인공신경망을 이용한 개인 신용평가 알고리즘

A Personal Credit Estimate Algorithm Using Artificial Neural Network

임성빈, 최우경, 김성현*, 김용민**, 전홍태
중앙대학교 전자전기공학부
*동원대학 디지털정보전자과
** 충청대학교 컴퓨터학부

Sung-Bin Lim, Woo-Kyung Choi, Sung-Hyun Kim*, Yong-Min Kim**, Hong-Tae Jeon
School of Electrical & Electronic Engineering, Chung-Ang university
*Dept. of Digital Information Electronics, Tong-Won Collage
**Dept. of Computer Science, Chung-Cheong University
E-mail : lovebiny@hanmail.net

요 약

최근 우리나라는 가계신용의 급신장과 신용불량의 급증 등으로 개인 신용부문이 금융기관의 건전성 유지에 부정적인 영향을 미치고 있다. 이러한 잠재적 문제를 사전에 방지하기 위해 금융기관 등에서는 개인 신용평가에 대한 수요가 커지고 있는 실정이다. 주어진 데이터로부터의 반복적인 학습 과정을 거쳐 패턴을 분류하고 또한 모델과 학습 방법에 따라 입력변수와 목적변수의 속성이 연속 형이나 이산형인 경우를 모두 다룰 수 있는 신경망 모델은 개개인의 다양하고 복잡한 데이터를 입력변수로 받아서 신용등급을 나누는데 우수한 능력을 보여줄 수 있다. 본 논문에서는 신경망 모델을 이용해 개인의 신용등급을 객관적이고 일률적으로 평가해서 등급을 나누어주는 알고리즘을 제안하고자 한다.

Key words : Neural Network, Credit Estimate, Learning, Pattern Classification

Ⅰ. 서론

인간의 행위 중 정보를 받아들여 이해하고 미래를 예측하는 행위를 학습이라고 정의할 때, 신용평가는 인공지능 분야 중에서도 자동학습 분야에 해당한다. 두뇌를 통한 학습은 과거에 수행한 작업 또는 작업들의 조합을 이용하여 새로운 상황에서 보다 효율적이고 효과적인 처리를 할 수 있도록 하는 변화의 과정이다. 자동학습은 인간의 이러한 학습 능력을 구현하고자 하는 시도이다[1]. 우리가 일반적으로 말하는 인공 신경망이란 컴퓨터가 사람의 학습 기능을 갖게 하기 위해 고안된 것이다. 학습의 기능을 갖는 사람의 두뇌는 다수의 뉴런이 서로 연결된 신경망으로 구성되어 있다. 인공 신경망은 이러한 생물학적 신경망에서 아이디어를 얻어 그래프 형태와 수학적 알고리즘으로 모델링한 것이다. 신경망은 생물학적 뉴런을 모델링한 유닛(unit)들과 그 유닛 사이

의 연결강도(weight)들로 이루어지며 반복과 훈련을 통해 각 유닛 사이의 안정적인 연결강도를 찾아간다[4].

신용평가는 경영 분류(business classification) 문제의 한 범주로 볼 수 있다. 분류는 여러 가지 개체 또는 사건들을 서로 다른 특성을 가진 집단으로 분류하고, 새로운 개체 또는 사건을 미리 정의된 집단으로 할당하는 것이다. 이 점에서 신용평가 모델은 표준으로 선정된 개인들을 신용도에 따라 몇 개의 집단으로 분류하고, 각각의 집단에 신용등급을 부여한 후, 새로운 평가 대상이 어떤 등급을 받을지 예측하는 것이다. 분류는 개체 또는 사건을 서로 다른 속성을 가지도록 분리하고 할당하는 알고리즘을 필요로 하는데, 이러한 알고리즘을 분류 기법(classification technique)이라 한다[1].

이전부터 분류 문제를 해결하기 위해 통계적인 방법을 비롯해 여러 가지 방법이 존재해 왔지만 그 중에서도 인공 지능을 이용한 방법이 좋은 성과를 보여주고 있다. 본고에서는 다양한 인공지능 방법(의사결정나무, 사례기반 추론, 인공신경망, 유전자 알고리즘 등)중에서 인공 신경망을 이용한 개인의 신용평가 알고리즘에 대해서 다루어 볼 것이다.

Ⅱ. 본론

2.1 인공 신경망(Artificial Neural Network)

인공 신경망은 인간의 두뇌와 유사한 처리를 하는 기계를 만들기 위한 노력이다. 물론 인간의 두뇌가 어떻게 활동하고 지능이 어떤 것인지에 대한 확정적인 결론은 없다. 인공신경망은 뉴론이라 불리는 인간의 신경세포와 유사한 처리요소(Processing Element, PE)로 이루어져 있다[4].

PE는 입력, 출력, 가중치, 뉴론 함수의 네 부분으로 되어 있다. 입력과 출력은 0/1, 연속치, 집단 등의 다양한 형태를 가질 수 있다. 각각의 입력은 효과에 대한 상대적인 가중치를 가지고, 가중치는 입력신호의 정도를 나타내기 위해 모형 내에서 결정된다. 뉴론 함수는 합산(summation), 활성화(activation), 전이(transfer), 학습(learning)의 네 가지가 있다. 합산 함수의 가중치에 따라 입력을 더하고, 그 결과는 활성화 함수의 입력이 된다. 전이함수는 활성화 함수의 결과를 받아 이미 정해진 크기와 비교한 후 다음 PE에게 적절한 가중치를 가진 출력을 보내준다. 학습함수는 현재의 출력과 원하는 출력을 비교하여 오차를 감소시킨다. 병렬적으로 동시에 실행되는 PE의 집합을 층(layer)이라 부른다. 입력 층과 출력 층이 문제에서 정의된 변수에 따라 확정적인데 반해 은닉 층 PE의 수는 유동적이다[1].

인공신경망은 크게 3가지의 유용한 특성을 가지고 있는데, 결함 내구성(fault tolerance)과 적응성(adaptability) 그리고 일반화(generalization)이다. 처리 노드가 많기 때문에 몇 개의 노드나 연결이 가진 결함이 비교적 시스템 전체의 결함을 초래하지는 않는다. 이러한 특성을 결함 내구성이라고 한다. 적용성은 인공 신경망의 자기 변화를 의미하는 것으로, 지속적인 학습을 통해 가중치를 변화시키면서 적합한 모델을 만드는 과정을 말한다. 마지막으로 일반화란 불완전하거나 사전에 알 수 없던 입력을 표현하는 경우에도 신경망은 합리적인 반응을 생성할 수 있다는 것이다.[1][2]

2.2 다층 퍼셉트론과 전파 학습 알고리즘

다층 퍼셉트론은 입력 층과 출력 층 사이에 하나 이상의 중간층이 존재하는 신경망으로 <그림 1> 에 나타낸 것과 같은 계층구조를 갖는다.

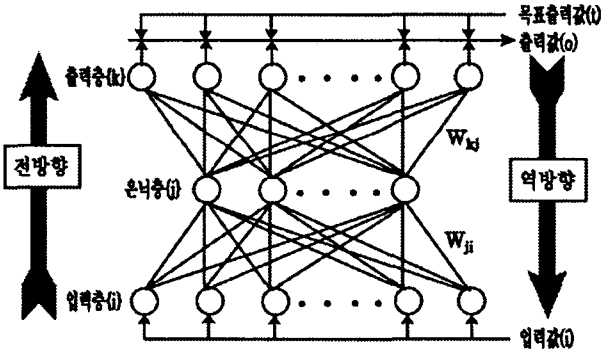

그림 1. 다층퍼셉트론의 구조

이 때 입력층과 출력층 사이의 중간층을 은닉층 (hidden layer) 이라 부른다. 네트워크는 입력층, 은닉층, 출력층 방향으로 연결되어 있으며, 각 층내의 연결과 출력 층에서 입력층으로의 직접적인 연결은 존재하지 않는 전방향(feed-for ward)네트워크이다. 다층 퍼셉트론은 단층 퍼셉트론과 유사한 구조를 가지고 있지만 중간층과 각 유닛의 입출력 특성을 비선형으로 함으로써 네트워크의 능력을 향상시켜 단층 퍼셉트론의 여러가지 단점들을 극복했다. 다층 퍼셉트론은

층의 수가 증가할수록 퍼셉트론이 형성하는 결정 구역의 특성은 더욱 고급화된다. 즉 단층일 경우 패턴공간을 두 구역으로 나누어주고, 2층인 경우 볼록산 (convex) 개 구역 또는 오목한 폐 구역을 형성하며, 3층인 경우에는 이론상 어떠한 형태의 구역도 형성할 수 있다[2][5].

역전파 학습알고리즘은 최소자승 (least mean square)알고리즘의 비선형적인 확장이다. 역전파 학습알고리즘은 미분의 반복규칙(chain-rule) 을 여러 번 반복적으로 적용하여 확률 근사치 프레임워크(stochastic-approximation framework) 와 관련지음으로써 유도해낼 수 있다. 역전파 학습알고리즘의 기본 원리는 다음과 같다. 입력층의 각 유닛에 입력패턴을 주면, 이 신호는 각 유닛에서 변환되어 중간층에 전달되고 최후에 출력층에서 신호를 출력하게 된다. 이 출력 값과 기대 값을 비교하여 차이를 줄여나가는 방향으로 연결강도를 조절하고, 상위층에서 역전파하여 하위 층에서는 이를 근거로 다시 자기 층의 연결강도를 조정해나간다[2][3].

[Backpropagation Learning Algorithm]

1. Apply the input vector, $X_p = (x_{p1}, x_{p2}, \ldots, x_{pN})^T$ to the input units.

2. Calculate the net-input values to the hidden layer:

$$net_{pj}^h = \sum_i w_{ji}^h x_{pi} + \theta_j^h$$

3. Calculate the outputs from the hidden layer:

$$i_{pj} = f_j^h(net_{pj}^h)$$

4. Move to the output layer.
Calculate the net-input values to each unit:

$$net_{pk}^o = \sum_j w_{kj}^o i_{pj} + \theta_k^o$$

5. Calculate the outputs:

$$o_{pk} = f_k^o(net_{pk}^o)$$

6. Calculate the error terms for the output units:

$$\delta_{pk}^o = (y_{pk} - o_{pk}) f_k^{o'}(net_{pk}^o)$$

7. Calculate the error term for the hidden unit:

$$\delta_{pj}^o = f_j^{h'}(net_{pj}^h) \sum_k \delta_{pk}^o w_{kj}^o$$

8. Update weights on the output layer:

$$w_{kj}^o(t+1) = w_{kj}^o(t) + \eta \cdot \delta_{pk}^o i_{pj}$$

9. Update weights on the hidden layer:

$$w_{ji}^h(t+1) = w_{ji}^h(t) + \eta \cdot \delta_{pj}^h x_{pi}$$

10. Be sure to calculate the error term:

$$E_p = \frac{1}{2} \sum_{k=1}^{M} \delta_{pk}^2$$

Ⅲ. 분류 알고리즘

본 논문에서는 신용평가 등급을 분류하는데 있어서 아래의 < 그림 2 > 와 같이 1개의 은닉층을 갖는 다층 신경망 구조로 직업유무, 연봉, 나이, 대출금, 부동산 보유량 등의 5개의 변수를 입력받은 후, 출력 단에서는 5개의 등급으로 분류를 해주는 알고리즘을 제안한다.

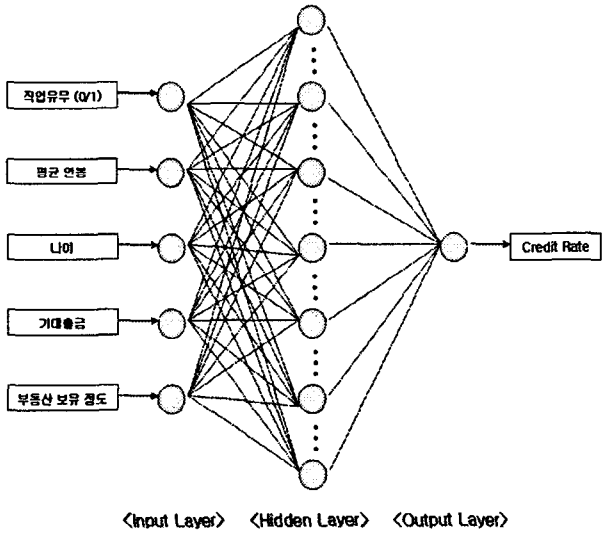

그림 2. 신용평가를 위한 신경망의 구조

신경망을 학습시키기 위해 임의로 100명의 데이터를 작성해서 등급을 분류한 후 그 수치를 test training set으로 사용하였다. 허용오차를 0.05로 제한하고 학습반복횟수는 100,000번으로 하여 은닉층 노드의 개수를 10, 15, 20, 25개로 변경해 가면서 학습을 시켜 보았다. 노드를 15개 이상으로 할 시에는 학습시간만 지연될 뿐 성능 향상이 없었기에 은닉 층의 노드를 15개로 고정하고 학습을 시킨 후 임의의 테스트용 데이터로

시험을 해본 결과 만족할만한 결과가 나오는 것을 볼 수 있었다.

Ⅳ. 결론 및 향후과제

신용등급을 분류하기 위한 방법으로는 전통적인 방법(주로 통계적인 방법)과 인공지능을 이용한 방법(의사결정나무, 사례기반 추론, 인공신경망, 유전자 알고리즘 등)이 있다. 이러한 방법 중, 인공신경망은 주어진 데이터로부터의 반복적인 학습 과정을 거쳐 패턴을 분류하고 또한 모델과 학습 방법에 따라 입력변수와 목적변수의 속성이 연속 형이나 이산형인 경우를 모두 다룰 수 있기 때문에 개개인의 다양한 데이터를 입력변수로 받아서 신용등급을 분류하는데 우수한 능력을 보여줄 수 있다. 본 논문에서는 하나의 은닉 층을 가진 다층 신경망과 오차 역 전파 학습 방법을 이용하여 개개인의 신용등급을 분류해주는 알고리즘을 제안하였다. 이를 바탕으로 해서 알고리즘을 더 구체화 시킨다면 은행에서 대출여부의 판단이나 한도액 결정 또는 카드회사에서 신용카드 발급여부를 결정하는데 용이하게 쓰일 수 있을 것이다. 그러나 실제 환경에서는 보다 다양하고 복잡한 개인 정보들이 사용될 것이며 그럴 경우에는 각각의 PE에 과다한 가중치가 물려 계산량이 기하급수적으로 늘어나서 효율이 떨어질 수가 있다. 이를 해결하기 위해 계층적인 구조를 사용한다거나 새로운 망구조를 개발하는 등의 연구가 필요하다.

감사의 글 : 본 논문은 뇌신경 정보학 연구사업에 의해 지원 받았습니다.

Ⅴ. 참고문헌

[1] 조홍규, "인공지능방법을 이용한 신용평가 모형에 대한 개관", 2003.

[2] 김대수, 신경망 이론과 응용(1), 하이테크정보, 1992.

[3] James A. Freeman, David M. Skapura, Neural Networks ; Algorithm, Applications, and Programing techniques, Addison-Wesley Publishing Company, 1991.

[4] Jyh-Shing Roger Jang, Chuen-Tsai Sun, Eiji Mizutani, Neuro-Fuzzy and Soft Computing A Computational Approach to Learning and Machine Intelligence, Prentice-Hall, Inc. 1997.

[5] Simon Haykin, NEURAL NETWORKS ; a comprehensive foundation, second edition, Prentice-Hall, Inc. 1999.

[6] Draelos, T., Hush, D., "A constructive neural network algorithm for function approximation", *IEEE International Conference on Neural Networks*, vol. 1, pp. 50-55, 1996.

[7] Chin-Teng Lin, C.S.George Lee, Neural Fuzzy Systems ; A Neuro-Fuzzy Synergism to Intelligent Systems, Prentice Hall, 1996.