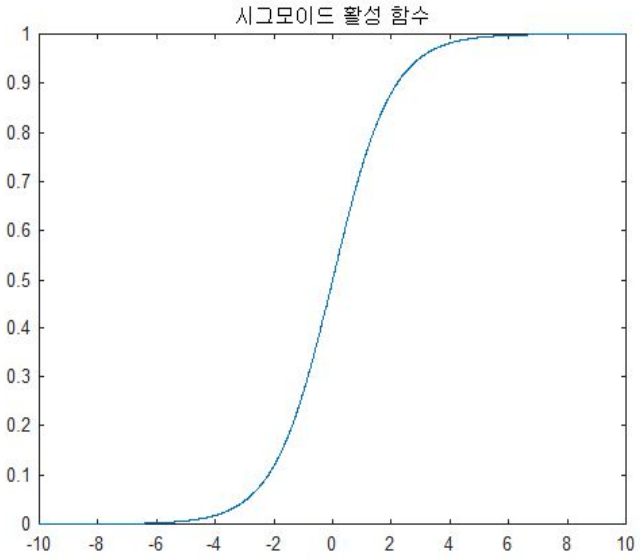


그림 2.5 시그모이드 활성 함수 출력

dCollection @ chosun

조선대학교
CHOSUN UNIVERSITY

$$f(x) = \frac{1}{1 + \exp - \alpha(\beta - t)} \tag{2.18}$$

식(2.18)에서 exp는 자연로그를 나타내며, α값은 시그모이드 활성 함수의 기울기 값, β값은 입력 값, 그리고 t값은 이동 값을 의미한다. 시그모이드 활성 함수의 조작 변수로는 기울기 값을 담당하는 α값이 있다. α값이 -10에 가까울수록 입력 범위인 β값은 수평에 가까워지며, α값이 -5 미만 인 경우의 입력 범위의 출력은 $y=x$에 가까운 형태를 보인다.

2. 하드 리미트 활성 함수

하드 리미트 활성 함수는 계단 형태를 가지고 있어서 계단(Step) 활성 함수 또는 부호(Sign) 활성 함수로도 불린다. 시그모이드 활성 함수의 경우 출력의 변화가 부드러운 것을 볼수 있다. 하지만, 하드 리미트 활성 함수는 임의의 점을 계기로 출력 값의 변화가 심하게 일어나는 것을 그림 2.6에서 확인 할 수 있다[38].

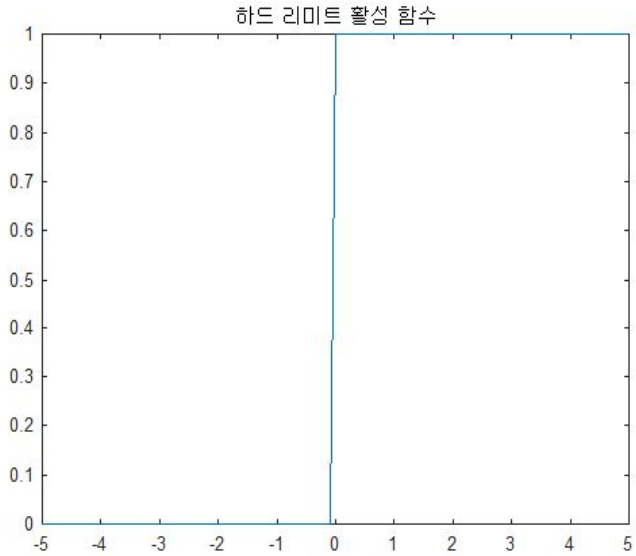


그림 2.6 하드 리미트 활성 함수 출력

dCollection @ chosun

조선대학교
CHOSUN UNIVERSITY

식(2.19)는 하드 리미트 활성 함수를 나타낸다.

$$f(x) = \begin{cases} 1, & \text{if } X \geq 0 \\ 0, & \text{if } X < 0 \end{cases} \tag{2.19}$$

식(2.19)는 0을 중심으로 X가 0보다 크거나 같을 경우, 출력 값이 1이 되고, X가 0보다 작을 경우, 출력이 0이 되는 것을 보여준다.

3. ReLU 활성 함수

ReLu는 Rectified Linear Unit의 약자로 정류한 선형 유닛 활성 함수로 해석할 수 있다. ReLu는 Hahnloser[39]에 의해서 Nature지를 통하여 처음 소개 되었다. 그러나 신경회로망의 활성 함수로 사용되어진지 얼마 되지 않았다. 그림 2.7은 ReLU의 출력을 보여준다.

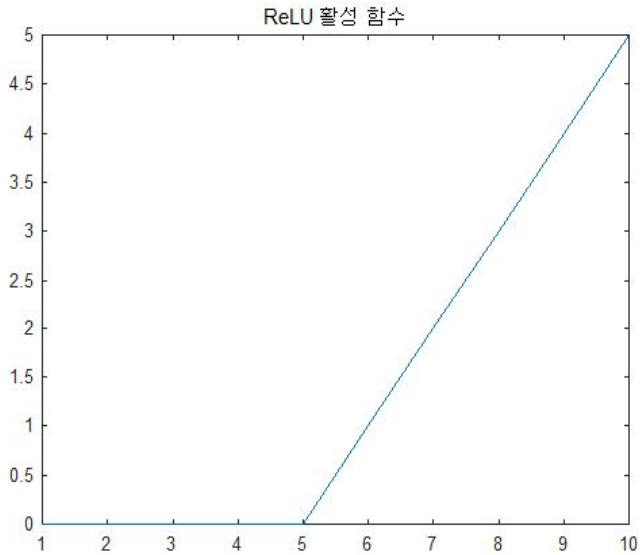


그림 2.7 ReLU 활성 함수 출력

dCollection @ chosun

조선대학교
CHOSUN UNIVERSITY

ReLU 활성 함수는 시그모이드 활성 함수에서 0과 1 사이의 값을 가지는데, 경사도 하강(Gradient Descent)을 사용하여 역전파 알고지름을 수행 할 때 경사도를 계속 곱하기 때문에 0으로 수렴하게 되고 층이 많아져 작동에 문제점이 있다. 따라서 이러한 문제점을 해결하기 위해 ReLU 활성 함수가 사용되었다.

ReLU 활성 함수는 입력 값이 0보다 작으면 0이 되고, 입력 값이 0보다 크면 입력 값 그대로 출력된다. 식(2.20)은 ReLU 활성 함수의 특징을 보여준다[40].

$$f(x) = \begin{cases} (x<0) & f(x) = 0 \\ (x \geq 0) & f(x) = x \end{cases} \tag{2.20}$$

ReLU 활성 함수의 다른 장점으로는 경사도의 왜곡이 없는 점과 선형 함수이므로 미분 계산이 매우 간단하다[41].

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

3장 TSK 기반의 ELM 예측기

제1절 TSK 기반 ELM 모델의 구조

위에서 언급 했듯이 ELM 예측기의 예측 성능은 데이터들의 형태에 따라서 예측 성능이 달랐고, 입력층과 은닉층 사이에 있는 가중치의 문제로 의미 있는 규칙을 찾지 못하는 경우가 있다. 그리고 성능을 높이기 위해서 은닉층 노드의 수를 과하게 늘려야 하는 문제들이 있다.

본 절에서는 TSK 퍼지 규칙과 랜덤 클러스터링을 이용한 ELM 예측기를 제안한다. 제안하는 방법은 그림 3.1과 같이 표현 할 수 있다. TSK 기반의 ELM 예측기의 구조는 TSK 퍼지 모델과 ELM 모델에 의해 변형된 구조를 가지고 있다. 다음은 TSK 퍼지 모델 기반의 ELM 예측기에 사용된 TSK 퍼지 모델과 랜덤 클러스터링 방법에 대하여 설명한다.

TSK 퍼지 모델은 Tomohiro Takagi, Michio Sugeno, GeunTaek Kang에 의해서 제안 된 적응 뉴로-퍼지 추론 시스템(ANFIS, Adaptive Neuro Fuzzy Inference System)의 한 종류로 세 명의 이름을 따서 TSK 퍼지 모델이라고 부른다[42]. 비선형 시스템을 적은 수의 규칙들로 표현 할 수 있는 능력을 갖고 있다[43-45]. 퍼지를 이용한 모델링을 하는 TSK 퍼지 모델은 전반부 과정은 비선형 시스템을 표현하는데 있어서 매우 중요하다. 전반부 과정에서는 입력 변수의 선택과 맴버쉽 함수의 형태와 수를 결정한다. 이를 통해서 퍼지 공간이 분할 되고, 퍼지 공간의 수가 곧 퍼지 규칙 수가 된다. 이 부분은 TSK 모델을 구성하는데 있어서 중요한 역할을 차지한다. 후반부는 선형 함수의 집합으로 구성된다. TSK 퍼지 모델은 식(3.1)로 표현 할 수 있다.

$$\begin{aligned} R^i\text{: if } x_1 \, is \, A^i_1(c^i_1, s^i_1) \text{ and} \\ x_2 \, is \, A^i_2(c^i_2, s^i_2) \text{ and}, \ldots \\ x_n \, is \, A^i_n(c^i_n, s^i_n) \\ then \\ y^i = f_i(x_1, x_2, \ldots, x_n) \\ = a^i_0 + a^i_1 x_1 + a^i_2 x_2 + \ldots + a^i_n x_n \\ i = 1, 2, \ldots, r \end{aligned} \tag{3.1}$$

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

식(3.1)에서 R^i은 i번째 퍼지 규칙을 나타낸다. x_i값은 입력 변수, y_i값은 출력, 그리고 r값은 퍼지 규칙의 수를 나타낸다. 아래에서 TSK 퍼지 모델의 구조에 대하여 간략하게 설명한다.

첫 번째 층에서 각 노드는 언어적인 레벨의 소속도를 발생하며, 식 (3.2)와 같이 소속 값을 출력으로 갖는다.

$$O_i^1 = \mu_{A_i}(y), \quad O_{i+2}^1 = \mu_{B_i}(y) \qquad i = 1, 2 \tag{3.2}$$

전제부 소속 함수로 식 (3.3)과 같은 가우시안 소속 함수를 선택한다.

$$\mu_{A_i}(x) = \exp\left\{-\left(\frac{x - c_i}{a_i}\right)^2\right\} \tag{3.3}$$

여기서, 는 전제부 파라미터이며, a_i는 소속 함수의 폭을 나타내고, c_i는 소속 함수의 중심을 나타낸다. 두 번째 층에서는 각 노드는 첫 번째 층에서 구한 소속 값을 규칙별로 곱한 가중치로 출력되며, 식 (3.4)와 같이 나타낼 수 있다.

$$O_i^2 = w_i = \mu_{A_j^i}(x) \times \mu_{B_j^i}(y) \qquad i = 1, 2 \tag{3.4}$$

세 번째 층에서는 노드 i는 모든 점화력의 합에 대한 i번째 규칙의 점화력의 비를 나타낸다. 다시 말하자면, 식 (3.5)과 같이 w_i에 대해 정규화 된 값을 출력한다.

$$O_i^3 = \overline{w_i} = \frac{w_i}{w_1 + w_2} \qquad i = 1, 2 \tag{3.5}$$

네 번째 층에서 노드 i는 식 (3.6)과 같이 정규화 된 값과 결론부의 곱으로 출력을 나타낸다.

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

$$O_i^4 = \overline{w}_i f_i = \overline{w}_i(p_i x + q_i y + r_i) \quad i = 1,2 \tag{3.6}$$

여기서, $\overline{w}_i$는 세 번째 층에서 얻어진 정규화 된 점화력(normalized firing strengths)을 나타낸다. 다섯 번째 층에서는 가중 평균법에 의해 최종적인 출력을 구 할 수 있다.

$$O_i^5 = y_i^* = \sum_{i=1}^{2} \overline{w}_i f_i = \frac{\sum w_i f_i}{\sum w_i} \tag{3.7}$$

TSK 퍼지 모델에서 일반적으로 맴버쉽 함수 파라미터 동정은 입력 변수의 최소값과 최대 값 사이를 임의의 개수로 등분해서 일률적으로 정의하고, 성능 지수가 최소가 되는 위치를 결정하게 된다. 그러나 데이터들 마다 가지고 있는 특징이 다르기 때문에, 그 특징에 따라 맴버쉽 함수의 파라미터를 정의하는 것이 중요하다.

TSK 퍼지 모델에서 전제부 파라미터가 주어질 때, 전체적인 출력 f는 다음 식 (3.8)과 같은 결론부 파라미터의 선형적인 조합에 의해 나타 낼 수 있다.

$$\begin{aligned} f &= \overline{w_1} f_1 + \overline{w_2} f_2 \\ &= (\overline{w_1} x) p_1 + (\overline{w_1} y) q_1 + (\overline{w_1}) r_1 + (\overline{w_2} x) p_2 + (\overline{w_2} y) q_2 + (\overline{w_2}) r_2 \end{aligned} \tag{3.8}$$

여기서, 결론부 파라미터는 최적의 파라미터의 효과적인 탐색을 위해 최소 자승법으로 추정된다. 최소 자승법으로 결론부 파라미터를 추정하는 알고리즘에 대한 설명은 식 (3.8)과 같다. 전제부 파라미터와 P개의 학습 데이터의 값이 주어지면 식 (3.8)을 식 (3.9)와 같이 결론부 파라미터에 의한 P개의 선형 방정식을 만들 수 있다.

$$\begin{bmatrix} \overline{w_1}^{(1)} & \overline{w_1}^{(1)} x^{(1)} & \overline{w_1}^{(1)} y^{(1)} & \overline{w_2}^{(1)} & \overline{w_2}^{(1)} x^{(1)} & \overline{w_2}^{(1)} y^{(1)} \\ \overline{w_1}^{(2)} & \overline{w_1}^{(2)} x^{(2)} & \overline{w_1}^{(2)} y^{(2)} & \overline{w_2}^{(2)} & \overline{w_2}^{(2)} x^{(2)} & \overline{w_2}^{(2)} y^{(2)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \overline{w_1}^{(p)} & \overline{w_1}^{(p)} x^{(p)} & \overline{w_1}^{(p)} y^{(p)} & \overline{w_2}^{(p)} & \overline{w_2}^{(p)} x^{(p)} & \overline{w_2}^{(p)} y^{(p)} \end{bmatrix} \begin{bmatrix} r_1 \\ p_1 \\ q_1 \\ r_2 \\ p_2 \\ q_2 \end{bmatrix} = \begin{bmatrix} d^{(1)} \\ d^{(2)} \\ \vdots \\ \vdots \\ \vdots \\ d^{(p)} \end{bmatrix} \tag{3.9}$$

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

여기서, $(x^{(k)}, y^{(k)}), d^{(k)}$은 k번째 학습 데이터를 나타내며, 식 (3.9)를 행렬로 나타내면 식 (3.10)과 같이 나타 낼 수 있다.

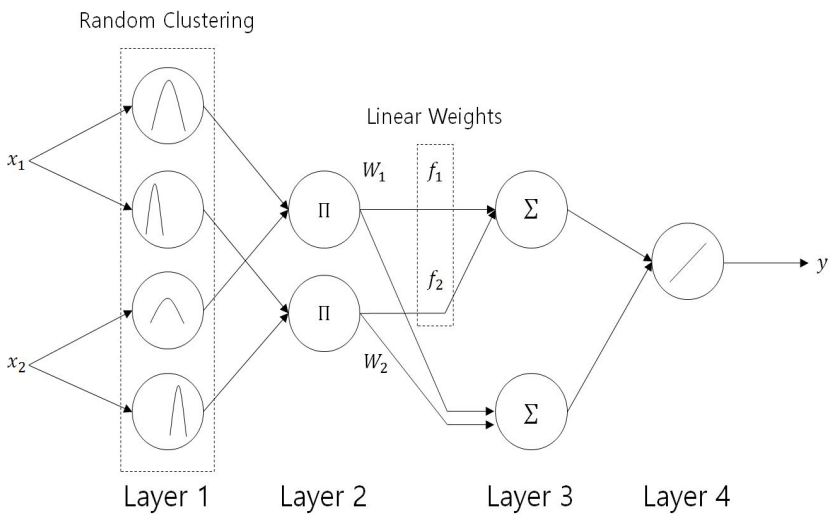
$$A\Theta = B \tag{3.10}$$

여기서, Θ는 결론부 파라미터 요소를 가진 미지의 벡터이며, 각각의 A, Θ, B의 차원은 $P \times M$, $M \times 1$, $P \times 1$이며, P는 학습 데이터의 수를 나타내고, M은 선형 파라미터의 수를 나타낸다. P는 M보다 보통 더 크기 때문에 일반적으로 식 (3.10)에 대한 정확한 해는 찾을 수 없게 된다. 대신에, 최소 자승법을 이용하여 $||A\Theta - B||^2$을 최소화하도록 의사 역행렬을 이용하여 식 (3.11)과 같이 구할 수 있다.

$$\overline{\Theta} = (A^T A)^{-1} A^T B \tag{3.11}$$

제2절 TSK 기반 ELM 예측기의 설계

본 논문에서는 TSK 퍼지 모델과 랜덤 클러스터링 방법을 이용한 ELM 예측기를 제안한다. 그림 3.1은 TSK 기반 ELM 예측기의 구조를 나타낸다.



Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

그림 3.1 TSK 기반 ELM 예측기 구조

그림 3.1과 같이, TSK 기반 ELM 예측기는 4개의 층을 가지고 있다. 첫 번째 층에 있는 모든 노드는 랜덤 클러스터링 방법에 의해 이루어진다. 여기서, 맴버쉽 함수는 가우시안 함수와 같은 적절한 매개 변수가 있는 맴버쉽 함수를 사용한다. 가우시안 맴버쉽 함수는 식(3.12)와 같이 두 개의 매개 변수 $\{c, \sigma\}$에 의해서 결정된다.

$$g(x, c, \sigma) = e^{-\frac{1}{2}\left(\frac{x-c}{\sigma}\right)^2} \tag{3.12}$$

식(3.12)에서 $\{c, \sigma\}$는 클러스터의 중심과 표준 편차를 나타낸다. 여기서 j번 째 입력 변수에 대한 표준 편차는 식(3.13)와 같이 계산된다.

$$\sigma_j = \alpha \frac{x_{\max} - x_{\min}}{2\sqrt{2}} \tag{3.13}$$

여기서, α는 각각의 데이터 차원에서 영향을 미치는 클러스터의 중심의 범위를 지정한다. $x_{\min}$와 $x_{\max}$은 입력 공간의 각 차원에서 최소 값과 최대 값을 나타낸다. 실제 최소 값과 최대 값의 범위가 0인 경우, 표준 편차 값을 계산 할 수 있도록 데이터를 기준으로 작은 범위를 계산한다.

랜덤 클러스터링 방법은 초기 맴버쉽 행렬에 의해 무작위로 얻은 클러스터의 중심을 결정한다. 맴버쉽 행렬 U는 0과 1 사이의 랜덤한 값으로 초기화 되어 식(3.14)과 같은 제약 조건이 만족된다.

$$\sum_{i=1}^{c} u_{ij} = 1, \quad \forall j = 1, 2, \ldots, n \tag{3.14}$$

그런 다음, 클러스터의 중심은 식(3.15)로 계산된다.

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

$$C_i = \frac{\sum_{i=1}^{n} u_{ij}^m x_j}{\sum_{j=1}^{n} u_{ij}^m} \tag{3.15}$$

첫 번째 TSK 퍼지 유형인 경우, 중요한 규칙의 일반적인 형태는 if-then 규칙을 가진다.

$$R^1 : If\ x_1\ is\ A_1 \text{ and } x_2\ is\ A_2,\ then\ f = a_1 x_1 + a_2 x_2 + a_0$$

여기서, $\{a_1, a_2, a_0\}$는 그림 3.1에서 보는 것과 같이 2층과 3층 사이의 선형 가중치 f_i를 갖는 결과적인 파라미터이다. 이러한 선형 매개 변수는 단 한 번의 순방향 학습에서 최소 자승법에 의해 추정된다. 매개 변수는 ELM과 동일한 방식으로 경사도 하강으로 업데이트 되지 않는다.

두 번째 층에 있는 모든 노드는 고정된 노드이며, 출력은 식(3.16)과 같이 들어오는 신호의 결과이다.

$$w_i = g(x_i; c_{11}, \sigma_1) g(x_2; c_{21}, \sigma_2) \tag{3.16}$$

여기서, 각각의 노드의 출력 w_i는 규칙의 실행 강도를 나타낸다. 세 번째 층과 네 번째 층의 모든 노드는 정규화 된 소성 강도 및 가중 평균법에 의해 수행된다. 식(3.17)과 같이 나타 낼 수 있다.

$$\hat{y} = \frac{\sum_i w_i f_i}{\sum_i w_i} \tag{3.17}$$

클러스터의 중심의 수를 결정하기 위해서 본 논문에서는 학습 데이터와 검증 데이터의 평균 제곱근 편차(RMSE, Root Mean Square Error)값을 반복적으로 수행해서 얻는다. 따라서, 클러스터의 중심의 수가 증가함에 따라 검증 데이터의 최적의 평균 제곱근 편차를 나타내는 클러스터의 중심의 수를 결정한다.

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

제4장 실험 및 결과분석

본 장에서는 기존에 사용되는 ELM 예측기와 제안된 방법인 TSK 기반의 ELM 예측기의 성능을 비교 평가한다. 컴퓨터의 실험 환경은 표 4-1과 같이 이루어져 있다. 실험에 사용된 데이터는 총 세 가지로 함수 데이터, Boston Housing 데이터, 그리고 MPG 데이터가 사용되었다. 실험 방법은 위에서 설명한 데이터들을 정규화하지 않은 상태, 정규화 한 상태, 두 가지 상황에서 ELM 예측기와 TSK 기반의 ELM 예측기를 사용하여 예측 성능을 비교한다. 예측 성능은 평균 제곱근 편차(RMSE, Root Mean Square Error)로 나타낸다. 평균 제곱근 편차는 추정 값 또는 예측기의 예측 값과 실제 환경에서 관찰되는 값의 차이를 나타낼 때 사용하는 측도이다 [46]. 식(4.1)과 같이 나타 낼 수 있다.

$$RMSE = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(y_j - \widehat{y}_j)^2} \tag{4.1}$$

여기서, y_i값은 결과 값이고, $\widehat{y}_i$값은 실제 값을 나타낸다. $y_i = \widehat{y}_i$ 일 경우, 100%의 예측 성능을 갖게 되며 평균 제곱근 편차는 0이 된다. 그러므로, 평균 제곱근 편차 값이 높으면 성능이 좋지 않고, 낮으면 성능이 좋다는 것을 알 수 있다.

표 4-1 컴퓨터 실험 환경

Division		Contents
HW	CPU	Intel(R) Core(R) 17-6700 3.40 GHz
	RAM	8.00 GB
	HDD	1 TB
SW	OS	Windows 10
	Program	Matlab R2016A

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

제1절 함수 데이터 예측 문제

함수 데이터는 Jyh-Shing Roger Jang의 Neuro-Fuzzy and Soft Computing 책의 12장 ANFIS의 세 번째 예제에 사용된 데이터로, 함수의 예측 문제에 사용되는 데이터이다[45]. 데이터의 크기는 200×2로, 1개의 입력과 1개의 출력으로 나타난다. 함수 데이터의 구조는 식(4.2)와 같이 나타난다.

$$f(u) = 0.6\sin(\pi u) + 0.3\sin(3\pi u) + 0.1\sin(5\pi u) \tag{4.2}$$

그림 4.1은 함수 데이터의 출력 값을 보여준다.

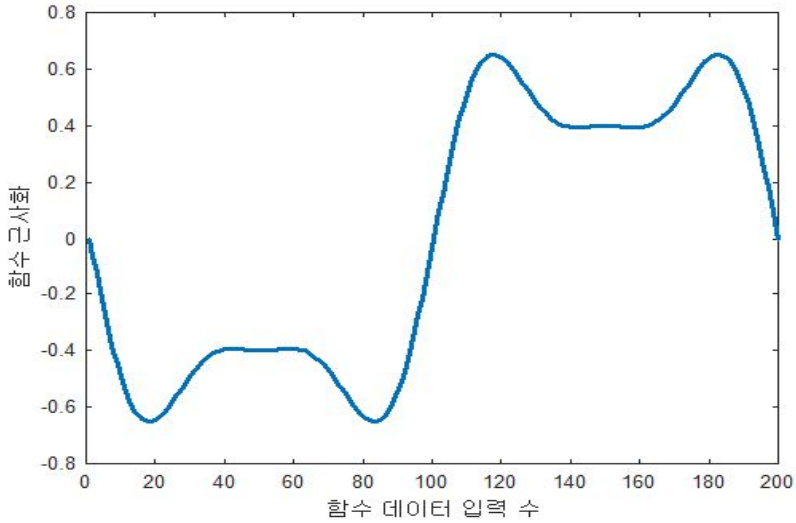


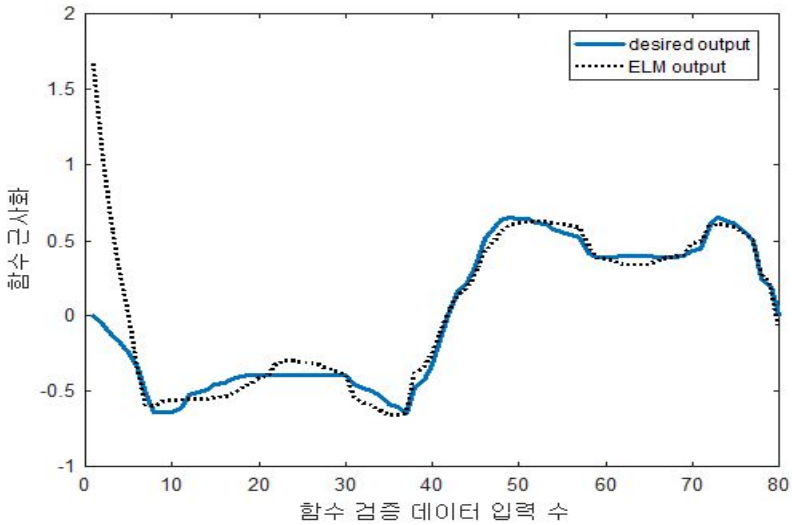
그림 4.1 함수 데이터의 출력 값

본 실험은 정규화 과정 여부에 따른 두 번의 실험을 진행하여 ELM 예측기와 TSK 기반의 ELM 예측기의 예측 성능을 비교했다. 그림 4.2은 정규화 과정을 하지 않은 함수 데이터에 대한 ELM 예측기와 TSK 기반의 ELM 예측기의 검증 출력을 나타내며, 그림 4.3는 정규화 과정을 하지 않은 데이터를 이용한 ELM 예측기와 TSK-ELM 예측기의 평균 제곱근 편차 값을 나타낸다. 그리고 그림 4.4는 정규화 과정을 한 함수 데이터에 대한 ELM 예측기와 TSK 기반의 ELM 예측기의 검증 출력

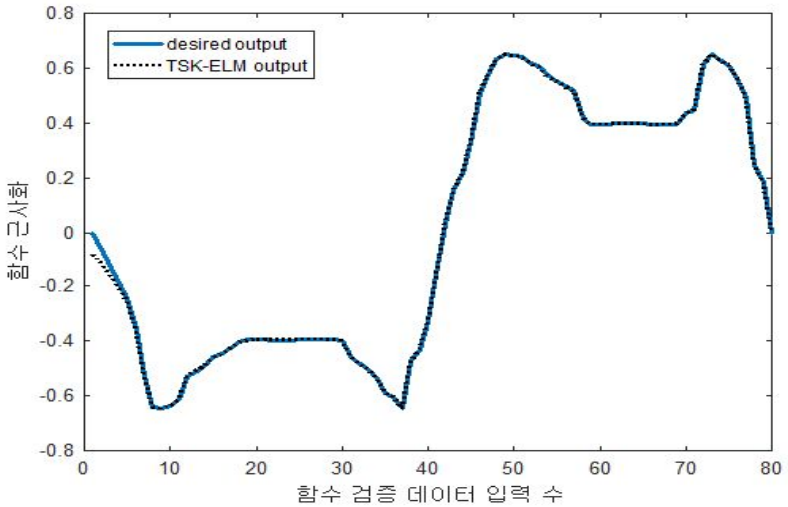
Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

을 나타내며, 그림 4.5은 정규화 과정을 한 데이터를 이용한 ELM 예측기와 TSK-ELM 예측기의 평균 제곱근 편차 값을 나타낸다. 함수 데이터를 사용한 두 예측기의 성능 결과는 표 4-2과 같이 나왔다. 표 4-2에서는 두 예측기의 최적의 성능을 가졌을 때의 구조도 함께 표기하였다. ELM 예측기는 은닉층의 노드 수를 표기 하였고, TSK 기반의 ELM 예측기는 퍼지 규칙의 수를 표기하였다.



(a) 기존 ELM 예측기의 검증 출력 값

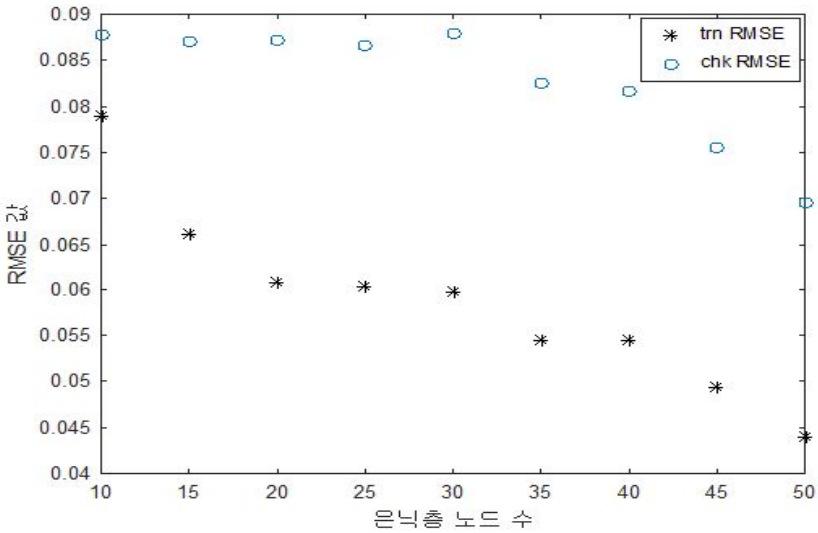


(b) TSK-ELM 예측기의 검증 출력 값

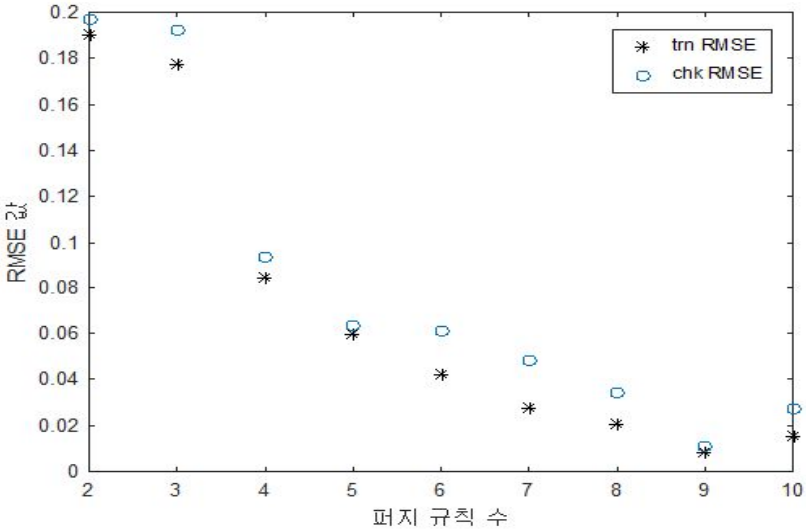
Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

그림 4.2 정규화 과정을 하지 않은 함수 검증 데이터 출력 값



(a) 기존 ELM 예측기의 평균 제곱근 편차 값

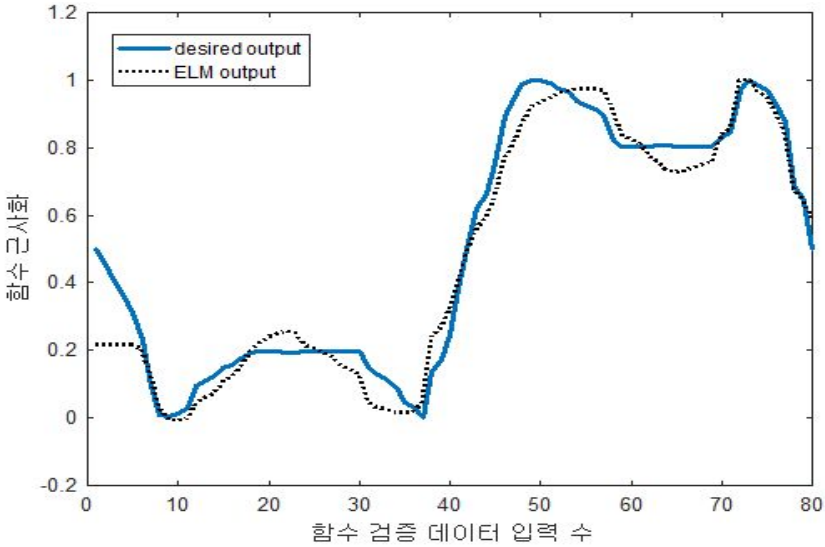


(b) TSK-ELM 예측기의 평균 제곱근 편차 값

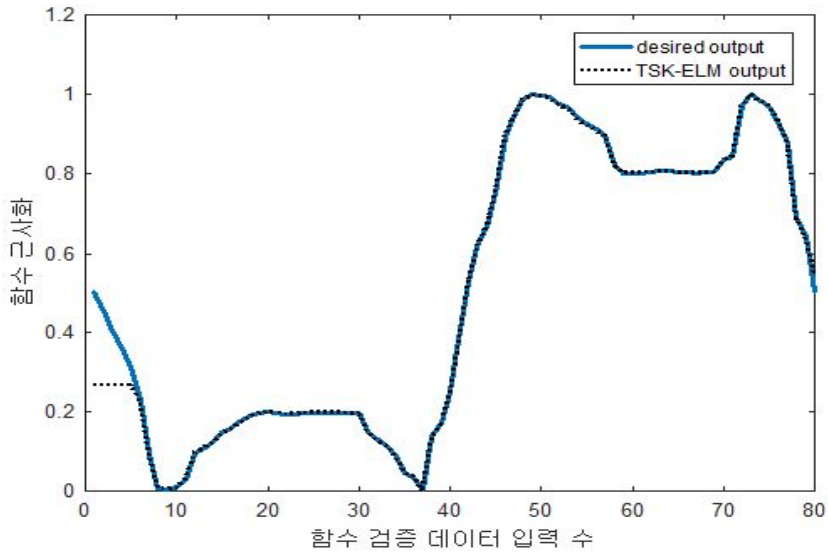
그림 4.3 정규화 과정을 하지 않은 함수 데이터 평균 제곱근 편차 값

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY



(a) 기존 ELM 예측기의 검증 출력 값

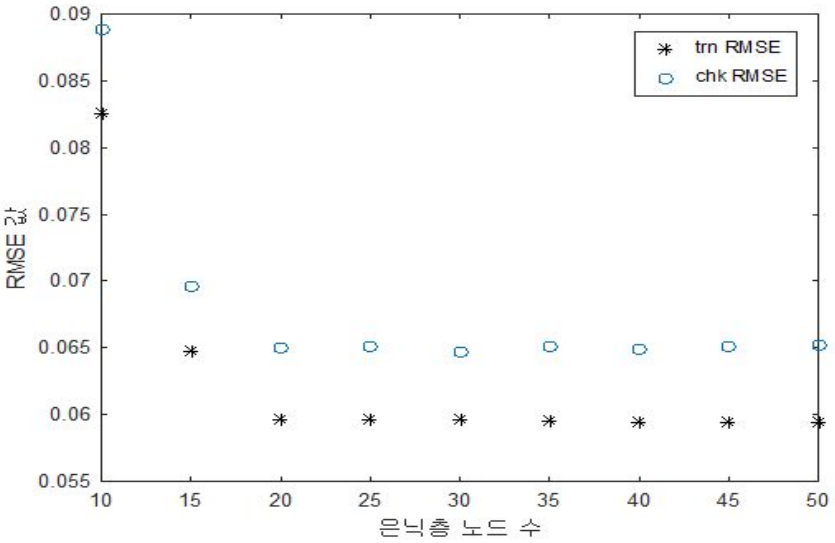


(b) TSK-ELM 예측기의 검증 출력 값

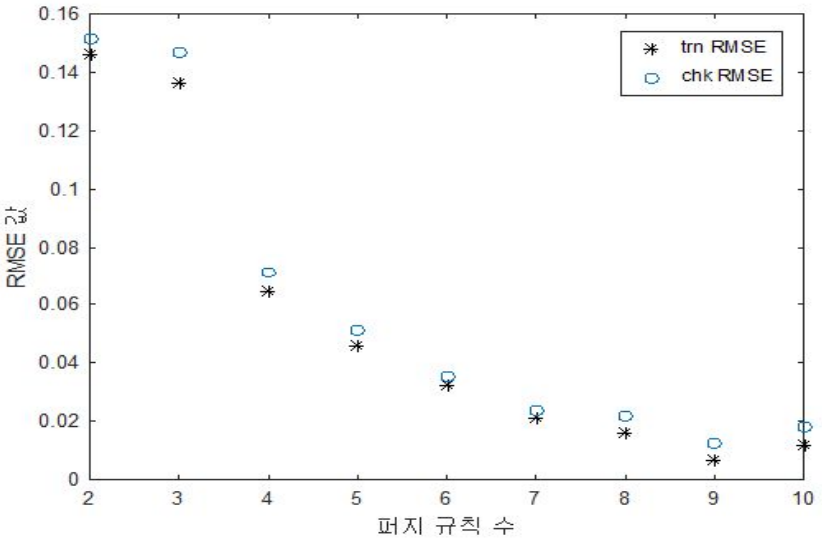
그림 4.4 정규화 과정을 한 함수 검증 데이터 출력 값

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY



(a) 기존 ELM 예측기의 평균 제곱근 편차 값



(b) TSK-ELM 예측기의 평균 제곱근 편차 값

그림 4.5 정규화 과정을 한 함수 데이터 평균 제곱근 편차 값

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

표 4-2 함수 데이터를 사용한 예측기 성능 결과

실험 방법	알고리즘 종류	은닉층 및 규칙 수*	학습 RMSE	검증 RMSE
비정규화	ELM[29]	50	0.044	0.070
	TSK-ELM	9*	0.010	0.011
정규화	ELM[29]	20	0.060	0.065
	TSK-ELM	9*	0.010	0.012

표 4-2에서 보이는 바와 같이, 데이터가 정규화 되지 않은 상태에서 TSK 기반의 ELM 예측기가 ELM 예측기보다 0.059 낮은 0.011의 평균 제곱근 편차 값을 가졌고, 데이터가 정규화 된 상태에서도 0.053 낮은 0.012의 평균 제곱근 편차 값을 가지는 것을 확인했다. 위 표에서 확인했듯이, TSK 기반의 ELM 예측기의 성능이도 좋은 것을 확인 할 수 있었다.

제2절 Boston Housing 데이터 예측 문제

Boston Housing 데이터는 Carnegie Mellon University에서 관리하는 StatLib 라이브러리에서 제공된다. 미국의 보스턴의 주택 가격 정보에 대한 데이터이로서, 자치 시 별 1인당 범죄율, 25.000평방피트를 초과하는 거주지역의 비율, 비소매상업지역이 점유하고 있는 토지 비율, 찰스강과의 근접성, 10ppm 당 농축된 일산화질소량, 주택 1 가구당 평균 방의 개수, 1940년대 이전에 건축된 주택의 비율, 5개 보스턴 직업센터까지의 접근성, 방사형 도로까지의 접근성 지수, 10.000달러 당 제산세율, 자치 시 별 흑인 비율, 모집단의 하위 계층의 비율, 자치 시 별 학생과 교사 비율 등의 정보를 통해 주택의 가격에 영향을 미치는 요소를 분석하고 예측하는 문제에 사용된다[45]. 데이터의 크기는 506×14로 입력 변수 13개, 출력 변수 1개로 구성되어진다. 그림 4.6는 Boston Housing 데이터의 출력을 보여준다.

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

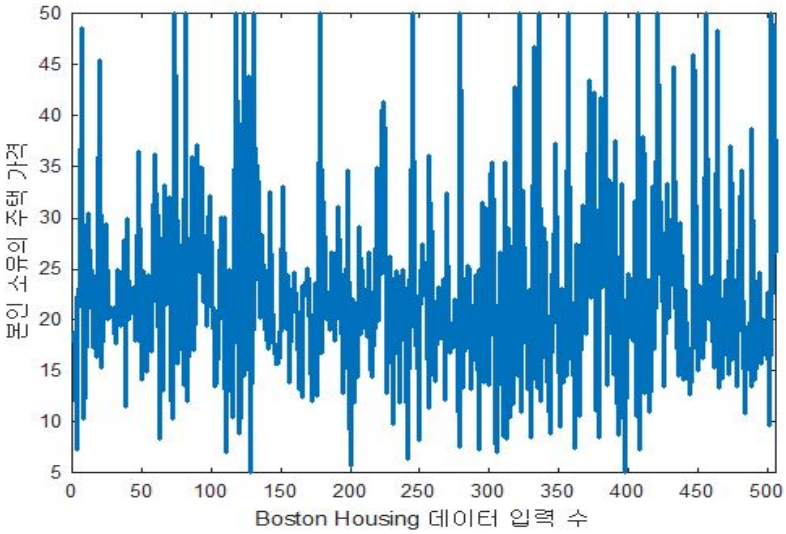
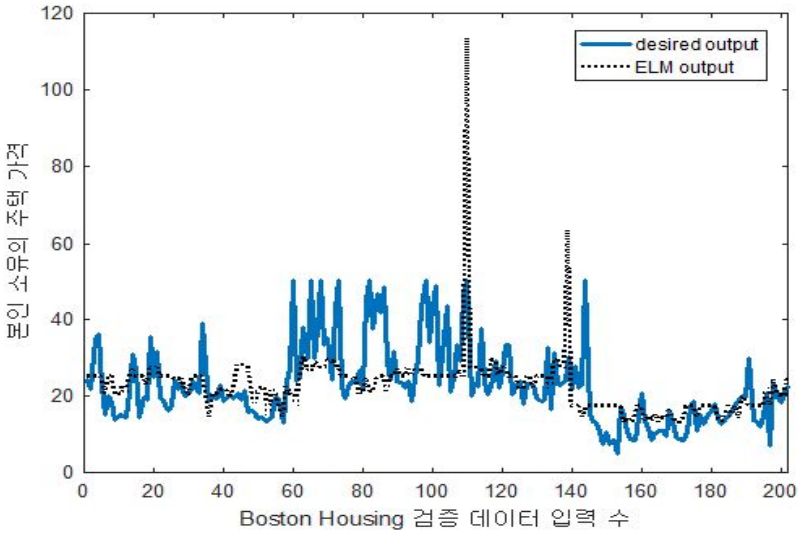


그림 4.6 Boston Housing 데이터 출력 값

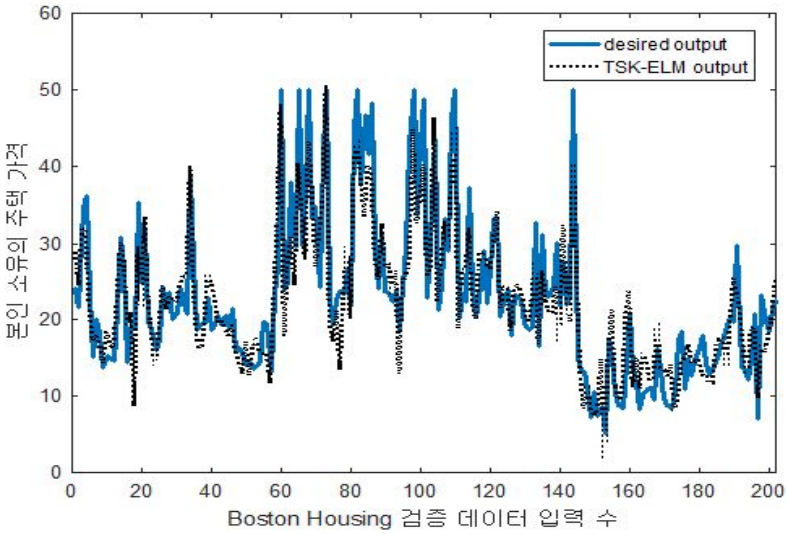
본 실험은 정규화 과정 여부에 따른 두 번의 실험을 진행하여 ELM 예측기와 TSK 기반의 ELM 예측기의 예측 성능을 비교했다. 그림 4.7은 정규화 과정을 하지 않은 Boston Housing 데이터에 대한 ELM 예측기와 TSK 기반의 ELM 예측기의 검증 출력을 나타내며, 그림 4.8은 정규화 과정을 하지 않은 데이터를 이용한 ELM 예측기와 TSK-ELM 예측기의 평균 제곱근 편차 값을 나타낸다. 그리고 그림 4.9는 정규화 과정을 한 Boston Housing 데이터에 대한 ELM 예측기와 TSK 기반의 ELM 예측기의 검증 출력을 나타내며, 그림 4.10은 정규화 과정을 한 데이터를 이용한 ELM 예측기와 TSK-ELM 예측기의 평균 제곱근 편차 값을 나타낸다. Boston Housing 데이터를 사용한 두 예측기의 성능 결과는 표 4-3과 같이 나왔다. 표 4-3에서는 두 예측기의 최적의 성능을 가졌을 때의 구조도 함께 표기하였다. ELM 예측기는 은닉층의 노드 수를 표기 하였고, TSK 기반의 ELM 예측기는 퍼지 규칙의 수를 표기하였다.

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY



(a) 기존 ELM 예측기의 검증 출력 값

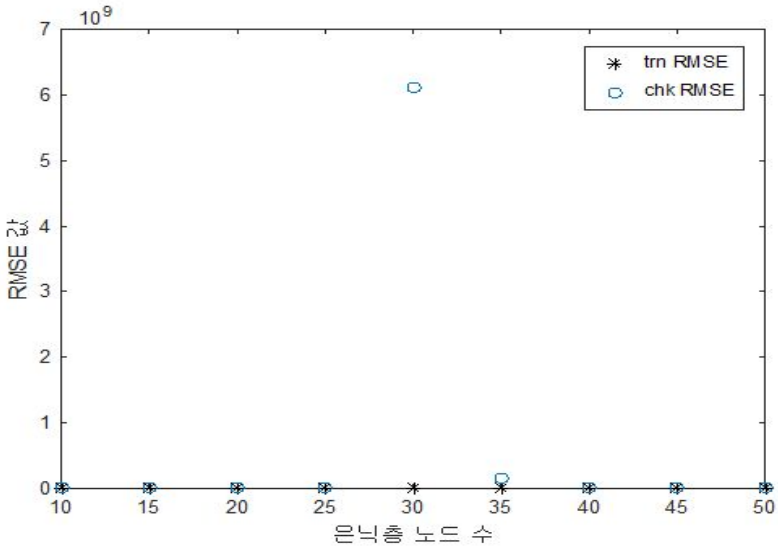


(b) TSK-ELM 예측기의 검증 출력 값

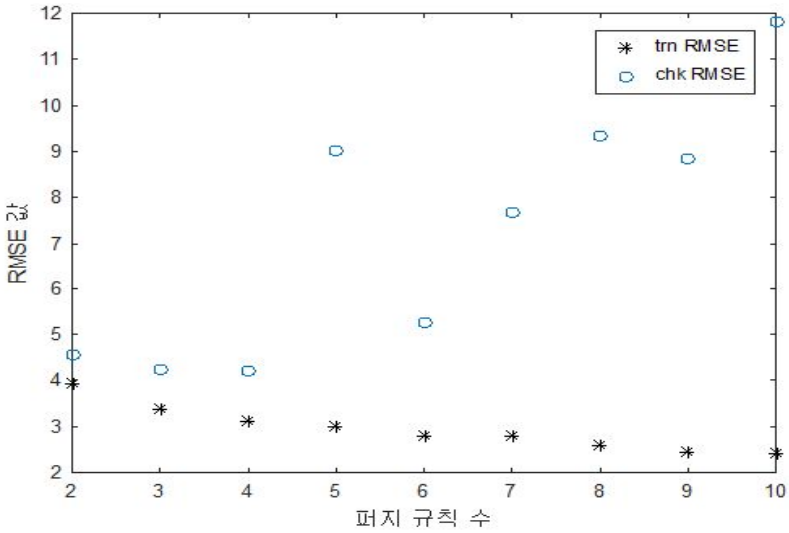
그림 4.7 정규화 과정을 하지 않은 Boston Housing 검증 데이터 출력 값

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY



(a) 기존 ELM 예측기의 평균 제곱근 편차 값

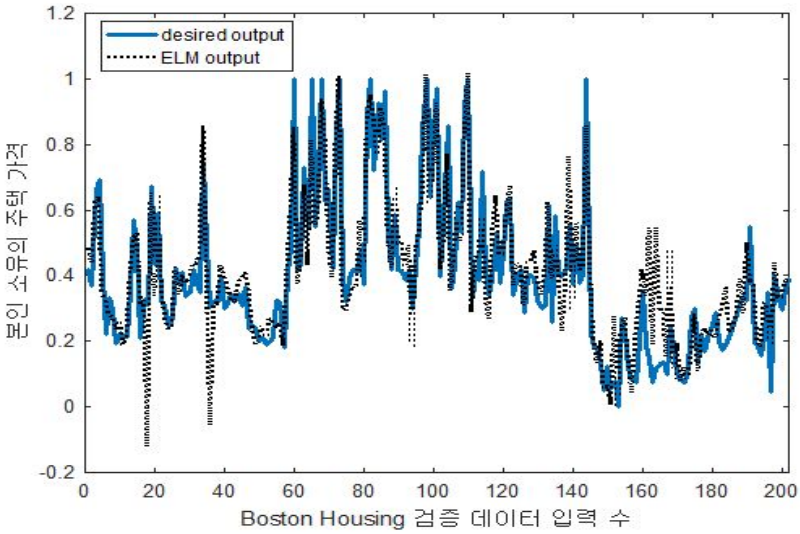


(b) TSK-ELM 예측기의 평균 제곱근 편차 값

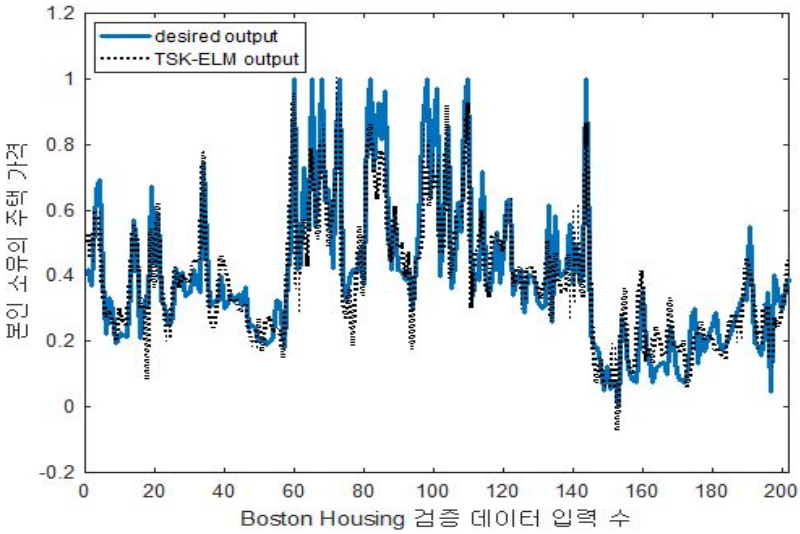
그림 4.8 정규화 과정을 하지 않은 Boston Housing 평균 제곱근 편차 값

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY



(a) 기존 ELM 예측기의 검증 출력 값

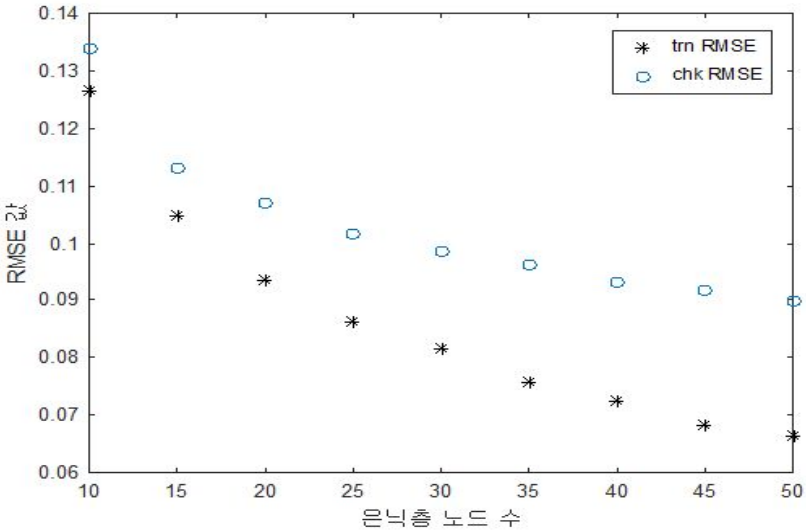


(b) TSK-ELM 예측기의 검증 출력 값

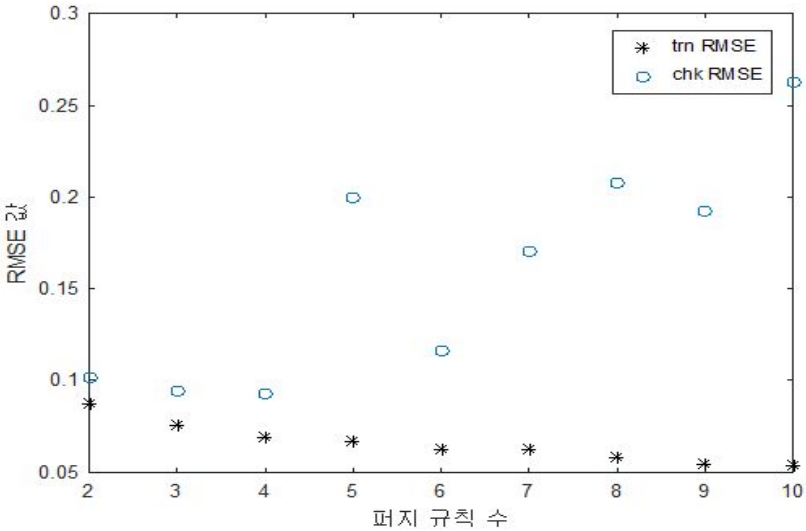
그림 4.9 정규화 과정을 한 Boston Housing 검증 데이터 출력 값

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY



(a) 기존 ELM 예측기의 평균 제곱근 편차 값



(b) TSK-ELM 예측기의 평균 제곱근 편차 값

그림 4.10 정규화 과정을 한 Boston Housing 데이터 평균 제곱근 편차 값

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

표 4-3 Boston Housing 데이터를 사용한 두 예측기 성능 결과

실험 방법	알고리즘 종류	은닉층 및 규칙 수*	학습 RMSE	검증 RMSE
비정규화	ELM[29]	20	7.747	8.240
	TSK-ELM	4*	3.112	4.191
정규화	ELM[29]	45	0.083	0.105
	TSK-ELM	4*	0.070	0.093

표 4-3에서 보이는 바와 같이, 데이터가 정규화 되지 않은 상태에서 TSK 기반의 ELM 예측기가 ELM 예측기보다 4.049 낮은 4.191의 평균 제곱근 편차 값을 가졌고, 데이터가 정규화 된 상태에서도 0.012 낮은 0.093의 평균 제곱근 편차 값을 가지는 것을 확인했다. 위 표에서 확인했듯이, TSK 기반의 ELM 예측기의 성능이도 좋은 것을 확인 할 수 있었다.

제3절 Auto-MPG 데이터 예측 문제

Auto-MPG 데이터도 Camegie Mellon University에서 관리하는 StatLib 라이브러리에서 제공되며, 1983년 American Statistical Association Exposition에서 처음 사용 되었다[45]. Auto-MPG 데이터는 자동차 연료 소비량(Miles Per Gallon)으로 서로 다른 종류의 자동차들로부터 얻어졌다. 데이터의 사이즈는 392×8로 입력 변수는 실린더의 수, 배기량, 마력, 무게, 가속력, 모델 년도로 총 6개이며, 출력 변수는 자동차 연료 소비량이다. 자동차 모델명이 문자열로 포함되어 있지만, 본 실험에서는 사용하지 않았다. 그림 4.11은 Auto-MPG 데이터의 출력 값을 보여준다.

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

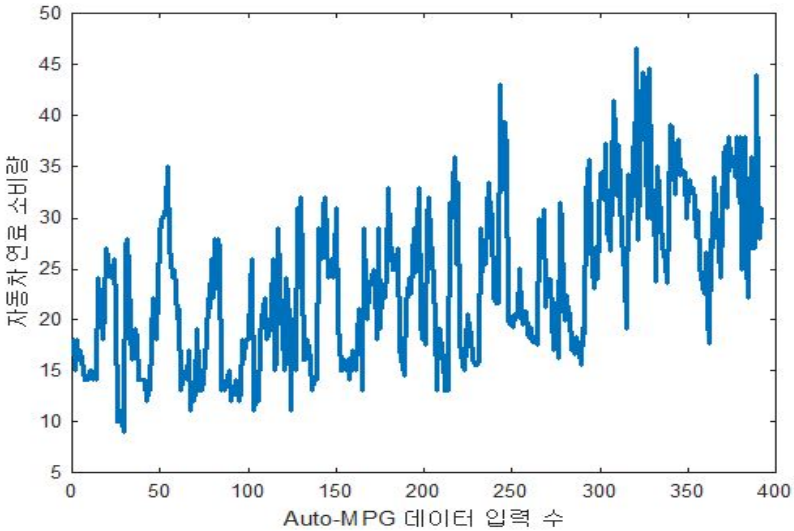
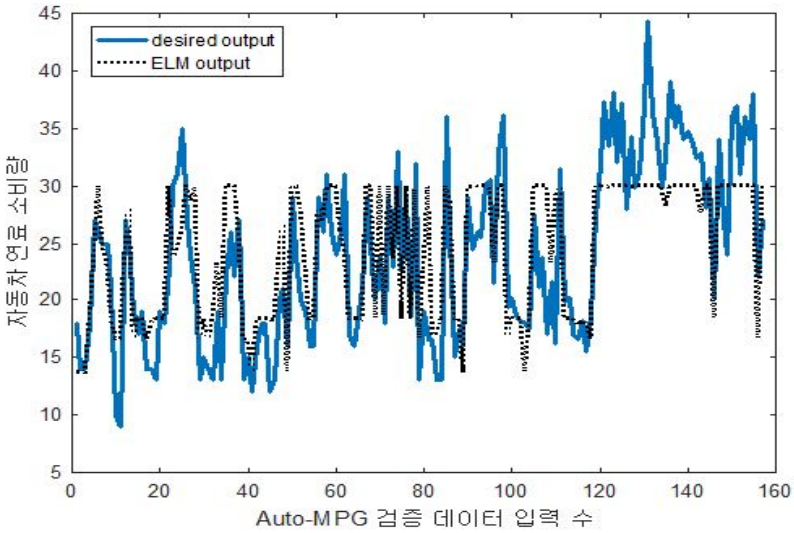


그림 4.11 Auto-MPG 데이터 출력 값

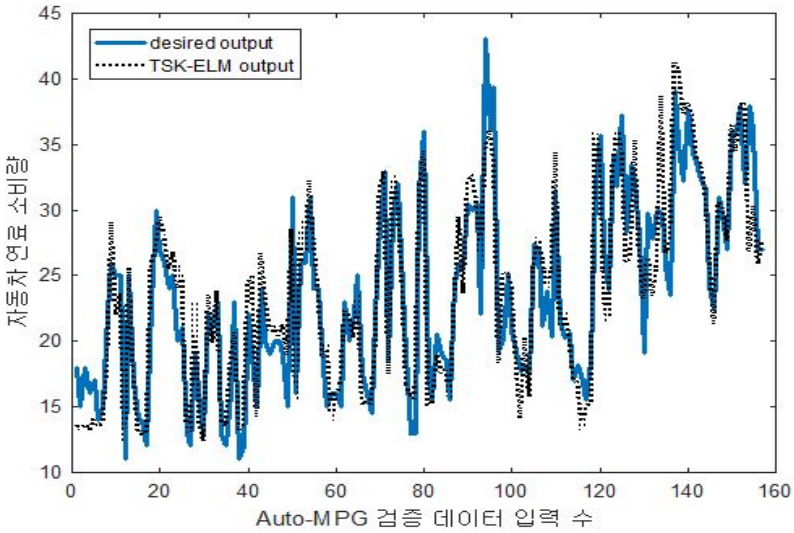
본 실험도 정규화 과정 여부에 따른 두 번의 실험을 진행하여 ELM 예측기와 TSK 기반의 ELM 예측기의 예측 성능을 비교했다. 그림 4.12은 정규화 과정을 하지 않은 Auto-MPG 데이터에 대한 ELM 예측기와 TSK 기반의 ELM 예측기의 검증 출력을 나타내며, 그림 4.13는 정규화 과정을 하지 않은 Auto-MPG 데이터를 이용한 ELM 예측기와 TSK-ELM 예측기의 평균 제곱근 편차 값을 나타낸다. 그리고 그림 4.14는 정규화 과정을 한 Auto-MPG 데이터에 대한 ELM 예측기와 TSK 기반의 ELM 예측기의 검증 출력을 나타내며, 그림 4.15은 정규화 과정을 한 Auto-MPG 데이터를 이용한 ELM 예측기와 TSK-ELM 예측기의 평균 제곱근 편차 값을 나타낸다. Auto-MPG 데이터를 사용한 두 예측기의 성능 결과는 표 4-3과 같이 나왔다. 표 4-4에서는 두 예측기의 최적의 성능을 가졌을 때의 구조도 함께 표기하였다. ELM 예측기는 은닉층의 노드 수를 표기 하였고, TSK 기반의 ELM 예측기는 퍼지 규칙의 수를 표기하였다.

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY



(a) 기존 ELM 예측기의 검증 출력 값

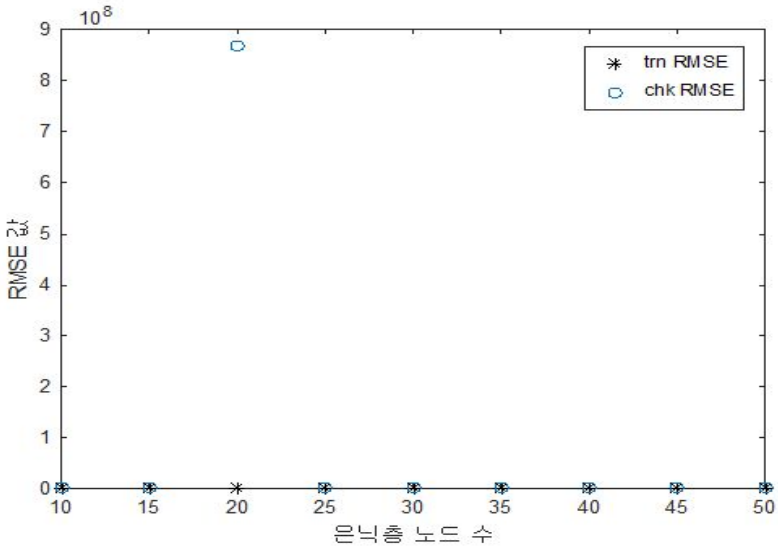


(b) TSK-ELM 예측기의 검증 출력 값

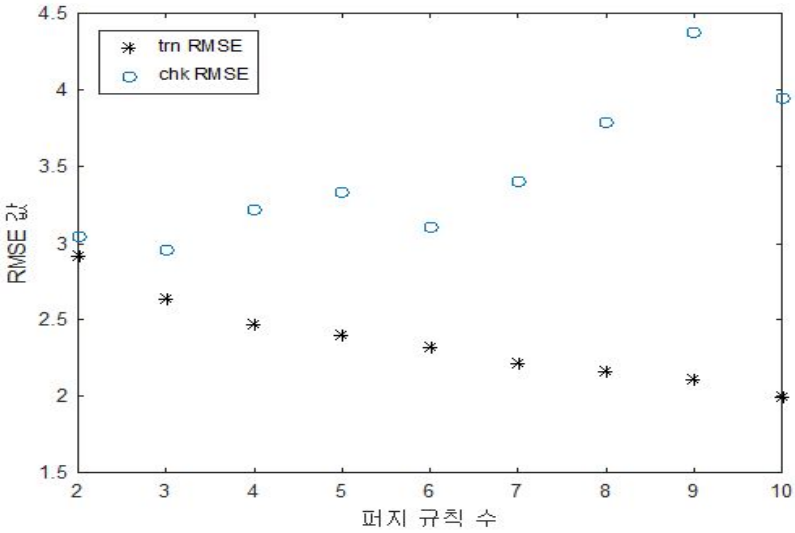
그림 4.12 정규화 과정을 하지 않은 Auto-MPG 검증 데이터 출력 값

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY



(a) 기존 ELM 예측기의 평균 제곱근 편차 값

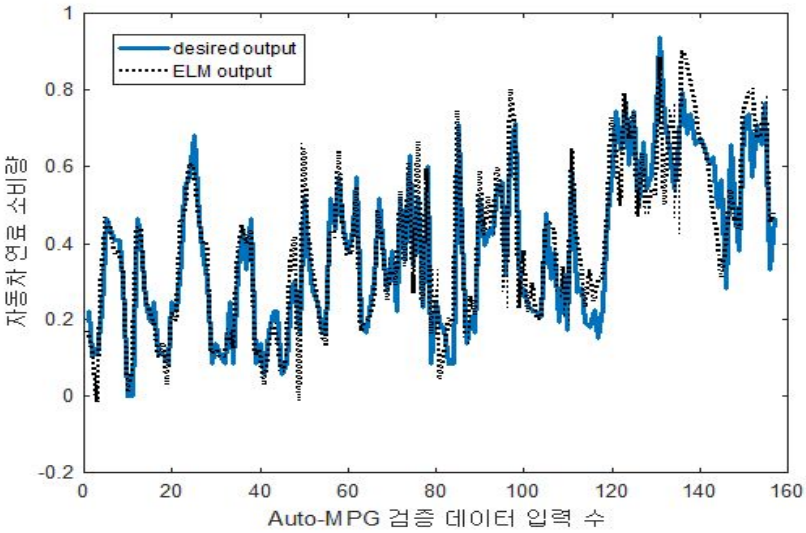


(b) TSK-ELM 예측기의 평균 제곱근 편차 값

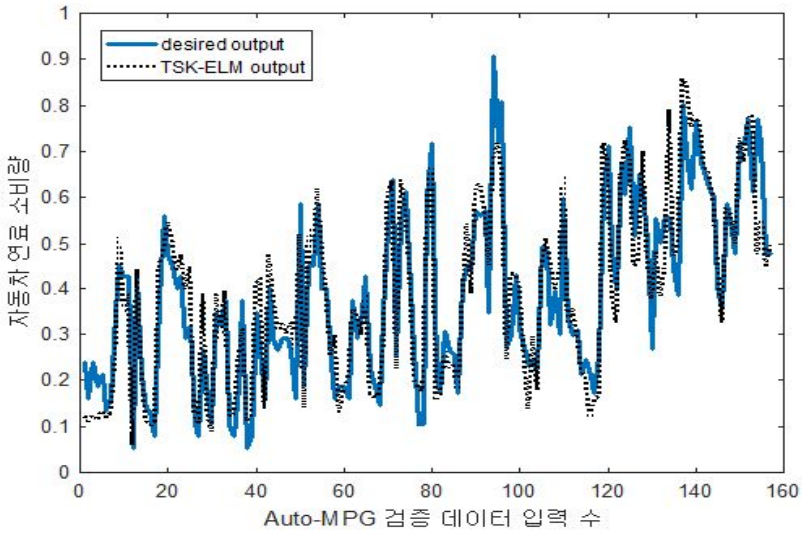
그림 4.13 정규화 과정을 하지 않은 Auto-MPG 데이터 평균 제곱근 편차 값

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY



(a) 기존 ELM 예측기의 검증 출력 값

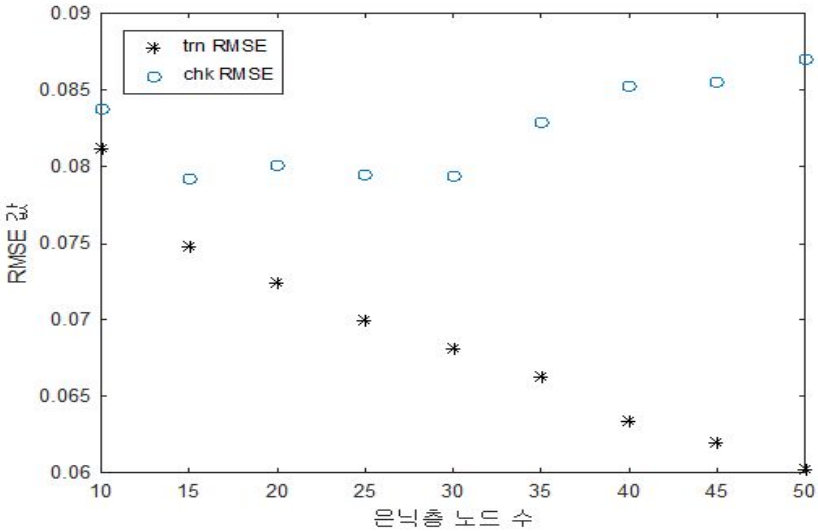


(b) TSK-ELM 예측기의 검증 출력 값

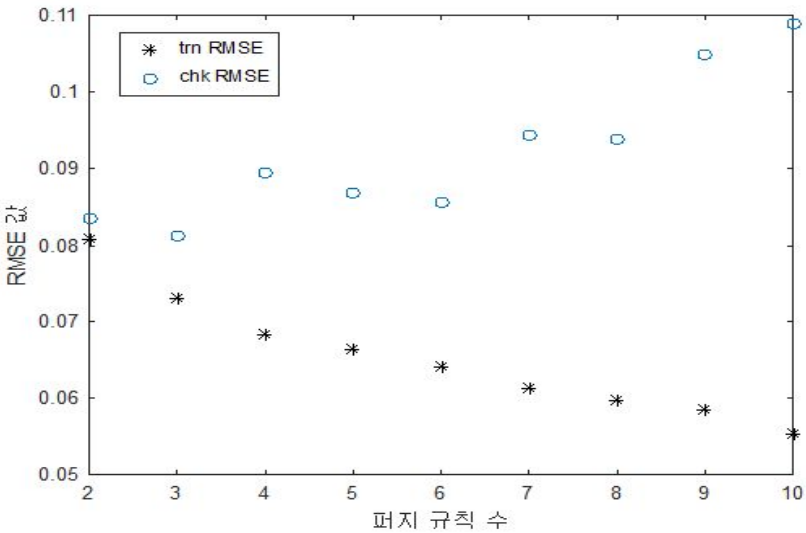
그림 4.14 정규화 과정을 한 Auto-MPG 검증 데이터 출력 값

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY



(a) 기존 ELM 예측기의 평균 제곱근 편차 값



(b) TSK-ELM 예측기의 평균 제곱근 편차 값

그림 4.15 정규화 과정을 한 Auto-MPG 데이터 평균 제곱근 편차 값

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

표 4-4 Auto-MPG 데이터를 사용한 예측기 성능 결과

실험 방법	알고리즘 종류	은닉층 및 규칙 수*	학습 RMSE	검증 RMSE
비정규화	ELM[29]	30	6.942	6.929
	TSK-ELM	3*	2.634	2.953
정규화	ELM[29]	20	0.191	0.211
	TSK-ELM	3*	0.073	0.081

표 4-4에서 보이는 바와 같이, 데이터가 정규화 되지 않은 상태에서 TSK 기반의 ELM 예측기가 ELM 예측기보다 3.976 낮은 2.953의 평균 제곱근 편차 값을 가졌고, 데이터가 정규화 된 상태에서도 0.13 낮은 0.081의 평균 제곱근 편차 값을 가지는 것을 확인했다. 표 4-5에서는 TSK 기반의 ELM 예측기에서 생성되는 전제부 파라미터를 나타내고, 표 4-6에서는 결론부 파라미터에 대해서 나타내고 있다.

표 4-5 TSK 기반의 ELM 예측기에서 생성되는 전제부 파라미터 값

	x_1 : 실린더 수		x_2 : 배기량		x_3 : 마력		x_4 : 무게		x_5 : 가속력		x_6 : 연식	
	A_1	A_{10}	B_1	B_{10}	C_1	C_{10}	D_1	D_{10}	E_1	E_{10}	F_1	F_{10}
c_i	5.44	5.45	199.62	190.27	108.53	103.22	3028	2940	15.35	15.55	75.83	76.19
a_i	2	2	0.025	0.025	0.053	0.053	0.002	0.002	0.602	0.602	0.833	0.833

표 4-5에서 c_i는 소속 함수의 중심을 나타내고, a_i는 소속 함수의 폭을 나타낸다.

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

표 4-6 TSK 기반의 ELM 예측기에서 생성되는 결론부 파라미터 값

규칙	p_i	q_i	r_i	s_i	t_i	o_i	v_i
1	-27.86	1.08	2.84	-0.06	-21.67	0.91	-229.66
2	-79.19	-0.15	1.12	0.46	-30.77	-8.21	-641.04
3	-38.78	4.10	3.67	-0.40	23.90	-73.00	6173
4	-10.30	-0.14	-0.01	0.00	-0.50	-10.94	1033
5	0.88	0.10	0.11	-0.01	-0.32	16.10	-1109
6	16.12	0.16	-0.15	0.00	2.71	2.86	-435.51
7	24.32	0.37	0.53	-0.03	0.94	0.96	-127.68
8	-4.93	0.26	0.16	-0.01	0.50	-19.69	1374
9	-30.37	-0.76	-2.50	0.20	-16.61	-17.30	1552
10	6.59	0.12	-0.02	-0.01	0.86	-17.51	1497

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

제5장 결론

최근 머신 러닝의 지도학습 방법의 하나인 ELM 모델은 Huang 등에 의하여 제안된 방법으로 여러 분야에서 많은 연구가 진행되고 있다. ELM 모델의 이론적인 연구로는 딥 러닝을 기반으로 하는 방법, 온라인 순차적 방법, 가중치 일반화 방법 등이 연구들이 있었다. ELM 분류기에 대한 연구로는 최소 클래스 분산 ELM을 이용한 동작 표현 분류 연구, 색 공간 기반 커널 ELM로 교통 표지판 인식, 딥 러닝 기반 ELM을 이용한 음성 감정 인식 등의 연구들이 있었다. 그리고 ELM 예측기에 대한 연구로는 단기 물 수요 예측에 대한 연구, 기상요소를 분석하고 일별 태양광 발전량을 예측하는 ELM 예측기 개발, 일별 최대 전력 수요 예측 알고리즘 개발, 광전자 신호를 예측하는 연구 등이 이루어졌다.

위 연구들의 결과들을 보면 ELM 분류기의 분류 성능은 대체적으로 잘 나오는 반면, ELM 예측기의 예측 성능은 일정하지 않은 것을 알 수 있다. 그 이유로는 데이터의 형태에 따라서 성능이 달랐고, 입력층과 은닉층 사이에 있는 가중치에서 의미 있는 값을 찾지 못하는 것과, 성능을 높이기 위해서 은닉층의 노드 수를 크게 늘려야 한다.

본 논문에서는 ELM 예측기의 문제점들을 보완 할 수 있는 TSK 퍼지 규칙과 랜덤 클러스터링 방법을 이용한 ELM 예측기를 제안했다. TSK 퍼지 규칙과 랜덤 클러스터링 방법을 이용한 ELM 예측기는 기존의 ELM 예측기에서 사용한 퍼지 클러스터링 방법이 아닌 랜덤하게 클러스터의 중심을 찾는 랜덤 클러스터링 방법을 사용하여, 적은 수의 의미 있는 퍼지 규칙 수를 생성 할 수 있었다. 제안한 ELM 예측기의 성능을 평가하기 위해서 함수 데이터, Boston Housing 데이터, Auto-MPG 데이터를 이용하여 기존의 ELM 예측기와의 예측성능을 비교하였다. 실험 결과 ELM 예측기의 최고의 성능을 나타낼 때, 은닉층의 수가 20~50개 사이로 일정하지 않고, 큰 폭으로 변화 하는 것을 확인 할 수 있었다. 반면에, TSK 기반의 ELM 예측기는 퍼지 규칙 수가 3~9개 사이로 ELM 예측기보다 적은 차이를 보이는 것을 확인했다. 그리고 평균 제곱근 편차의 값도 TSK 기반의 ELM 예측기의 성능이 좋은 것 확인할 수 있었다.

본 연구에서는 TSK 퍼지 규칙과 랜덤 클러스터링 방법을 이용하여 ELM 예측기의 성능을 향상 시켰다. 향후에는 은닉층과 출력층 사이에 위치하고 있는 활성 함

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

수를 새로운 활성 함수로 바꾼다면 더 좋은 결과를 얻을 수 있을 것으로 판단되며, 성능을 더욱 향상 시킬 수 있는 방법에 대해 지속적으로 연구할 계획이다.

dCollection @ chosun

조선대학교
CHOSUN UNIVERSITY

참고문헌

[1] 조성준, 강석호, “머신러닝(인공지능)의 산업 응용”, 대한산업공학회 ie 매거진, pp 34-38, 2016.

[2] 김항석, 신현정, “준지도 학습 및 신경망 알고리즘을 이용한 전기가격 예측”, 대한산업공학회지, Vol. 39, No. 1, pp 30-45, 2013.

[3] 박주영, 지승현, 성기훈, 허성만, 박경욱, “데이터 기반 활률론적 최적제어와 근사적 추론 기반 강화 학습 방법론에 관한 고찰”, 한국지능시스템학회 논문지, Vol. 25, No. 4, pp 319-326, 2015.

[4] Shifei Ding, Lili Guo, Yanlu Hou, “Extreme Learning Machine with Kernel Model based on Deep Learning”, Neural Computing and Applications, pp 1-10, 2016.

[5] Nan-yang Liang, Guang-bin Huang, P. Saratchandran, N. Sundararajan, “A Fast and Accurate Online Sequential Learning Algorithm for Feedforward Networks”, IEEE Transactions on Neural Networks, Vol. 17, pp 1411-1423, 2006.

[6] W. Zong, Guang-Bin Huang, Yiqiang Chen, “Weighted Extreme Learning Machine for Imbalance Learning”, Neurocomputing, Vol. 101, pp 229-242, 2013.

[7] Jiwen Cao, Zhiping Lin, Guang-Bin Huang, “Self-Adaptive Evolutionary Extreme Learning Machine”, Neural Processing Letters, Vol. 36, pp 285-305, 2012.

[8] Guang-bin Huang, Qin-Yu Zhu, Chee-Kheong Siew, “Extreme Learning Machine : Theory and Applications”, Neurocomputing, Vol. 70, pp 489-501, 2006.

[9] Niki Martinel, Christian Micheloni, Gian Luca Foresti, “The Evolution of Neural Learning Systems” IEEE Systems, Man, and Cybernetics Magazines, Vol. 1, pp 17-26, 2016.

[10] Le AN, Bir Bhanu, “Image Super-Resolution by Extreme Learning

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

Machine", 24th IEEE International Conference on Image Processing, pp 2209-2212, 2012.

[11] Guang-Bin Huang, Zuo Bai, Liyanaarachchi Lekamalage Chamara Kasun, "Local Receptive Fields Based Extreme Learning Machine", IEEE Computational Intelligence Magazine, Vol. 10, pp 18-29, 2015.

[12] Alexandros Iosifidis, Anastasios Tefas, Ioannis Pitas, "Minimum Class Variance Extreme Learning Machine for Human Action Recognition", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 23, pp 1968-1979, 2013.

[13] ZHige Xie, Kai Xu, Ligang Liu, Yueshan Xiong, "3D Shape Segmentation and Labeling via Extreme Learning Machine", Eurographics Symposium on Geometry Processing, Vol. 22, 2014.

[14] Yujin Zeng, Xin Xu, Dayong Shen, Yuqiang Fang, Zhipeng Xiao, "Traffic Sign Recognition Using Kernel Extreme Learning Machines With Deep Perceptual Features", IEEE Transactions on Intelligent Transportation Systems, pp 1-7.

[15] Kun Han, Dong Yu, Ivan Tashev, "Speech Emotion Recognition Using Deep Neural Network and Extreme Learning Machine", INTERSPEECH, 2014.

[16] 신강욱, 홍성택, "ELM(Extreme Learning Machine) 기반의 단기 물 수요예측 알고리즘", 대한전기학회 학술대회 논문집, pp 1728-1729, 2011.

[17] 이창성, 지평식, "ELM을 이용한 일별 태양관발전량 예측모델 개발", 대한전기학회논문지, Vol. 64, No. 3, pp 164-168, 2015.

[18] 지평식, 김상규, 임재윤, "ELM을 이용한 일별 최대 전력 수요 예측 알고리즘 개발", 대한전기학회논문지, Vol. 62, No. 4, pp 169-174, 2013.

[19] 지평식, 임재윤, "ELM을 이용한 특수일 최대 전력수요 예측 모델 개발", 대한전기학회논문지, Vol. 64, No. 2, pp 74-78, 2015.

[20] 최기선, 이동훈, 김선환, 이경우, 전명근, "정수장 운영효율 향상을 위한 ELM 기반 단기 물 수요 예측", 한국조명전기설비학회 논문지, Vol. 23, pp 108-116, 2009.

[21] Amita P. Mahuli, B. Niraniana Krupa, Koshy George, "Prediction of

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

Photoplethysmogram Signals Using Sequential ELM", International Joint Conference on Neural Networks(IJCNN), pp 1977-1984, 2016.

[22] Doreswamy, M Umme Salma, "BAT-ELM : A Bio Inspired Model for Prediction of Breast Cancer Data", International Jiont Conference on Applied and Theoretical Computing anf Communication Technology(iCATccT), pp 501-506, 2015.

[23] Xue Mo, Youqing Wang, Xiangwei Wu, "Hypoglycemia Prediction Using Extreme Learning Machine (ELM) and Regularized ELM", 25th Chinese Control and Decision Conference(CCDC), pp 4405-4409, 2013.

[24] Basant Yadav, Sudheer Ch, Shashi Mathur, Jan Adamowski, "Discharge Forecasting Using an Online Sequential Extreme Learning Machine (OS-ELM) Model : A Case Study in Neckar River, Germany", Measurement, Vol 92, pp 433-445.

[25] Xia Chen, Zhao Yang Dong, Ke Meng, Yan Xu, Kit Po Wong, H. W. Ngan, "Electricity Price Forecasting with Extreme Learning Machine and Bootstrapping", IEEE Transactions on Power Systems, Vol. 27, pp 2055-2062, 2012.

[26] Beatriz Saavedra-Moreno, Sancho Salcedo-Sanz, Leopoldo Carro-Calvo, Luis Prieto, "Very Fast Training Neural-Computation Techniques for Real Measure-Correlate-Predict Wind Operations in Wind Farms", Journal of Wind Engineering Industrial Aerodynamics, 2013.

[27] Hui-Xin Tian, Zhi-Zhong Mao, "An Ensemble ELM Based on modified AdaBoost.RT Algorithm for Predicting the Temperature of Molten Steel in Ladle Furnace", IEEE Transactions on Automation Science and Engineering, Vol. 7, pp 73-80, 2009.

[28] Qi Yu, Mark van Heeswijk, Yoan Miche, Rui Nian, Bo He, Eric Severin, Amaury Lendasse, "Ensemble Delta Test Extreme Learning Machine(DT-ELM) for Regression", Neurocomputing, Vol. 129, pp 153-158, 2014.

[29] Jianzhou Wang, Jianming Hu, "A Robust Combination Approach for Short-Term Wind Speed Forecasting and Analysis-Combination of the

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

ARIMA(Autoergressice Integrated Moving Average), ELM(Extreme Learning Machine), SVM(Support Vector Machine) and LSSVM(Least Square SVM) Forcasts Using a GPR(Gaussian Process Regression) Model”, Energy, Vol. 93, pp 41-56, 2015.

[30] Guang-Bin Huang, Qin-Yu Zhu, Chee-Kheong Siew, “Extreme Learning Machine : A New Learning Scheme of Feedforward Neural Networks”, IEEE International Joint Conference on Neural Networks, Vol. 2, pp 985-990, 2004.

[31] Zhan-Li Sun, Kin-Fan Au, Tsan-Ming Choi, “A Neuro-Fuzzy Inference System Through Integration of Fuzzy Logic and Extreme Learning Machines”, IEEE Transactions on Systems, Man, and Cybernetics, Part B(Cybernetics), Vol. 37, pp 1321-1331, 2007.

[32] Denis Serre, “Matrices : Theory and Applications”, Springer, 2002.

[33] Erik Cambria, Guang-Bin Huang, “Extreme Learning Machine[Trends & Controversies]”, IEEE Intelligent Systems, Vol. 28 pp 30-59, 2013.

[34] Zhihong Man, Kevin Lee, Dianhui Wang, Zhenwei Cao, Suiyang Khoo, “Robust Single-Hidden Layer Feedforward Network-Based Pattern Classifier”, IEEE Transactions on Neural Networks and Learning Systems, Vol. 23, pp 1974-1986, 2012.

[35] Zhang Dezheng, Li Hongtao, Aziguli, Fan Linlin, Yang Fengbo, “Predicting Strip Thickness of Hot Rolled Sheet Based on the Combination of Correlation Analysis and Extreme Learning Machine”, IEEE 10th Conference on Industrial Electronics and Applications(ICIEA), pp 477-482, 2015.

[36] Chen-Sen Ouyang, Tzu-Chin Kao, Yu-Yuan Cheng, Chih-Hung Wu, Chiung-Hui Tsai, Meng-Wei Wu, “An Improved Fuzzy Extreme Learning Machin for Classification and Regression”, International Conference on Cybernetics, Robotics and Control(CRC), pp 91-94, 2016.

[37] Shunji Uchimura, Yoshihiko Hamamoto, Shingo Tomita, “On the Effect of the Nonlinearity of the Sigmoid Function in Artificial Neural Network Classifiers”, IEEE International Conference on, Vol. 6, pp

Collection @ chosun

조선대학교
CHOSUN UNIVERSITY

281-284, 1995.

[38] Qingshan Liu, Jun Wang, “A One-Layer Recurrent Neural Network with a Discontinuous Hard-Limiting Activation Function for Quadratic Programming”, IEEE Transactions on Neural Networks, Vol. 19, pp 558-570, 2008.

[39] Institute of Neuroinformatics, https://www.ini.uzh.ch/people/rich.

[40] 한정수, 곽근창. “컨벌루션 신경회로망과 ReLU 함수 기반 ELM 분류기를 이용한 영상 분류”, 한국정보기술학회 논문지, Vol. 15, pp 15-23, 2017.

[41] Vinod Nair, Geoffrey Hinton, “Rectified Linear Units Improve Restricted Boltzmann Machines”, Processings of the 27th International Conference on Machine Learning(ICML-10), pp 807-814, 2010.

[42] Jyh-Shing Roger Jang, Chuen-Tsai Sun, Eiji Mizutani, “Neuro-Fuzzy and Soft Computing”, Prentice Hall, 1997.

[43] Tommhiro Takagi, Michio Sugeno, “Fuzzy Identification of Systems and Its Applications to Modeling and Control”, IEEE Transactions on Systems, Man, and Cybernetics, Vol. 15, pp 116-132, 1985.

[44] Michio Sugeno, GeunTaek Kang, “Structure Identification of Fuzzy Model”, Fuzzy Sets and Systems, Vol. 28, pp 15-33, 1988.

[45] Myung-Geum Chun, Keun-Chang Kwak, Jeong-Woong Ryu, “Application of ANFIS for Coagulant Dosing Process in a Water Purification Plant”, IEEE Fuzzy Systems Conference Processing, Vol. 3, pp 1743-1748, 1999.

[46] UCI Machine Learning Repository, https://archive.ics.uci.edu/ml/datasets

Collection @ chosun