

안정화된 딥 네트워크 구조를 위한 다항식 신경회로망의 연구

A Study on Polynomial Neural Networks for Stabilized Deep Networks Structure

전 필 한* · 김 은 후* · 오 성 권†

(Pil-Han Jeon · Eun-Hu Kim · Sung-Kwun Oh)

Abstract - In this study, the design methodology for alleviating the overfitting problem of Polynomial Neural Networks(PNN) is realized with the aid of two kinds techniques such as L_2 regularization and Sum of Squared Coefficients (SSC). The PNN is widely used as a kind of mathematical modeling methods such as the identification of linear system by input/output data and the regression analysis modeling method for prediction problem. PNN is an algorithm that obtains preferred network structure by generating consecutive layers as well as nodes by using a multivariate polynomial subexpression. It has much fewer nodes and more flexible adaptability than existing neural network algorithms. However, such algorithms lead to overfitting problems due to noise sensitivity as well as excessive training while generation of successive network layers. To alleviate such overfitting problem and also effectively design its ensuing deep network structure, two techniques are introduced. That is we use the two techniques of both SSC(Sum of Squared Coefficients) and L_2 regularization for consecutive generation of each layer's nodes as well as each layer in order to construct the deep PNN structure. The technique of L_2 regularization is used for the minimum coefficient estimation by adding penalty term to cost function. L_2 regularization is a kind of representative methods of reducing the influence of noise by flattening the solution space and also lessening coefficient size. The technique for the SSC is implemented for the minimization of Sum of Squared Coefficients of polynomial instead of using the square of errors. In the sequel, the overfitting problem of the deep PNN structure is stabilized by the proposed method. This study leads to the possibility of deep network structure design as well as big data processing and also the superiority of the network performance through experiments is shown.

Key Words : Polynomial neural networks, L_2 regularization, Least sward estimation, Sum of Squared Coefficients(SSC), Deep network structure

1. 서 론

근래에는 많은 연구자들이 시스템 모델링 방법을 다양한 방법으로 연구하고 있다[1]. 그중 기본이 되는 것이 입출력 데이터에 의한 선형계의 동정과 그 중 기본이 되는 것이 입출력 데이터에 의한 선형계의 동정과 예측문제에 관한 회귀분석 모델링 기법과 같은 수학적 모델링이 유용하게 사용되고 있다. 그러나 모델링을 하고자 하는 시스템이 복잡하고, 대규모 구조인 경우에는 함수식으로 모델의 표현이 한정되지 않기 때문에 회귀분석을 적용할 수 없는 경우가 있다. 일반적으로 복잡한 다변수계의 모델링은 많은 입출력 변수 중에서 모델을 구성하는 변수를 선택하여 모델의 구조를 결정하는데 어려움이 있으며, 추정방법의 계수가 많으면 추정에 필요한 데이터양이 방대해지는 문제점을 지니고 있다. 그러므로 시

스템에 대하여 파악하기 힘든 변수들에 대해선 이에 대한 경험적 지식을 갖춘 모델 설계자가 이런 변수들을 추측과 경험에 의존하여 시스템을 설계한다면, 좋은 예측 신뢰도를 가진 모델을 만들기란 거의 불가능하게 된다. 이러한 단점을 극복하기 위하여, 모델 설계시 설계자의 선입견을 첨가하지 않는 순수한 객관적 모델구축이 필요하게 되었다. A. G. Ivakhnenko는 이런 문제점을 해결하기 위한 방법으로 GMDH(Group Method of Data Handling)알고리즘을 제안했다[2]. GMDH 알고리즘은 입력변수의 선택과 입출력 데이터의 분할, 부분 표현식을 정의하여 시스템 방정식을 예측한다. 그러므로 입출력 관계나 모델의 함수 형이 특정하지 않은 비선형 관계의 경우에 2변수 2차식에 의한 부분 표현식을 계층적으로 조합하여 비선형 모델 추정식을 얻을 수 있다. 각 계층은 부분 표현식에 의한 회귀분석을 적용함으로써, 입출력 관계의 비선형 함수를 용이하게 얻을 수 있고, 매우 적은 데이터로 복잡하고 다변수인 비선형계의 동정과 예측이 가능했다. 그러나 자연계의 다양한 시스템을 예측하는데 있어 2변수로 국한을 하여 다양한 입력변수들끼리의 상호의존적인 면을 약하게 하였으며, 2차식으로는 다소 복잡한 시스템을 표현하기에는 역부족이다. 또한 간단히 표현될 수 있는 경우에는 간략화된 구조로 나타내는 것이 효과적이지만 모든

† Corresponding Author : School of Electrical and Electronic Engineering, the University of Suwon, Korea.
E-mail : ohsk@suwon.ac.kr

* School of Electrical and Electronic Engineering, the University of Suwon, Korea.

Received : January 19, 2017; Accepted : September 26, 2017

경우를 2차식으로 표현하므로 시스템에 맞는 적응력이 떨어지는 것이 단점이다. 본 논문에서는 최적의 모델설계를 위하여 2변수 2차식 GMDH 알고리즘의 한계를 극복하고 기존의 신경회로망 구조와 같이 고정되지 않고 새로운 구조를 생성하는 네트워크 구조를 가지는 PNN(Polynomial Neural Networks) 알고리즘을 이용하였다[3][4]. 2변수 2차식만 으로 고정되어 한계가 드러난 GMDH 알고리즘을 다변수 에 의해 변수의 개수를 확장하고, 2차식은 1차식(linear) 2차(quadratic), 3차(modified quadratic)등으로 차수 또한 확장한다. 뿐만 아니라 네트워크 각 세대의 입력변수 수 및 노드의 다항식 차수가 서로 다른 구조설계를 통하여 주어진 시스템 특성에 맞는 구조로 적응능력을 향상 시킬 수 있다. 하지만 PNN역시 확률 모델 및 지능형 모델의 경우 모델의 구조 및 데이터의 특성에 따라 과적합 문제가 발생한다. 과적합은 모델을 설계할 때 학습 데이터에 특징을 과도하게 학습한 나머지 예측 및 분류 율이 떨어지는 현상을 말한다. 이러한 문제를 입력변수간의 다중공선성(Multi-collinearity)이 발생함에 따른 것으로 보이며, 다항식 신경회로망에 L_2 regularization을 적용하였다[6-8]. 또한 이와 같이 Overfitting문제가 다중공선성 때문에 파라미터의 편차가 커지는 문제로 보고 각 층의 노드를 선택하는 과정에서 목적함수를 바꿔서 노드 선택에 변화를 주었다. 위와 같은 방법을 통해 다항식 신경회로망의 Overfitting 문제를 개선하였다.

- 따라서 본 논문에서 제시하는 핵심적인 내용은 다음과 같다.
- a) L_2 -norm regularization의 도입으로 계수의 추정에 Penalty항을 추가하여 크기가 작은 계수를 추정함으로써 Overfitting을 감소시키며,
 - b) 계수의 제곱 합(Sum of Squared Coefficients :SSC)를 도입하여 각 층의 Node 계수들의 합이 작은 노드들의 그룹을 선택함으로써 Overfitting을 감소시키며,
 - c) a)와 b)의 두 방법을 사용한 PNN의 Layer의 확장을 이용하여, Overfitting을 감소를 통한 PNN의 딥 학습에 의한 딥 네트워크 구조생성을 통해 빅데이터의 효율적인 처리 가능성을 제시하였다.

본 논문의 1장에서는 서론으로 본 논문에서 제시 하는 내용에 대하여 설명하였고, 본 논문 2장에서는 PNN알고리즘에 대하여 설명한다. PNN의 구조를 설명하고, PNN을 구성하는 PN(Polynomial Neuron)에 대하여 설명하고, 최소자승법에 대하여 설명한다. 3장에서는 연결가중치에 편차를 줄이기 위해서 사용한 L_2 regularization을 설명한다. 이 방법으로 능형 회귀(ridge regression)모형에서 사용하는 패널티 항을 제안된 패턴 분류기의 비용함수에 적용하여 견결가중치의 편차를 줄이는 동시에 실험 데이터의 일반화 성능을 향상시키는 방법과 PNN의 목적함수를 변형하여 새로 적용한 방법론에 관하여 설명한다. 그리고 4장에서는 Machine Learning 데이터인 가스터빈 발전소의 NOx 배출공정 데이터와 Automobile Miles per Gallon(MPG) 데이터 그리고 Boston Housing Data(BHD)를 이용하여 정규화를 적용한 다항식 신경회로망의 성능을 평가한다. 마지막으로 5장에서는 결론에 대하여 설명한다.

2. PNN 알고리즘과 구조

2장에서는 다항식 신경 회로망(Polynomial Neural Network : PNN)에 대해서 설명한다. 다항식 신경회로망은 GMDH 방법을 기반으로 하고 있으며, 연속되는 층과 노드를 생성하고 다변수 다차식 구조의 확장된 부분 표현 식을 이용하는 네트워크이다.

각 노드는 그림 1에서 보듯이, 각 입력이 들어오면 입력이 들어간 노드의 출력을 추정하기 위한 부분 표현 식을 선택하게 된다. 부분 표현식은 표 1에서의 3가지 중 하나의 다항식 구조를 선택하게 되며 노드의 출력은 다음 노드의 입력이 된다.

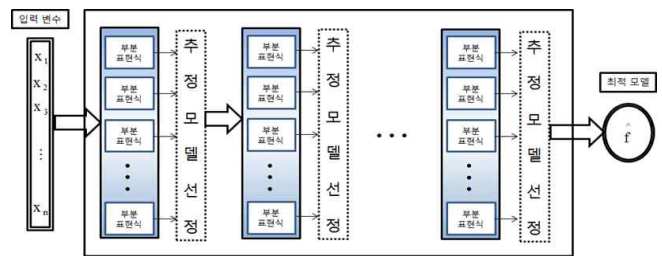


그림 1 PNN의 전체 설계 흐름도

Fig. 1 Overall design flowchart of PNN structure

2.1 다항식 뉴런(Polynomial Neuron : PN)

다항식 뉴런은 그림 2와 같이 구성되어 있으며, 다항식 신경회로망의 가장 기본이 되는 뉴런으로 선택된 입력변수의 수만큼 입력이 들어오면 표1에서 나타나는 3가지의 형태중 하나의 다항식 구조를 선택하여 다항식을 이루는 구조를 가진다[5]. 각 PN들은 하나의 출력을 가지고 있으며 다음 층의 입력이 된다.

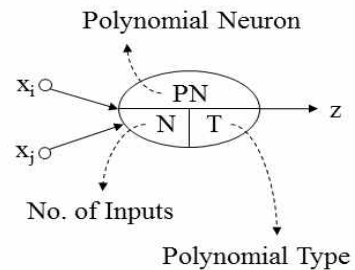


그림 2 Polynomial Neuron(PN)의 구조

Fig. 2 Structure of Polynomial Neuron

그림 2는 기본적인 PN의 구조로 2개의 입력을 가지고 있으며, 이와 같이 형성된 여러 개의 노드들 중에서 성능지수를 만족하는 노드들이 선택되어 다음단의 입력으로 사용된다.

표 1은 2개의 입력을 가지는 PN의 다항식 구조로 각각의 뉴런들은 다음과 같은 부분 표현식을 가지게 된다. 이러한 구조를 가진 뉴런들은 그림 1에서와 같이 각각의 층에 높은 성능지수를 가지는 PN들의 선택에 의해서 실행되며 최적의 성능을 보일 때

까지 층을 증가해가며 생성되며, 이는 최적의 다항식 신경회로망을 구축하기 위한 하나의 방법론에 해당한다.

2.2 PNN알고리즘

표 1 다항식 구조

Table 1 Types of Polynomial Neurons

| Type | Type | Polynomial Neuron |
|--------|--------------------|---|
| Type 1 | Linear | $y = a_0 + a_1x_1 + a_2x_2$ |
| Type 2 | Quadratic | $y = a_0 + a_1x_1 + a_2x_2 + a_3x_1x_2 + a_4x_1^2 + a_5x_2^2$ |
| Type 3 | Modified Quadratic | $y = a_0 + a_1x_1 + a_2x_2 + a_3x_1x_2$ |

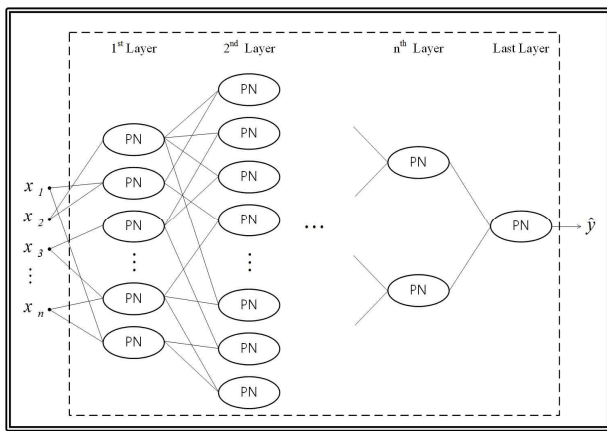


그림 3 Polynomial Neural Networks(PNN)의 전체구조

Fig. 3 Overall scheme of PNN

GMDH 알고리즘에 기초한 PNN 알고리즘은 고계다항식과 다변수 입력에 의하여 차수 및 입력변수의 선택과 알고리즘의 최종 종료판정 기준에 의하여, 파라미터 동정에 의한 모델을 선정하게 되고, 계층적 모델추정에 의하여 최종적으로 최적의 모델추정식을 얻는다.

식 (1)은 PN에 들어가는 입력으로 2입력 2차식의 방정식이 주어졌을 때의 일반적인 구조를 나타낸 것이다.

$$z_k = A + Bx_i + Cx_j + Dx_i^2 + Ex_j^2 + Fx_ix_j \quad (1)$$

식 (1)과 같은 다항식들이 그림1과 같이 각각의 PN들과 결합되어 최종적으로 고차의 다항식을 얻을 수 있게 된다. 최종 출력 식은 다음과 같은 형태를 가지게 된다.

$$\hat{f} = A + \sum_{i=1}^n Bx_i + \sum_{i=1}^n \sum_{j=1}^n Cx_ix_j + \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n Dx_ix_jx_k + \dots \quad (2)$$

여기서 A, B, C, D,... 은 계수이다. 최종 출력식 \hat{f} 를 얻기 위해, 각 뉴런의 부분 표현 식을 구성한다. 그리고 선형회귀분석방법중 하나인 최소 사승법을 통하여 부분표현식의 계수를 상정하

고, 중간변수 z_k 를 구한다. 중간변수에 의한 입출력 데이터를 구성하여, 다음 계층에서도 이와 같은 형태의 연산을 반복하여 알고리즘의 종료 판정을 만족할 때 까지 반복한다. 알고리즘이 종료할 때 까지 선택된 부분 표현식 들의 조합을 통하여 최종모델 y 를 얻는다. PNN알고리즘의 순서는 다음과 같다.

[단계 1] 입력변수 설정

출력변수 y 에 관련된 n 개의 입력변수를 설정한다. 필요하면 데이터를 정규화 또는 차원축소 같은 과정을 가진다. 입력변수는 x_1, x_2, \dots, x_n 라고 정의한다.

[단계 2] 데이터 분할

N 개의 입출력 데이터 $y = x_1, x_2, \dots, x_n, i = 1, 2, \dots, N$ 을 학습용 데이터와 평가용 데이터로 분할한다. 학습용 데이터는 모델을 동정하기 위해 사용하는 데이터이고, 평가용 데이터는 모델을 평가하기 위해 사용된다. 학습용 데이터와 평가용 데이터의 개수를 각각 N_{tr} 개와 N_{te} 개로 한다. 여기서, $N = N_{tr} + N_{te}$ 이다.

[단계 3] PNN 구조 선택

PNN 구조는 1층 이후 몇 개의 층이 연속 되면서 각층의 부분표현식을 구성하는 방법에 따라 모델의 구조가 정해지는 특성 때문에 다양한 구조 생성을 가능하게 되는 유연성을 가진다. PNN의 1층과 2층 이상에서 각각 서로 동일 입력변수 수를 가지는 경우와 서로 다른 입력변수 수를 가지는 경우의 두 구조로 분류되며 이를 기본 PNN 구조, 변형된 PNN 구조라 한다.

(a) 기본 PNN 구조 - PN의 입력변수 수가 동일

Case 1. 각 층이 동일한 차수를 가지는 구조.

Case 2. 각층의 PN의 차수가 서로 다른 차수 혹은 변형된 형태의 차수를 가지는 구조.

(b) 변형된 PNN 구조 -PN의 입력변수 수가 1층과 2층 이상에서 서로 다름

Case 1. 각 층이 동일한 차수를 가지는 구조.

Case 2. 각층의 PN의 차수가 서로 다른 차수 혹은 변형된 형태의 차수를 가지는 구조.

[단계 4] PN의 차수 및 입력변수 수 선택

n 개의 입력변수 x_1, x_2, \dots, x_n 중 PN의 입력변수를 선택한다. 입력 데이터 수와 PN의 입력변수의 개수에 따라 각 층의 PN의 개수가 결정된다. r 개의 변수 선택시 입력변수에 따른 PN의 개수는 $k = \frac{n!}{(n-r)!r!}$ 가 되며 n 은 전체 입력변수의 수이고, r 은 선택된 입력변수의 수이다. PN의 부분 표현식 을 2변수 2차식이라 하면, n 개의 입력변수 중에서 두 개의 입력변수 x_i, x_j 를 선택하였을 때 다음과 같은 2변수 2차식을 구성하게 된다.

$$z_k = a_0 + a_1x_i + a_2x_j + a_3x_ix_j + a_4x_i^2 + a_5x_j^2 \quad (3)$$

$$(k = 1, 2, \dots, n(n-1)/2)$$

여기서, a_0, a_1, \dots, a_5 는 계수를 나타내며, z_k 는 k 번째 PN의 출력 값이 된다.

[단계 5] 매개변수 추정

선형 회귀분석에 의해 학습 데이터를 사용하여, 식(4)을 최소화 하는 최적의 계수 a_0, a_1, \dots, a_n 을 구한다.

$$\Pi = \frac{1}{N} \sum_{i=1}^N (y_i - z_k)^2 \quad (4)$$

PNN 구조의 계수들은 파라미터로서 입출력 데이터가 주어졌을 때 최소자승법에 의해 최적 매개변수 추정을 행한다. 최소자승법에 의한 부분 표현식의 계수 동정은 식 (5)에 의해 구해진다.[12] 각 층에서,

$$Y = XA$$

여기서, $A = (X^T X)^{-1} X^T Y$ (5)

$$X = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1i} \\ x_{21} & x_{22} & \dots & x_{2i} \\ \vdots & \vdots & \dots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{ni} \end{bmatrix}, Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, A = \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{bmatrix}$$

[단계 6] 평가 후 노드의 선택

계수 a_0, a_1, \dots, a_n 를 추정한 식 (5)의 부분 표현 식에 평가용 데이터를 대입하고, 평가용 데이터에 대한 성능 지수를 구한다. 본 논문에서 사용한 노드의 선택 방법은 다음과 같이 두 가지 방법으로 구분할 수 있다.

(a) 방법 1

연어진 $\frac{n!}{(n-r)!r!}$ 개의 $\Pi_1, \Pi_2, \dots, \Pi_{n!/(n-r)!r!}$ 에 대하여, 값이 작은 순으로 나란히 정렬하고 이들 중 값이 작은 순으로 다시 ω 개를 선택하여, $\Pi_1, \Pi_2, \dots, \Pi_{n!/(n-r)!r!}$ 개로 재배열 하고, ω 개를 제외한 나머지 값들을 버린다.

$$\frac{n!}{(n-r)!r!} < W \text{ 일 경우 다음세대 노드의 개수는 } \frac{n!}{(n-r)!r!} \text{ 가 됨}$$

$$\frac{n!}{(n-r)!r!} \geq W \text{ 일 경우 다음세대 노드의 개수는 } W \text{ 가 됨}$$

(b) 방법 2

각 노드에서 추정한 계수 a_0, a_1, \dots, a_n 의 제곱 합을 구하여, $\frac{n!}{(n-r)!r!}$ 개의 계수의 제곱 합(Sum of Squared Coefficient: SSC) 값이 작은 순으로 나란히 정렬하고 이들 중 값이 작은 순으로 ω 개를 선택하여, $\frac{n!}{(n-r)!r!}$ 개로 재배열 하고, ω 개를 제외한 나머지 값들을 버린다.

$$SSC(\text{Sum of Squared Coefficient}) = \sum_{p=1}^n a_p^2 \quad (6)$$

$$\frac{n!}{(n-r)!r!} < W \text{ 일 경우 다음세대 노드의 개수는 } \frac{n!}{(n-r)!r!} \text{ 가 됨}$$

$$\frac{n!}{(n-r)!r!} \geq W \text{ 일 경우 다음세대 노드의 개수는 } W \text{ 가 됨}$$

본 논문의 경우 기존의 방법 1과 새로운 방법 2 두 가지 방법을 사용하여 모델을 추정해 간다. 방법 1의 경우 PNN의 층이 증가함에 따라 모델의 일반화 성능이 떨어지게 되어 overfitting이 발생한다. 이러한 문제를 입력변수간의 다중 공선성을 원인으로 보고 노드의 계수의 값이 커지게 되어 안정적이지 못하고 변화가 민감하게 반응하게 . 방법 2에서는 다중공선성의 영향을 적게 받는 노드를 찾기 위하여 각 노드의 계수의 제곱 합을 구하여 값이 작은 노드를 다음 세대의 입력으로 사용하였다. 각 세대의 노드의 개수는 최초에는 $\frac{n!}{(n-r)!r!}$ 로 정의하였고, 2세대에서 최종 출력세대까지는 각 노드의 개수를 일정개수 W 로 정의하였다. 그러므로 모델링 작업에서 오는 많은 계산 량과 과대한 연산 시간을 방지하였다.

[단계 7] 종료판정

알고리즘을 종료하는 방법에는 다음의 2가지 방법이 사용된다.

(a) 방법 1

[단계 6]에서 얻어진 PI가 다음의 부등식을 만족하는 경우에 알고리즘을 종료한다.

$$\Pi \geq PI \text{ (PI는 전세대의 PI를 의미한다)} \quad (7)$$

(a) 방법 2

종료 판정을 세대로 결정하여 종료하는 경우로 방법 1의 경우에서 현재의 오차가 앞 세대의 오차보다 크다고 하여 종료를 하면, 실제적으로는 종료 된 이후 세대에서의 좋은 결과 값을 가진 것들을 무시하게 된다. 또한 너무 많은 세대를 모델링 하는 경우에는 결과 값의 큰 차이를 보지 못하며, 또한 많은 실행 시간과 계산량을 요구하게 된다. 따라서 이러한 약점을 보완하기 위하여 본 논문에서는 방법 2의 경우를 택하여 5세대에서 7세대 까지 알고리즘을 실행한 후 종료하는 방법을 사용한다.

[단계 8] 새로운 입력변수 선택

식 (7)을 만족하지 않을 경우, $x_{1i} = z_{1i}, x_{2i} = z_{2i}, \dots, x_{wi} = z_{wi}$ 가 되어 새로운 입출력 데이터를 구성하고, 단계 4부터 단계 8까지를 반복한다.

알고리즘에 종료할 경우, PI를 얻는 부분표현식인 식(3)에 전 세대에서의 중간변수를 대입하고, 동일한 조작을 되풀이하여 최종 추정 모델 \hat{f} 를 얻는다.

3. L₂ 정규화 방법

확률 모델 및 지능형 모델의 경우 모델의 구조 및 데이터의 특성에 따라 과적합(overfitting) 문제가 발생한다. 과적합은 모델을 설계할 때 학습 데이터에 특징을 과도하게 학습한 나머지 예측 및 분류율이 떨어지는 현상을 말한다. 일반적으로 과적합이 발생하는 요인은 크게 두 가지로 구분할 수 있다. 첫 번째로 데이터의 분포와 관련이 있다. 학습 데이터도 어떤 모집단의 부분

집합으로 생각 할 수 있으며 데이터의 분포가 밀집되거나 어느 한쪽에 치우쳐져 있다면 전역학습이 불가능하다. 따라서 모집단에 가까운 학습 데이터를 구성할수록 과적합을 줄일 수 있지만 데이터를 수집하고 분석 하는데 있어서 많은 시간과 비용을 투자해야하는 어려움이 있다. 두 번째는 모델의 연결가중치의 편차가 클수록 과적합이 발생 할 가능성이 높다. 이러한 원인중 하나로 입력변수간의 다중공선성 이라 할수 있는데 다중공선성은 입력변수간의 연관성이 강하여 생기는 입력변수간의 비직교적 상태를 말한다. 다중 공선성이 발생할 경우 그 변수들을 처리하지 않고 연결가중치를 구하게 되면 연결가중치의 편차가 커져서 신뢰할수 없게 된다. 연결가중치는 데이터의 작은 변화나 회귀 모형식에서 예측변수를 추가하거나 제라함에 따라 상당히 민감하게 움직이며, 편차가 클수록 예측에 양좋은 영향을 주게 된다. 본 연구에서는 연결가중치의 편차를 줄이기 위해 L_2 정규화를 사용하였으며 이 방법은 능형 회귀(ridge regression)모형에서 사용하는 페널티항을 제안된 PNN의 성능평가에 적용하여 연결가중치의 편차를 줄이는 동시에 실험 데이터의 일반화 성능을 향상시켰다.

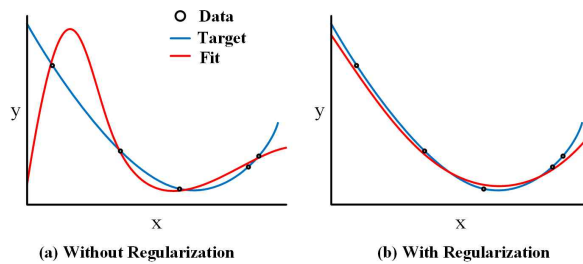


그림 4 Regularization 의 효과
Fig. 4 Effect of regularization

선형 회귀 분석에 가장 많이 사용되는 비용함수는 오차제곱합(sum of squared error)이며 최소자승법을 통해 파라미터를 추정한다. 능형 회귀 분석은 입력변수간의 다중공선성(multicollinearity) 문제를 해결하기 위해 제안된 방법으로 최소자승법을 개선하였다.

$$SSE = \sum_{i=1}^N (y_i - a_0 - \sum_{p=1}^n x_{ip} a_p)^2 \quad (8)$$

여기서, x_{ip} ($i=1, \dots, N; p=1, \dots, n$)은 입력변수를 나타내고, y_i 는 출력변수를 나타낸다. a_0 와 a_p 는 모델의 파라미터를 의미한다. 식 (8)은 식 (9)와 같은 정규 방정식(normal equation)으로 표현 된다.

$$A = (X^T X)^{-1} X^T Y \quad (9)$$

여기서, 입력변수간의 강한 선형관계(다중공선성)로 인해 $X^T X$ 가 특이행렬(singular matrix)이거나 특이행렬에 가까운 행렬이 되면 파라미터를 구할 수 없다. 따라서 L_2 정규화 방법은 특이행렬을 가역행렬로 만들어주기 위해 $X^T X$ 행렬의 대각원소에 임

의의 양의 실수를 더하며 식 (10)와 같이 표현된다.

$$A = (X^T X + \lambda I)^{-1} X^T Y \quad (10)$$

여기서, λ 는 정규화 매개변수(Regularization parameter)로 일반적으로 $\lambda > 0$ 값을 갖는다. 능형 회귀 모형의 경우 최적의 정규화 매개변수를 추정하기 위해 능형추적(ridge trace)을 수행한다.

L_2 정규화 방법에 의해 L_2 페널티 항을 적용한 오차제곱합은 식 (11)과 같이 표현된다.

$$\sum_{i=1}^N (y_i - a_0 - \sum_{p=1}^n x_{ip} a_p)^2 + \lambda \sum_{p=1}^n a_p^2 \quad (11)$$

본 논문에서는 L_2 정규화를 이용하여 PNN의 과적합 문제를 해결하기 위해 부분표현식의 파라미터 산출과정에 페널티항을 추가 하였다. 아래 그림은 λ 의 설정에 따른 결과의 차이를 보여준다. 그림5 과 같이 정규화 과정에서 λ 를 0이상의 값으로 매기지만 어떠한 값을 매기느냐에 따라 overfitting를 방지하기도 하지만 너무 큰 값을 주면 오히려 해를 찾지 못하는 모습을 볼수 있다. 때문에 λ 의 설정은 매우 중요하다.

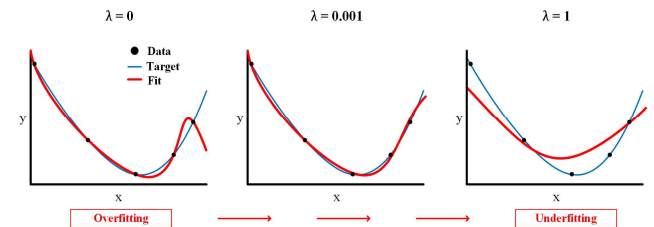


그림 5 Lamda값에 따른 L_2 정규화 효과
Fig. 5 L_2 regularization effect according to Lamda value

4. 실험 및 결과고찰

4.1 실험 방법

실험은 두 가지 방법으로 나누어서 실험 하였다. 방법 1에서는 각Layer의 노드 선택에 있어서 기존의 성능평가를 적용하여 L_2 regularization을 적용하지 않았을 경우와 L_2 regularization을 적용한 PNN 성능을 평가하는 실험을 진행한다. 방법 2는 각Layer의 노드 선택에 있어서 식 (6)의 SSC값이 가장 적은값을 선택하여 regularization을 적용하지 않았을 경우와 L_2 regularization을 적용한 PNN 성능을 평가하였다. Test 데이터의 성능평가는 사용하는 데이터에 따라서 MSE(Mean Squared Error)와 RMSE(Root Mean Squared Error)를 사용하였다. 식 (12)이 각각 MSE와 RMSE를 구하는 식을 나타낸다.

$$MSE = \frac{1}{N} \sum_{i=1}^N (\hat{y}_i - y_i)^2, \quad RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (\hat{y}_i - y_i)^2} \quad (12)$$

\hat{y}_i 는 모델의 출력을 나타내고, y_i 는 원 출력을 나타낸다.

4.2 실험에 사용한 데이터

Machine Learning 데이터인 가스터빈 발전소의 NOx 배출공정 데이터와 Automobile Miles per Gallon(MPG) 데이터 그리고 Boston Housing Data(BHD)를 이용하였다. 데이터는 전체데이터의 60%를 학습데이터로 전체 데이터의 40%를 평가 데이터로 사용하여 실험에 사용하였다.

4.2.1 가스 터빈 발전소의 NOx 배출 메커니즘 데이터

최근 환경보전의 필요성을 크게 인식하며 CO, CO2, SO, SO2, NO, N2O, NO2 및 프레온 가스 등과 같은 공해물질을 줄이고, 이들을 관리하기 위한 연구가 선진국을 중심으로 이루어지고 있다. 본 논문 역시 환경보전의 필요성을 인식하며, 이에 따른 화력발전소 대기오염물질의 배출패턴에 관한 모델을 정립하고 공기오염물질인 NOx의 배출량을 예측하고자 한다. 특히, 화력발전소의 가스 터빈에서 발생하는 독성이 강한 NOx의 배출 메커니즘

표 2 PNN의 실험 조건

Table 2 Experimental conditions of PNN

| | Parameters | Values |
|-----|--------------------------|-----------|
| PNN | Division of data | 60% : 40% |
| | Number of Inputs | [2 3] |
| | Number of selected nodes | 30 |
| | Polynomial type | quadratic |
| | lamda | [0 1] |

표 3 L₂ regularization을 적용하지 않은 NOx 데이터 성능지수의 값

Table 3 Values of performance index of NOx data without L₂ regularization

| Number of inputs | Layer | Without regularization (Method 1) | | | Without regularization (Method 2) | | |
|------------------|-------|-----------------------------------|--------|---------|-----------------------------------|--------|-------|
| | | PI | EPI | SSC 1 | PI | EPI | SSC 2 |
| 2input Quadratic | 1st | 4.133 | 4.446 | 0.529 | 23.53 | 18.366 | 0.001 |
| | 2nd | 2.647 | 4.484 | 846.017 | 3.61 | 4.239 | 4.544 |
| | 3rd | 1.516 | 3.342 | 1.926 | 2.413 | 3.145 | 0.512 |
| | 4th | 1.276 | 3.311 | 1.980 | 3.037 | 3.832 | 0.509 |
| | 5th | 1.195 | 3.730 | 0.780 | 3.081 | 3.594 | 0.505 |
| | 6th | 1.115 | 3.207 | 1.167 | 2.67 | 3.750 | 0.508 |
| | 7th | 1.079 | 51.421 | 0.659 | 2.653 | 3.717 | 0.51 |
| 3input Quadratic | 1st | 2.540 | 4.970 | 0.223 | 23.422 | 18.823 | 0.001 |
| | 2nd | 1.176 | 2.596 | 86.733 | 2.584 | 3.324 | 0.857 |
| | 3rd | 0.859 | 2.069 | 3.068 | 1.068 | 1.834 | 0.356 |
| | 4th | 0.682 | 16.301 | 2.912 | 1.084 | 1.474 | 0.367 |
| | 5th | 0.554 | 2.E+03 | 3.931 | 0.905 | 2.209 | 0.462 |

Method 1 : Method based on selection of the best PI, Method 2 : Method based on selection of minimal SSC

SSC 1 : Value of SSC of the best PI in each layer, SSC 2 : Value of minimal SSC in each layer

에 제안된 알고리즘을 도입하여 배출 패턴 모델을 구축하고 평가한다[10]. 가스 터빈 발전소의 NOx 배출 메커니즘 데이터는 총 5입력(Tamb, COM, LPT, Pcd, Texh) 1출력(NOx)의 260개의 데이터로 구성되어 있으며, 모델의 성능지수 평가방법은 식 (13)에서와 같이 RMSE를 이용하였고 다른 모델링 과의 비교성능을 표 5에 나타내었다.

표 3과 그림 6(a), (b)는 L₂ regularization을 적용하지 않았을 때의 방법 1과 2에서의 PNN 성능을 보여주고 있다. 방법 1에서는 2입력 일 때 7층에서 EPI가 51.4216 이고 3입력 일 때에는 4층에서 16.3013로 올라가 PI와의 성능차가 매우 심하게 벌어지는 overfitting이 발생 하였다. 방법 2에서는 2입력일 때와 3입력일 때 PI와 EPI의 성능차가 거의 없었고, EPI의 성능도 매우 좋아진 것을 알 수 있었다. 그림 6(c)에서는 방법 1과 2 실험에서의 각층의 SSC의 크기를 비교 했다. 방법 1의 SSC 크기는 4층까지 불안정한 모습을 보이다 이후 안정되었고, 방법 2에서는 2입력일 때 2층에서 4.5445까지 오르는 모습을 보이다 이후 0.5 수준으로 비슷한 크기를 보이고 3입력에서도 거의 일정한 수준을 보여 방법 2의 모델에서 근사화 및 일반화 성능이 우수함을 알 수 있다.

표 4와 그림 7(a), (b)는 방법 1과 방법 2에 L₂ regularization을 적용하여 PNN의 성능을 비교 하였다. 2입력에서 방법 1은 성능이 나아지는 것을 확인하였지만 PI와 EPI의 격차가 점차 벌어지는 것에 비하여 방법 2에서는 PI와 EPI의 격차가 점차 줄어들었고 7층에서의 EPI도 2.2968로 방법 1에서의 EPI인 2.7379보다 낮은 성능을 보이며 일반화 성능 또한 방법 1보다 우수함을 알 수 있다. 3입력에서는 방법 1과 2에서의 성능이 4층까지 낮아지다 5층부터 overfitting이 발생하였다. L₂ regularization을 적용하였을 때 방법 1과 2에서의 SSC는 2입력에서는 방법 2가 매우 안정적이지만 3 입력에서는 방법 1이 좀 더 안정적인 모습을 보였다.

표 4 L₂ regularization을 적용한 NOx 데이터 성능지수의 값
Table 4 Values of performance index of NOx data with L₂ regularization

| Number of inputs | Layer | With regularization (Method 1) | | | With regularization (Method 2) | | |
|------------------|-------|--------------------------------|--------------|--------------|--------------------------------|--------------|--------------|
| | | PI | EPI | SSC | PI | EPI | SSC |
| 2input Quadratic | 1st | 4.134 | 4.447 | 0.529 | 23.53 | 18.366 | 0.001 |
| | 2nd | 2.878 | 3.609 | 108.24 | 4.737 | 4.604 | 0.478 |
| | 3rd | 1.572 | 3.283 | 2.915 | 4.796 | 5.612 | 0.446 |
| | 4th | 1.363 | 2.866 | 0.869 | 3.492 | 4.364 | 0.498 |
| | 5th | 1.261 | 2.847 | 0.625 | 2.432 | 3.097 | 0.514 |
| | 6th | 1.186 | 2.908 | 4.341 | 2.073 | 2.575 | 0.505 |
| | 7th | 1.118 | 2.737 | 1.565 | 1.985 | 2.296 | 0.513 |
| 3input Quadratic | 1st | 2.562 | 4.981 | 0.223 | 23.422 | 18.823 | 0.001 |
| | 2nd | 1.54 | 2.901 | 0.415 | 2.601 | 3.336 | 0.85 |
| | 3rd | 1.017 | 2.364 | 0.387 | 1.067 | 1.459 | 0.356 |
| | 4th | 0.957 | 1.251 | 0.396 | 1.049 | 1.3 | 0.357 |
| | 5th | 0.945 | 5.145 | 0.36 | 0.972 | 11.76 | 0.483 |

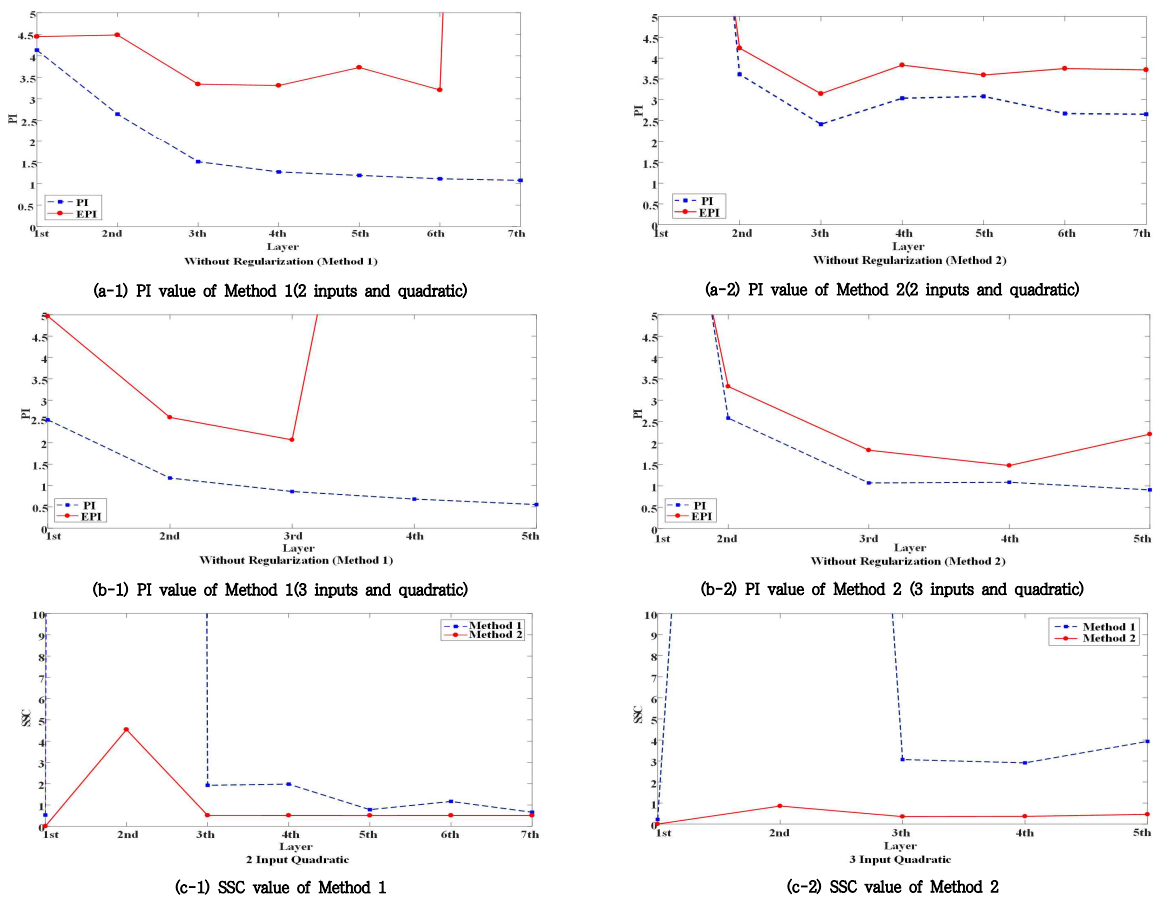


그림 6 L2 regularization을 적용하지 않은 NOx 데이터 성능지수의 값
 Fig. 6 Values of performance index of NOx data without L2 regularization

표 5 다른 모델링과의 성능 비교
 Table 5 Comparison of performance of the proposed model with other models

| Model | | PI | EPI |
|--------------------------------|------------------------|----------|--------|
| Regression Model[13] | | 16.862 | 19.684 |
| FNN[13] | | 3.725 | 5.291 |
| RBFNN [13] | | 0.703 | 3.717 |
| RBFNN Using K-means clustering | | 1.004 | 3.603 |
| Proposed PNN | Without regularization | Method 1 | 0.859 |
| | | Method 2 | 1.084 |
| | With regularization | Method 1 | 0.957 |
| | | Method 2 | 1.049 |

4.2.2 Automobile Miles per Gallon(MPG)데이터

제안된 모델의 성능 평가를 위해서 시스템 모델링에 널리 사용되는 비선형 데이터 Automobile Miles per Gallon(MPG)을 사용했다. (ftp://ics.uci.edu/pub/machine-learning-datasets/auto-mpg) [11]. MPG 데이터는 차량의 고유한 정보에 대한 기준 용량(1 Gallon)의 연료로 차량이 주행한 거리(Miles)관계를 갖

표 6 L2 regularization을 적용하지 않은 MPG 데이터 성능지수의 값
 Table 6 Values of performance index of MPG data without L2 regularization

| Number of inputs | Layer | Without regularization (Method 1) | | | Without regularization (Method 2) | | |
|------------------|-------|-----------------------------------|--------|-------|-----------------------------------|-------|-------|
| | | PI | EPI | SSC | PI | EPI | SSC |
| 2input Quadratic | 1st | 3.038 | 2.845 | 0.001 | 3.038 | 2.845 | 0.001 |
| | 2nd | 2.801 | 2.838 | 9.259 | 4.162 | 4.114 | 0.631 |
| | 3rd | 2.655 | 3.012 | 6.748 | 3.785 | 3.621 | 0.532 |
| | 4th | 2.573 | 2.876 | 8.745 | 2.919 | 2.807 | 0.533 |
| | 5th | 2.499 | 7.953 | 0.960 | 3.00 | 3.059 | 0.494 |
| | 6th | 2.439 | 18.29 | 18.01 | 2.989 | 3.167 | 0.504 |
| | 7th | 2.369 | 4.E+04 | 67.78 | 2.958 | 3.051 | 0.643 |
| 3input Quadratic | 1st | 2.866 | 2.737 | 0.665 | 2.986 | 2.91 | 0.004 |
| | 2nd | 2.546 | 3.032 | 67.53 | 2.754 | 2.637 | 0.338 |
| | 3rd | 2.379 | 6.919 | 22.82 | 2.746 | 2.774 | 0.357 |
| | 4th | 2.244 | 60.11 | 6.457 | 2.695 | 2.625 | 0.373 |
| | 5th | 2.121 | 2.E+04 | 3.371 | 2.644 | 2.756 | 1.029 |

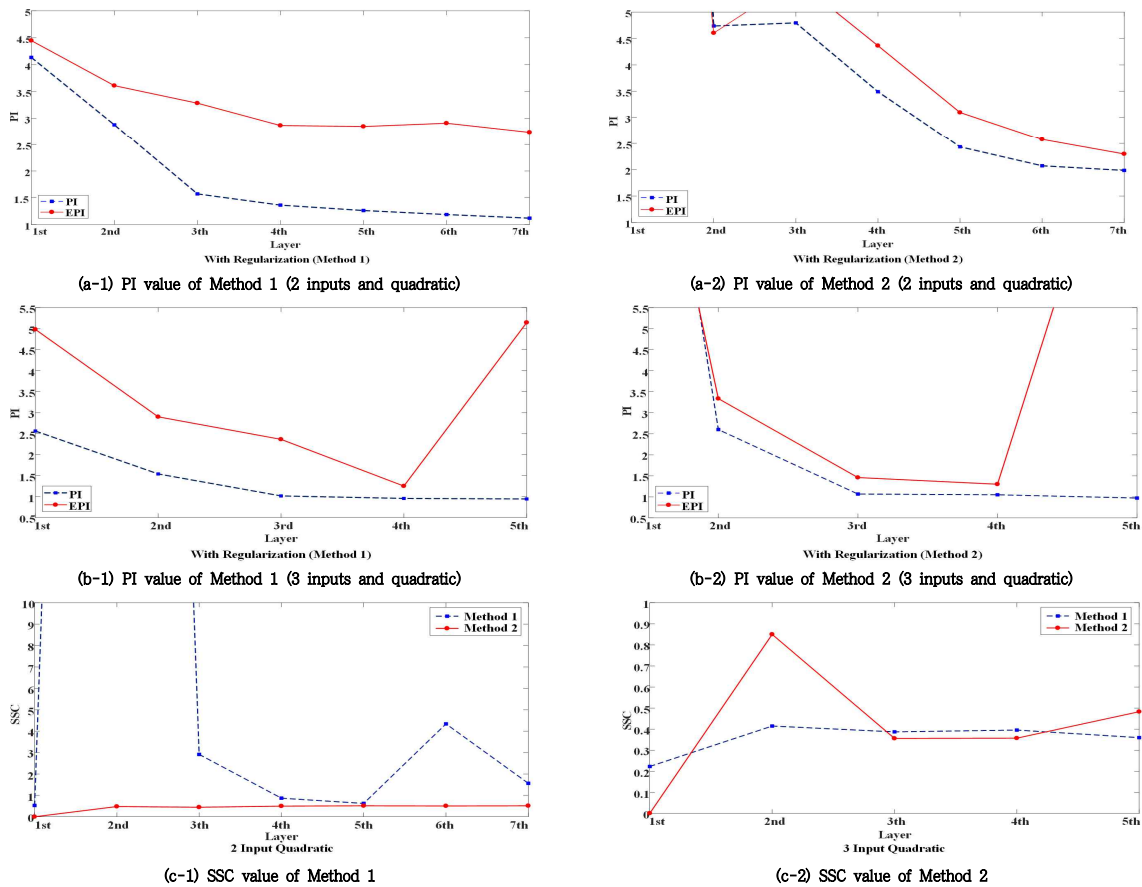


그림 7 L2 regularization을 적용한 NOx 데이터 성능지수의 값
Fig. 7 Values of performance index of NOx data with L2 regularization

는다. MPG는 총392개의 입출력데이터 쌍을 가지며, 7입력-1출력으로 구성되어 있다. 입력변수는 cylinders, displacement, horse power, weight, acceleration, model year, origin으로 구성되어 있으며, 출력변수는 앞서 말한 gallon당 주행거리가 된다. 제안된 모델의 평가를 위하여 MPG 데이터를 학습데이터(60%)와 테스트 데이터(40%)로 데이터를 무작위로 나누어 실험했다. 성능지수는 RMSE를 사용하였고 다른 모델링과의 비교성능을 표 8에 나타내었다.

표 6은 L₂ regularization을 적용하지 않았을 경우의 방법 1과 2에서의 PNN 성능을 보여주고 있다. 방법 1에서는 2입력일 때 6층에서 EPI가 18.2922이고 3입력일 때에는 4층에서 60.111로 올라가 PI와의 성능차가 매우 심하게 벌어지는 overfitting이 발생 하였다. 방법 2에서는 2입력일 때와 3입력일 때 PI와 EPI의 성능차가 거의 없었고, EPI의 성능도 매우 좋아진 것을 알 수 있었다.

표 7은 방법 1과 2에 L₂ regularization을 적용하여 PNN의 성능을 비교 하였다. 2입력에서 방법 1과 2에서의 PI와 EPI의 차이가 거의 없이 전개되었으며, 방법 1의 6층에서의 EPI가 2.6767로 L₂ regularization을 적용함으로써 성능이 많이 개선됨을 확인했다. 3입력에서는 방법 1의 성능이 1층 이후 개선되지 않고 4층

부터 overfitting이 발생하였다. 방법 2에서는 성능이 개선되어 성능의 우수함을 알 수 있다.

표 7 L₂ regularization을 적용한 MPG 데이터 성능지수의 값
Table 7 Values of performance index of MPG data with L₂ regularization

| Number of inputs | Layer | With regularization (Method 1) | | | With regularization (Method 2) | | |
|------------------|-------|--------------------------------|--------|-------|--------------------------------|-------|-------|
| | | PI | EPI | SSC | PI | EPI | SSC |
| 2input Quadratic | 1st | 3.039 | 2.845 | 0.001 | 3.039 | 2.845 | 0.001 |
| | 2nd | 2.984 | 3.103 | 13.51 | 4.343 | 4.347 | 0.526 |
| | 3rd | 2.685 | 2.782 | 1.675 | 4.226 | 3.815 | 0.531 |
| | 4th | 2.618 | 2.741 | 0.996 | 4.226 | 3.867 | 0.501 |
| | 5th | 2.561 | 2.729 | 0.633 | 2.991 | 2.983 | 0.503 |
| | 6th | 2.518 | 2.676 | 0.516 | 2.951 | 2.93 | 0.509 |
| | 7th | 2.514 | 3.038 | 0.691 | 2.949 | 2.929 | 0.509 |
| 3input Quadratic | 1st | 2.871 | 2.743 | 0.665 | 2.986 | 2.918 | 0.004 |
| | 2nd | 2.649 | 2.962 | 8.544 | 3.11 | 3.323 | 0.376 |
| | 3rd | 2.473 | 2.984 | 1.979 | 2.651 | 2.659 | 0.370 |
| | 4th | 2.352 | 6.0333 | 1.142 | 2.578 | 2.821 | 0.413 |
| | 5th | 2.266 | 27.24 | 2.073 | 2.588 | 2.735 | 0.423 |

표 8 다른 모델링과의 성능 비교

Table 8 Comparison of performance of the proposed model with other models

| Model | | PI | EPI | |
|--------------------------------|------------------------|----------|-------|-------|
| Regression Model[13] | | 3.202 | 3.485 | |
| RBFNN [13] | | 2.233 | 2.342 | |
| RBFNN Using K-means clustering | | 2.32 | 2.924 | |
| Proposed PNN | Without regularization | Method 1 | 2.866 | 2.737 |
| | | Method 2 | 2.695 | 2.625 |
| | With regularization | Method 1 | 2.518 | 2.676 |
| | | Method 2 | 2.651 | 2.659 |

표 9 다른 모델링과의 성능 비교

Table 9 Comparison of performance of the proposed model with other models

| Model | | PI | EPI | |
|--------------------------------|------------------------|----------|-------|-------|
| Regression Model[13] | | 4.741 | 3.956 | |
| RBFNN [13] | | 3.117 | 3.995 | |
| RBFNN Using K-means clustering | | 3.083 | 4.127 | |
| Proposed PNN | Without regularization | Method 1 | 3.787 | 3.927 |
| | | Method 2 | 3.438 | 3.850 |
| | With regularization | Method 1 | 4.156 | 4.266 |
| | | Method 2 | 3.327 | 3.439 |

4.2.3 Boston Housing Data(BHD)

이 데이터는 보스턴 지역의 부동산의 정보와 관련이 있으며, 13 입력-1출력으로 구성된 총 506개의 입출력데이터 쌍으로 이루어져 있다.(ftp://ftp.ics.uci.edu/pub/machine-learningdatabases/housing/housing.data)[12]. 입력변수는 부동산에 영향을 미치는 요인들로 이루어져 있으며, 출력변수는 부동산의 평균가격을 나타낸다. 제안된 모델의 평가를 위하여 boston housing 데이터를 학습데이터(60%)와 검증데이터(40%)로 랜덤하게 나누었고, 성능지수는 식(12)과 같이 RMSE를 사용하였고 다른 모델링과의 비교성능을 표 9에 나타내었다.

5. 결 론

본 논문에서는 기존 PNN의 overfitting 문제를 개선하기 위하여 새로운 방법을 제시하고, L_2 regularization을 적용하여 개선된 PNN을 제안하여 Deep 네트워크구조의 가능성을 제시하였다. 제안된 PNN 구조의 특징은 복잡한 비선형계의 근사 동정과 예측이 가능하고, 입력 변수의 선택방법, 다항식 차수와 입력변수수에 의한 부분표현식의 표현방법, 각 네트워크 구조의 선택방법 등에 따른 최적구조 설계를 통하여 주어진 시스템 특성에 맞는 구조로 적응능력을 향상시킨 모델 구조를 생성할수 있다. 하지만 층이 증가함에 따라 입력변수간의 다중공선성 문제로 인한 과적합 또한 증가하기 때문에 이를 해결하기 위하여 노드의 계수의 제곱합을 노드의 평가에 사용하고 L_2 regularization을 적용하여

개선시키고자 하였다. 성능 평가를 위해 Machine Learning 데이터인 Nox, MPG, BHD 데이터를 이용하였다. 이를 기존의 PNN과 제안된 PNN을 비교하였고, 계수의 제곱합을 사용하는 방법과, L_2 regularization을 적용하였을 때 과적합의 발생이 늦어진 것을 확인할 수 있었다. 결과적으로 계수의 제곱합(Sum of Squared Coefficients : SSC)기법 및 L_2 -norm regularization 기법의 도입을 통한 제안된 방법의 PNN은 기존의 PNN의 overfitting 문제를 효과적으로 개선할 수 있을 뿐 아니라, 딥구조를 통한 빅데이터 처리의 효율적인 처리 가능성을 제시하였다. 또한 다른 모델링과 성능을 비교하여 딥 구조 PNN의 성능이 기존의 PNN뿐만 아니라 다른 모델링방법에 비해 선호된 성능의 네트워크 구조를 생성할 수 있음을 보였다.

감사의 글

This study was supported by the R&D Center for Valuable Recycling (Global-Top R&D Program) of the Ministry of Environment.(Project No:2016002250002)

References

- [1] V. Rouss, W. Charon, and G. Cirrincione, "Neural model of the dynamic behaviour of a non-linear mechanical system," Mechanical Systems and Signal Processing, Vol 23, pp. 1145-1159, 2009.
- [2] A. G. Ivahnenko, "Polynomial theory of complex systems," IEEE Trans. on Systems, Man and Cybernetics, Vol. SMC-1, pp. 364-378, 1971.
- [3] S. K. Oh and W. Pedrycz, "The Design of Self-Organizing Polynomial Neural Networks,"Information Sciences, Vol. 141, pp. 237-258, 2002.
- [4] S. K. Oh, T. C. Ahn, and W. Pedrycz, "Fuzzy Polynomial Neural Networks-Based Structure and Its Application to Nonlinear Process Systems," 7th IFSA World Conference, Vol. 2, pp. 495-499, 1997.
- [5] Ho-Sung Park, Ki-Sang Kim, Sung-Kwun Oh "Design of Particle Swarm Optimization-based Polynomial Neural Networks"THE TRANSACTION OF THE KOREAN INSTITUTE OF ELECTRICAL ENGINEERS 60(2), 2011.2, 398-406
- [6] AE Hoerl, RW Kennard. "Ridge regression: Biased estimation for nonorthogonal problems", Technometrics, Vol. 12, No. 1, p. 55-67, 1970
- [7] A. N. Tikhonov, and V. Y. "Arsenin, solution of ill-posed problems.", Washington: Winston & Sons, 1977
- [8] Q. Fan, J. M. Zurada, W. Wu, "Convergence of online gradient method for feedforward neural networks with

smoothing L1/L2 regularization penalty", Neurocomputing, Vol 131, pp. 208-216, 2014

- [9] Y. H. Pyo, K. H. Lee, K. H. You, "Earthquake magnitude estimation using Recursive Least Squares", IEEE International Conference on Cloud Computing and Big Data, pp. 394-397, 2016
- [10] H. S. Park, Y. H. Jin, S. K. Oh, "Evolutionary Design of Radial Basis Function-based Polynomial Neural Network with the aid of Information Granulation", The Transactions of Korean Institute of Electrical Engineers, Vol. 60, No. 4, pp. 862-870, 2011.
- [11] S. K. Oh, W. D. Kim, H. S. Park, M. H. Son, "Identification Methodology of FCM-based Fuzzy Model Using Particle Swarm Optimization", The Transactions of Korean Institute of Electrical Engineers, Vol. 60, No. 1, pp. 184-192, 2011.
- [12] L. P. Maguire, B. Roche, T. M. McGinnity, L. J. McDaid, "Predicting a chaotic time series using a fuzzy neural network", Information Science, Vol. 112, pp. 125-136, 1998.
- [13] S. K. Oh, Y. H. Kim, H. S. Park, J. T. Kim, "Design of Data-centroid Radial Basis Function Neural Network with Extended Polynomial Type and Its Optimization", The Transactions of Korean Institute of Electrical Engineers, Vol. 60, No. 3, pp. 639-647, 2011.



오 성 권(Sung-Kwun Oh)

1981년 연세대학교 전기공학과 공학사, 동 대학원 석사(1983), 박사(1993), 1983~1989년 금성산전연구소(선입연구원). 1996~1997년 캐나다 Manitoba 대학 전기 및 컴퓨터공학과 Post-Doc. 1993~2004년 원광대학교 전기전자 및 정보공학부 교수, 2005~현재 수원대학교 전기전자공학부 교수. 2002~현재 대한전기학회, 퍼지및지능시스템학회 편집위원, 2013년~현재 Information Sciences 편집위원. 관심분야는 퍼지 시스템, 퍼지-뉴럴 네트워크, 자동화 시스템, 고급 computational intelligence, 지능 제어 등.

Tel: 031-229-8162, Fax: 031-220-2667

E-mail: ohsk@suwon.ac.kr

저 자 소 개



전 필 한 (Pil-Han Jeon)

2016년 수원대학교 전기공학과 공학사, 2016~현재 수원대학교 전기공학과 석사과정. 관심분야는 퍼지 이론, 신경 회로망, 클러스터링, 최적화 알고리즘, 컴퓨터 지능 및 지능 제어 등.

Tel: 031-222-6544

E-mail: gks2991@suwon.ac.kr



김 은 후 (Eun-Hu Kim)

2009년 수원대학교 전기공학과 공학사, 동 대학원 석사(2011), 박사(2016), 2016~현재 수원대학교 전기공학과 Post-Doc. 관심분야는 퍼지 이론, 신경 회로망, 클러스터링, 최적화 알고리즘, 컴퓨터 지능 및 지능 제어 등.

Tel: 031-222-6544, Fax: 031-220-2667

E-mail: wdkim@suwon.ac.kr