

신경망을 이용한 시계열의 분해분석[†]

지원철

홍익대학교 정보산업공학과

Decomposition Analysis of Time Series Using Neural Networks[†]

Won Chul Jhee

This evapaper is toluate the forecasting performance of three neural network(NN) approaches against ARIMA model using the famous time series analysis competition data. The first NN approach is to analyze the second Makridakis (M2) Competition Data using Multilayer Perceptron (MLP) that has been the most popular NN model in time series analysis. Since it is recently known that MLP suffers from bias/variance dilemma, two approaches are suggested in this study. The second approach adopts Cascade Correlation Network (CCN) that was suggested by Fahlman & Lebiere as an alternative to MLP. In the third approach, a time series is separated into two series using Noise Filtering Network (NFN) that utilizes autoassociative memory function of neural network. The forecasts in the decomposition analysis are the sum of two prediction values obtained from modeling each decomposed series, respectively. Among the three NN approaches, Decomposition Analysis shows the best forecasting performance on the M2 Competition Data, and is expected to be a promising tool in analyzing socio-economic time series data because it reduces the effect of noise or outliers that is an impediment to modeling the time series generating process.

1. 서 론

시계열분석은 과거를 이해하고 미래를 예측하기 위해 기관측된 현상들을 설명할 수 있는 법칙을 찾는 것이다. 주어진 시계열의 생성과정에 대한 확정적 모형이 알려져 있다면 시스템의 초기조건만으로 원하는 예측값을 쉽게 구할 수 있지만, 시계열의 생성과정에 대한 모형이 알려져 있지 않는 경우가 대부분이므로 대상시스템의 변화를 지배하는 법칙과 시스템의 실제 상태를 동시에 파악하여야만 한다. 따라서, 시계열분석은 시계열 생성과정에 대한 모형을 과거자료에 내재된 규칙성(Regularity)에 근거하여 추정 가능하며 시스템의 미래 행위는 과거와 크게 다르지 않다는 가정하에서 주어진 자료의 특성과

약(Characterization), 모형화(Modeling) 및 예측(Forecasting)의 세 단계를 거쳐서 이루어진다.

근대적 시계열분석 모형의 개발은 1927년 Yule이 태양 흑점의 연간변동을 예측하기 위하여 사용한 자기회귀(Autoregressive, AR)모형으로부터 비롯되었으며, 이후 반세기 동안 선형시스템에 의한 시계열분석이 주류를 이루어 왔다. 하지만, ARIMA모형[8]에 의해 대표되어지는 선형시스템에 의해 설명하기 어려운 현상들이 많다는 것을 인식하기 시작하면서 새로운 접근방법들이 개발되기 시작하였으며, 관련 연구들은 크게 두 가지로 분류할 수 있다. 첫 번째는 시계열의 생성과정이 확정적 모형에 의해 설명되어질 수 있다는 가정하에 동적 시스템이론에 근거하여 시간지연(time-delay)을 포함하는 상태공

[†] 본 연구는 한국과학재단의 연구비 지원으로 이루어졌음.
(과제번호 : 951-0905-115-2).

간(state space) 재구성에 의해 시계열 생성과정을 모형화 하려는 것이다[22]. 두 번째는 신경망으로 대표되는 기계학습(Machine Learning)분야의 발전에 영향을 받은 것으로 인공지능 연구의 관심이 규칙기반시스템으로부터 지식획득문제의 해결을 위한 자료중심의 분석방법으로 옮겨감에 따라 시계열분석에의 응용이 가능해졌다. 특히 1980년대말에 들어와서는 비선형모형에 대한 관심이 높아졌으며, 혼돈이론[9,17] 신경망[29,34, 38,42] 및 통계학[14,21,30]분야에서 개발된 다양한 기법들이 현실 세계의 비선형성을 효과적으로 다룰 수 있는 수단을 제공할 것으로 믿었다.

하지만, 시계열분석을 위한 새로운 접근방법들이 상당한 기대를 받았음에도 불구하고 예측성가에 있어서는 상당한 논란이 있어온 것을 부정할 수 없다. 이는 시계열분석이 다양한 학문분야에서 연구되고 있으며 새로운 예측알고리즘의 개발이 비교적 쉬운 반면 해당 알고리즘의 예측성과 및 기존 알고리즘과의 연관성에 대한 정확한 평가는 쉽지 않았기 때문이다. 다시 말해 관련 연구성과들을 정확히 평가할 수 있는 수단의 부족으로 인하여 관련 문헌의 방대함에도 불구하고 과거의 연구결과들을 바탕으로 새로운 연구를 시작하기 어려운 상황이었다. 이와 같은 상황을 인식하고 Makridakis et al[32, 33]은 시계열 데이터베이스를 구축한 후 예측의 정확성을 경쟁하는 대회를 처음 개최하였다. 이 대회는 마크리다키스 경쟁대회(Makridakis Competition)라고 불리며, 현재까지 시행된 세 차례의 대회를 구분하기 위해 M1, M2, M3라고 각각 부른다. 또, 신경망을 이용한 시계열분석 연구를 해 오던 Weigend도 Santa Fe 연구소와 NASA의 지원하에 유사한 성격의 대회를 개최하였다[43]. 두 대회의 공통적 특징은 구축된 시계열 데이터베이스에 대한 예측방법에 제한을 두지 않음으로써, 자연스럽게 여러 예측방법론들이 비교되도록 하여 시계열연구자들에게 연구결과의 객관적인 검증기회를 제공하였다는 점이다. 두 시계열 데이터베이스는 단일변수 시계열분석(Univariate Time Series Analysis) 연구에 있어 중요한 벤치마크 자료로 활용되고 있다.

본 연구의 목적은 시계열분석에 있어 신경망의 유용성을 검증하고자 하는 것이다. 신경망을 이용한 시계열분석이 새로운 유형의 연구는 아니지만 앞서 지적한 바와 같이 신경망의 예측성가에 대한 논란이 존재하므로 M2 경쟁자료를 사용하여 신경망의 예측성가를 객관적으로 평가를 하고자 한다. 과거 연구에서 많이 사용되었던 신경망모형은 다계층 퍼셉트론(Multi-Layer Perceptron, MLP)이었는데, 본 연구에서는 MLP의 대안으로 최근 관심을 모으고 있는 Cascade Correlation Net-

work(CCN) [16]을 사용한 시계열분석을 시도한다. 또, MLP를 자기연상 기억장치(Autoassociative Memory)로 사용하여 시계열로부터 잡음 또는 임의변동 요소들을 제거한 후 예측모형을 구성하는 신경망을 이용한 분해분석방법을 개발하였다. 신경망을 이용한 시계열분석방법들의 예측성가를 전통적 선형모형인 자기회귀-이동평균(ARIMA)모형과 비교평가하기 위하여 AUTOBOX V3.0[4]을 사용하였다.

본 논문의 구성은 연구에 사용된 시계열 데이터베이스인 M2 경쟁자료에 대해 설명한 후 본 연구의 예측모형인 신경망을 이용한 예측방법론들에 대한 소개 및 시계열예측을 위한 신경망의 구조결정 및 사용방법에 대해 설명하였다. 네 가지 예측방법론들의 예측성가는 단단계 예측값을 기본으로 평균 절대오차율(Mean Absolute Percentage Error, MAPE)을 사용하여 평가한 후, 앞으로의 연구방향을 제시하였다.

2. M2 경쟁자료

시계열분석을 위한 대표적인 벤치마크 데이터베이스로는 마크리다키스자료와 산타페자료를 들 수 있다. 산타페자료는 레이어 관련 실험자료, 수면중 무호흡증환자의 수면상태 분석자료, 환율변동자료, 백색왜성 관련 천문자료 및 바하의 푸가 등 5개의 실세계의 자료와 대회를 위해 인위적으로 생성한 시계열 하나 등 총 6개의 시계열로 구성되어 있으며, 각 시계열내 관측값의 수도 1000개에서 3만 개에 이를 정도로 충분한 관측값을 제공하고 있어 비선형모형의 안정적 추정이 가능하다.

이에 반해 마크리다키스 경쟁자료는 모두 사회경제 시스템으로부터 얻은 시계열들로 구성되어 있으며, 마크리다키스 경쟁대회의 목적은 다양한 단일변수 시계열예측방법론들의 예측성능을 비교분석하기 위한 것으로 첫번째 대회결과는 1982년에 발표되어 학계에서 많은 논란을 불러일으켰으며 예측방법론의 발전에도 큰 공헌을 하였다[32]. 흔히 M1 경쟁자료로 불리는 시계열 데이터베이스는 총 1001개의 시계열들로 구성되어 있어 너무 방대하므로 101개로 간추린 데이터베이스가 벤치마크데이터로 많이 사용되어왔다. M1 경쟁자료 중 101개의 시계열에 대해 신경망을 이용한 분석도 몇 차례 보고된 바 있다[27,38].

M2 경쟁대회는 M1 경쟁대회의 분석결과에 대한 비판을 감안하였다. 즉, M1이 재고관리나 일정관리분야처럼 짧은 시간에 많은 수의 예측값을 얻어야 하는 상황을 고려한 반면, M2에서는 예산작성과 같이 비교적 장기간의 예측값이 요구되며 전문가의 판단에 의한 예측값의 수정이 예측의 정확성을 높일

표 1. M2 경쟁자료의 분류

관측기간	관측값 수	시계열번호	관측값의 마지막 연월
분기별	167	m007, m008, m009, m010, m011, m012	1988. 3.분기
월별	82	m001, m002, m003, m004, m005, m006, m020, m021, m022, m023, m024, m025	1988. 10.
	94	m013, m014, m015, m016, m017, m018, m026, m027	1988. 10.
	45	m019	1988. 10.
	237	m028, m029	1988. 9.

수 있다는 주장을 반영하였다. 따라서, M2에서는 시계열의 수를 대폭축소하고, 5명의 전문가들을 직접 참여시켰다. 1993년에 발표된 M2의 주요결과[33]는 임의변동이 심한 시계열의 경우 정교한 예측방법이나 전문가의 판단에 의한 예측값의 보정이 지수평활법 등과 같은 단순한 예측방법이나 이들로부터 얻은 예측값들을 결합하여 사용하는 것보다 더 좋은 예측성과를 보이지 못했다는 것으로 M1의 결과와 일치하는 것이다. 최근 M3 대회가 열렸으나 아직 결과는 발표되지 않았다[39]. 본 연구에서 사용될 M2 경쟁자료는 미국내의 서로 다른 산업에 속해 있는 4개의 기업으로부터 수집된 23개의 월별 시계열 자료와 거시경제변수들에 관한 6개의 분기별 시계열자료를 합해 29개의 시계열로 구성되어 있는데, 관측기간과 관측값 수에 의해 정리한 것이 <표 1>이다. M2 경쟁에서는 월별자료의 경우 14 또는 15개의 예측값을, 분기별자료의 경우에는 5개의 예측값을 요구하였다.

3. 시계열예측을 위한 신경망모형

3.1. 신경망과 예측작업

신경망을 예측작업에 응용하는 연구는 신경망을 예측작업의 보조적인 수단으로 사용하는 방법과 신경망을 직접 예측함수로 이용하는 두 가지 접근방법으로 분류할 수 있다. 첫번째 접근방법은 다시 두 가지로 나누어지는데, 첫번째는 여러 예측모형에 의해 얻어진 예측값들을 결합하는 데 신경망을 사용하는 것으로 Foster et al[19]은 6개의 지수평활법에 의해 얻어진 예측값들을 결합하기 위한 가중값 결정에 MLP를 사용하였다. 두 번째는 주어진 시계열에 가장 적절한 예측모형을 선정하기 위하여 신경망을 사용하는 것으로 Chu & Widjaja[13]는 주어진 시계열의 데이터패턴을 인식한 후 6가지의 지수평활법 중에서 하나를 선정하는 예측모형 선정시스템을 신경망을 이용하

여 개발하였다. 또 사회경제분야의 예측에 많이 사용되어 왔던 Box-Jenkins 또는 ARMA모형을 사용함에 있어 모형선정의 어려움을 극복하기 위해 신경망을 패턴 분류기로 사용하여 ARMA모형수립을 자동화하려는 연구[24,25,26]도 같은 범주에 속한다.

두 번째 접근방법은 신경망을 직접 시계열의 생성과정에 대한 추정함수로 사용하는 것으로, Lapedes & Farber[29]가 백프로퍼게이션 알고리즘에 의해 학습된 MLP가 기관측값들에의 적합은 물론 예측에 있어서도 기존의 계량적 기법들보다 우수함을 최초로 보인 이후 여러 연구들에 의해 뒷받침되었다 [1,12,26,27,38,41,42,43]. 하지만, White[44], Fishwick[18] 및 Ripley[36] 등은 기존의 통계적 방법들보다 신경망이 더 좋은 예측성과를 보인다고 할 수 없다는 연구결과를 발표하였다. 이와 같이 엇갈리는 연구결과들에 대해 Charfield[10]는 신경망이 새로운 유력한 예측수단임에 틀림없지만 체계적인 실증적 분석이 필요함을 강조하였다.

최근까지 MLP는 시계열분석에 있어 표준적 신경망으로 사용되어 왔는데 이는 MLP가 비선형 회귀분석을 하는 것과 같은 효과가 있으며 만능근사기(Universal Approximator)라는 주장이 나올 만큼 주어진 자료에의 적합능력이 뛰어나며[6,23], 실제 세계의 시계열들 중 상당량은 비선형모형에 의해 효과적으로 분석될 수 있기 때문이다. 하지만, MLP의 예측성과에 대한 평가가 일치하지 않는 이유 중의 하나는 너무 많은 파라미터를 갖는 비선형모형의 경우 과잉적합(Overfitting)의 위험이 크다는 점이다.

특히, 사회경제 시스템으로부터 얻어진 시계열들은 관측값의 수가 적고 잡음이 많이 포함되어 있으므로 과잉적합의 문제를 해결하는 것이 매우 중요하다. 신경망의 과잉적합문제를 해결하기 위해 다음과 같은 방법들이 제시되어 왔다.

- 1) 학습된 MLP로부터 불필요한 가중값(Weight)들을 제거하는 방법[28,42].

- 2) 사전에 MLP내 일부 가중값들의 값을 고정시킴으로써 (Invariance의 도입) MLP내의 추정되어야 할 파라미터(Free Parameter)의 수를 줄이는 방법.
- 3) MLP가 주어진 학습자료에 대해 충분히 일반화될 수 있도록 효과적으로 학습자료를 확장하는 방법[7].
- 4) 시계열자료를 철저히 분석하여 MLP설계에 대한 아이디어를 얻는 방법.

첫번째 방법인 가중값 제거(Weight Elimination 또는 Pruning) 방법들은 catastrophic effect를 피하기가 쉽지 않고, 두 번째 방법인 신경망의 구조에 Invariance를 도입하는 방법은 사전에 명확한 신경망 설계지침이 존재하여야 하나, 현재까지 이에 대한 연구결과 부족과 MLP 내부의 구조변화로 인한 결과를 예상하기 어렵다는 점에서 현실적용성이 떨어진다. 또 세 번째 방법인 학습자료의 충분한 획득은 사회경제 시스템으로부터 얻어지는 대부분의 시계열들은 관측값의 수가 적고, 실험에 의한 추가 관측값의 생성도 불가능하므로 완전한 해결책이라고는 할 수 없다. 최근 Geman *et al*[20]은 MLP를 현실문제에 적용함에 있어 발생하는 문제를 편차/분산 딜레마(Bias/Variance-Dilemma)라고 표현하였다.

즉, 분산의 제어에 초점을 맞출 경우 추정된 모형의 편차가 커지고 편차의 제어에 초점을 맞출 경우 분산이 커지므로 양자간의 적절한 타협점을 찾는 것이 필요한데 MLP는 구조상 이와 같은 타협점을 찾기가 쉽지 않다는 것이다. Geman *et al*의 주장은 그 동안 제시된 MLP관련 학습이론들에 문제가 있는 것이 아니라, 주어진 문제를 MLP를 사용하여 표현하는 데 어려움이 있다는 것이다.

즉, 신경망을 이용한 문제해결에 있어서 적합한 신경망모

형의 선택 및 선택된 신경망의 적절한 구조 및 크기를 결정하는 것이 가장 중요한 문제라는 것이다. 시계열예측을 위해 리커런트 신경망모형을 이용하거나, ARMA모형 선정과정에서 얻은 결과를 신경망의 설계에 응용하는 연구들이나[26,27], 본 연구에서 새로운 구조의 신경망인 CCN을 사용하거나 시계열의 분해분석을 시도한 것은 모두 이러한 노력의 일환으로 볼 수 있다.

3.2. 다계층 퍼셉트론 (Multi-Layered Perceptron, MLP)

시계열 예측에 사용되는 MLP의 기본구조는 <그림 1>과 같다. <그림 1>의 MLP는 단단계 예측값(one-step-ahead forecast)을 얻기 위한 것으로 다단계(multi-step-ahead) 예측값을 원할 경우에는 출력노드의 간단한 변경으로 해결할 수 있다.

본 연구에서는 하나의 은닉층만을 갖는 MLP를 사용했는데 이는 Hornick *et al*[23]의 연구결과를 따른 것이다. MLP를 이용하여 좋은 예측성적을 얻기 위해서는 입력노드 및 은닉노드의 수를 적절히 결정하여야 하는데, 입력노드수의 결정은 MLP에만 국한되는 문제는 아니다.

하지만 은닉노드의 수를 결정하는 문제는 과거의 몇몇 연구에도 불구하고[5,11] 아직도 시행착오에 의존할 수밖에 없어 주어진 시계열에 적합한 MLP의 구조를 결정하기 위해서는 많은 시간과 노력이 필요하다.

3.3. Cascade Correlation Network (CCN)

MLP 사용의 어려움 즉, 은닉층의 노드수를 결정하는 데 따른 어려움을 해소하고자 Fahlman & Lebiere[16]에 의해 제시된

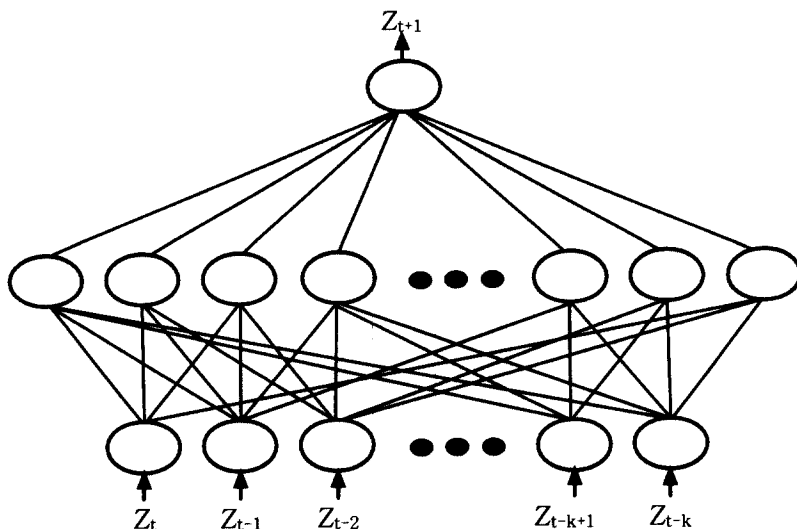


그림 1. 단단계 시계열예측을 위한 MLP의 구조.

CCN을 사용한다. Fahlman [15]은 백프로퍼게이션 알고리즘의 학습속도가 느린 원인으로 Step-size 결정문제와 Moving Target 문제의 두 가지를 지적하고 있다. Step-size 결정문제는 표준 백프로퍼게이션 학습에 있어 전역오차함수에 대해 1차 부분도함수만을 사용하여 가중치의 갱신량을 계산하려 하기 때문에 발생한다. 이 문제를 완화하기 위해 관성항을 사용하거나 2차도함수를 사용하는 방법들이 많이 제시되었다[40].

Moving Target 문제는 MLP의 경우 신경망의 구조가 고정되어 있고 변화하는 입출력 패턴에 대해 MLP 내부의 가중값들이 모두 동시에 변화하게 되므로 생기는 문제이다. 즉, 신경망 내부의 모든 노드들이 동시에 유용한 특성추출기(feature detector)가 되려 하기 때문에 오히려 학습속도를 늦추고 경우

에 따라서는 학습결과와 편차가 심해지는 현상을 보인다는 것이다. 따라서, CCN에서는 Step-size 결정문제는 Quickprop 학습방법에 의해 해결하고 Moving Target 문제에 대해서는 새로운 신경망 구조를 제시함으로써 백프로퍼게이션 학습방법에 의한 MLP의 한계를 벗어나고자 했다.

CCN의 구조는 <그림 2>와 같다. CCN의 학습은 은닉층의 노드들이 없는 상태에서 단순히 입출력 노드만을 가지고 시작한다. 학습을 진행시켜 나가면서 오차의 감소가 더 이상 진전되지 않는 시점에서 일단 학습을 멈추고 해당 CCN의 성능을 평가하여 학습결과에 만족하면 학습을 종료하고, 만약 학습결과에 만족하지 못할 경우에는 은닉층의 노드들을 하나씩 순차적으로 증가시켜 가면서 학습과정을 반복한다. CCN을 시계열

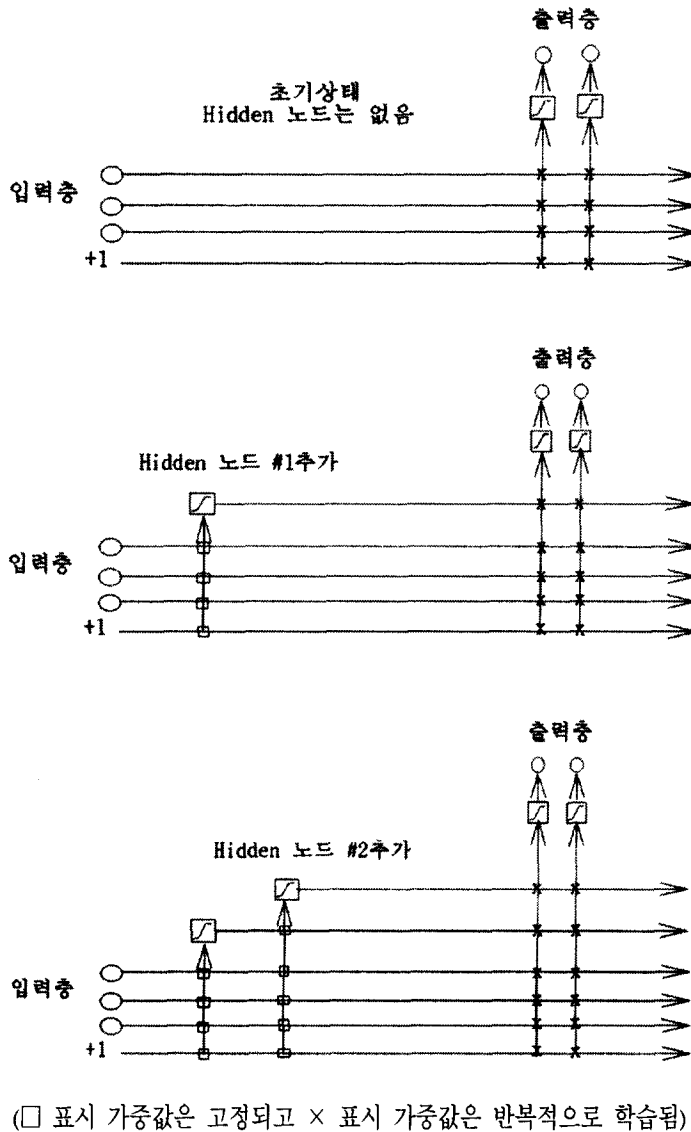


그림 2. Cascade Correlation Network의 구조 및 학습 과정.

예측에 이용한 연구로는 지원철 등[31]이 있다. 최근 CCN의 학습방법을 개선하고자 하는 연구가 많이 진행되고 있는데, 본 연구에서는 표준 학습방법을 사용하였다[35].

3.4. MLP에 의한 시계열의 분해

기존의 분해분석은 이동평균과 지수평활법 등을 사용하여 시계열을 계절성, 추세 및 임의변동으로 분리한 후, 각 요소들에 대해 개별적인 예측을 실시한 후 예측값들을 합하여 최종 예측값을 얻는 것이다. 본 연구에서도 전통적 분해분석과 같은 아이디어로부터 출발하였지만 시계열을 추세와 계절성을 합한 시계열과 잔차시계열로 분리한 후 각각에 대해 신경망을 사용하여 예측을 행한 후 최종 예측값을 얻도록 하였다. 시계열의 분리과정에서도 MLP를 사용하였는데 이를 Noise Filtering Network(NFN)이라고 하였는데, <그림 3>은 NFN의 구조를 나타낸 것이다. NFN에 의해 분리된 시계열들에 대해 다시 <그림 1>과 같이 MLP를 함수근사기로 사용하여 예측값을 얻었다.

다시 말해, 총 3개의 신경망을 이용하여 예측값을 얻는 방법을 DAUNN (Decomposition Analysis Using Neural Network)이라고 하였다. 이와 같이 신경망을 동일한 시계열에 대해 두 번 적용하여 예측값을 얻는 이유는 M1 경쟁자료를 분석한 연구 [27]에 의하면, 관측값의 수가 적고 잡음이 심한 시계열의 경우 해당 시계열이 추세나 계절성을 가지면 MLP의 예측성능이 기대만큼 발휘되지 못한다는 사실에 근거한 것으로, M2에서도 계절성의 분리가 예측성과를 높이는 데 기여한다는 분석결

과가 보고되었다.

MLP를 NFN으로 사용하기 위해 신경망의 자기연상기억기능을 이용하였다[37]. 즉, NFN의 입출력층에 같은 수의 노드를 사용하고 입출력이 같은 값을 갖도록 학습자료를 구성하는 것이다. 동시에 은닉층에는 입출력층보다 적은 수의 노드만을 사용함으로써 NFN은 시계열내에 존재하는 잡음들을 걸러내고 추세나 계절성과 같이 주어진 시계열의 주요특징만을 기억하게 된다. 주어진 시계열에 대해 Sliding Window 기법에 의해 NFN을 학습시킬 경우 NFN의 출력은 한 시점의 관측값에 대해 출력층의 노드수만큼의 출력값들을 가지게 되므로 다시 이들의 평균값을 구하여 해당 시점의 최종 출력값으로 정하였다. 시계열 $\{Z_t, t \in T\}$ 가 주어졌을 때 신경망을 이용하여 두 개의 시계열, 즉 추세와 계절성 등 시계열의 주요특징만을 추출한 추세시계열 ZTS와 잔차시계열 TRS로 분리하는 분해분석에 의한 예측과정은 다음과 같다.

단계 1 : NFN의 구조결정

- (1.1) NFN에의 입력패턴의 크기, 즉 슬라이딩 윈도우의 크기 k 를 정하고, NFN의 입출력층의 노드수를 모두 k 로 하고 은닉층의 노드수는 k 보다 적은 수로 정한다.

단계 2 : NFN의 훈련

- (2.1) 사전에 정의한 학습횟수(Epochs)만큼 (2.2)를 반복한다.
- (2.2) for $i = 1, \dots, T-k$ 에 대해 다음을 반복한다.
 - (2.2.1) $\{Z_i, Z_{i+1}, \dots, Z_{i+k-1}\}$ 를 NFN의 입력 및 출력값으로 동시에 대응시킨다.
 - (2.2.2) 백프로퍼게이션 알고리즘에 의해 NFN의 가중값들을 수정한다.

단계 3 : 시계열의 분리

- (3.1) 학습된 NFN으로부터 출력값들을 얻는다. i 번째 윈도우에 의한 NFN의 출력값을 $\{Z_i^i, Z_{i+1}^i, \dots, Z_{i+k-1}^i\}$ 으로 정의한다.
- (3.2) $\bar{Z}_i = \frac{1}{k} \sum_{j=1}^k Z_{i+j-1}^i$, for $i = 1, \dots, T$ 를 계산하여 잡음이 제거된 시계열 $Z^{TS} = \{\bar{Z}_1, \bar{Z}_2, \dots, \bar{Z}_T\}$ 를 얻는다. 이때 NFN의 출력값들이 k 개 미만인 경우에는 해당 출력값들만으로 평균값을 얻는다.
- (3.3) $\{Z^{RS}\}$ 는 원래 시계열 $\{Z\}$ 로부터 $\{Z^{TS}\}$ 의 값들을 차감하여 얻는다.

단계 4 : 분리된 시계열의 모형화 및 예측

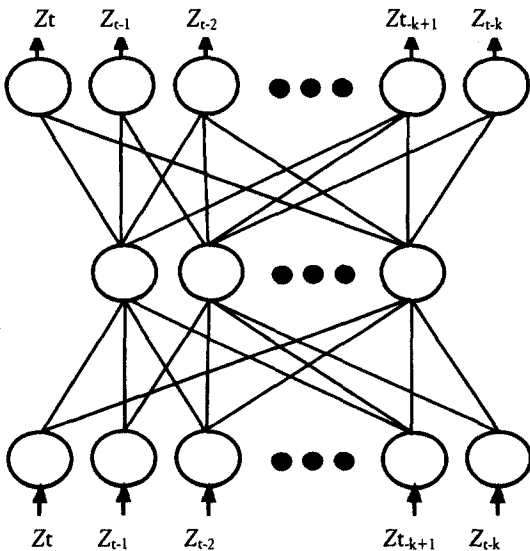


그림 3. Noise Filtering Network의 구조.

- (4.1) $\{Z^{TS}\}$ 에 대해 <그림 1>과 같은 MLP를 사용하여 신경망의 학습 및 예측작업을 실시하여 $\{\hat{Z}_t^{TS}, t \in T\}$ 을 얻는다.
- (4.2) $\{\hat{Z}_t^{RS}\}$ 에 대해서도 (4.1)과 같은 작업을 반복하여 $\{\hat{Z}_t^{RS}, t \in T\}$ 를 얻는다.
- (4.3) 주어진 시계열에 대한 예측값 $\{\hat{Z}_t, t \in T\}$ 는 $\{\hat{Z}_t^{RS} + \hat{Z}_t^{TS}, t \in T\}$ 로 계산한다.

<그림 4>는 M2 경쟁자료 중 자동차회사로부터 얻은 시계열(m006)에 대해 NFN을 적용하여 추세시계열과 잔차시계열로 분리한 것을 나타낸 것으로 추세시계열은 해당 시계열내의 추세 및 계절성이 합쳐진 것으로 볼 수 있다. <그림 4(a)>에서 NFN에 의해 분리된 추세시계열은 기존의 분해분석에 의한 추세나 계절성을 분리할 경우 원래의 시계열에 비해 상당한 시차를 가지고 추세나 계절성이 파악되는 단점이 보완된 것을 보여 준다. 또 <그림 4(b)>에서 잔차시계열들은 추세나 계절성이 거의 나타나지 않음을 확인할 수 있다.

4. 예측성과의 비교분석

4.1. 실험조건의 설정

주어진 시계열 $\{Z_t, t \in T\}$ 에 대한 예측값을 얻기 위해 반복적으로 단단계 예측값 (iterative one-step-ahead forecast)를 얻는 절차를 사용하였다. 즉, 과거의 관측값 $\{Z_t, t \in T\}$ 을 이용하여 시계열의 생성과정을 모형화한 예측함수 f 를 추정한 후, f 를 이용하여 예측기간 L 동안의 예측값들 $\{\hat{Z}_{t+r}, r \in L\}$ 을 차례로 얻어 가는 것으로 수식으로 표현하면 다음과 같다.

$$\begin{aligned} \hat{Z}_{t+1} &= f(Z_t, Z_{t-1}, \dots) \\ \hat{Z}_{t+2} &= f(\hat{Z}_{t+1}, Z_t, Z_{t-1}, \dots) \\ &\vdots \\ \hat{Z}_{t+r} &= f(\hat{Z}_{t+r-1}, \hat{Z}_{t+r-2}, \dots, \\ &\quad \hat{Z}_{t+1}, Z_t, Z_{t-1}, \dots) \end{aligned}$$

본 연구에서 예측함수 f 를 추정하기 위한 사용된 신경망 모형들은 MLP, CCN 및 신경망을 이용한 분해분석(DAUNN) 등이며, 신경망의 예측성과를 비교하기 위한 기본모형으로 ARIMA가 사용되었다. 신경망의 예측성과를 결정하는 요인으로는 신경망의 구조, 학습방법 및 학습 파라미터들과 시계열

의 생성과정 모형화에 사용될 입력공간의 크기, 즉 신경망 입력노드의 수 등 다양하다. 특히 신경망의 과잉적합 위험을 회피하기 위해서는 적절한 입력공간의 크기를 결정하는 것이 중요하기 때문에, 본 연구에서는 AUTOBOX 3.0을 이용한 ARIMA모형화의 결과를 사용하였는데 주어진 시계열이 ARIMA(p, d, q)를 따를 경우 $Max(p+d, q+1)$ 를 신경망 입력노드의 수로 사용하였다[2,27]. 신경망의 은닉노드수 결정문제는 CCN의 경우 자동으로 결정되지만, MLP의 경우에는 사용자가 최종 결정하여야 한다. <그림 1>과 같이 MLP를 바로 예측함수로 사용할 경우 은닉층에 입력노드의 수와 같은 수의 노드를 사용하였다. DAUNN의 NFN의 경우 은닉층에는 입력층의 적은 수의 노드들이 사용되어야 하므로 해당 시계열이 월별자료인 경우에는 12-4-12를 분기별자료인 경우에는 4-2-4를 입력, 은닉 및 출력층에 각기 사용하였다. NFN에 의해 두 개로 분리된 시계열들에 대해서는 각기 앞서 설명한 입력층 및 은닉층의 노드수 결정방법을 사용하였다.

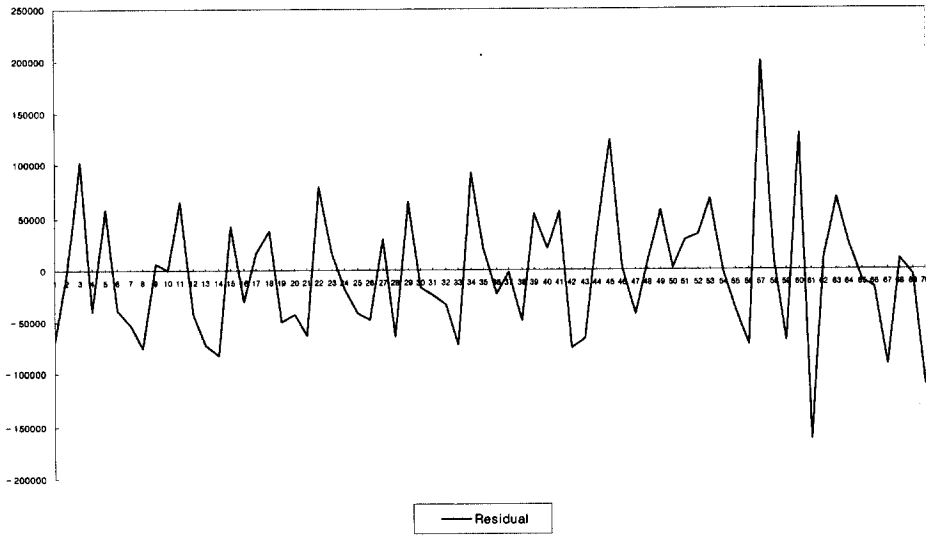
본 연구에서 사용된 모든 MLP들은 하나의 은닉층만을 가지고 있으며 은닉층과 출력층의 전이함수로는 시그모이드함수가 사용되었으며, 신경망의 학습을 위해서 시계열들은 (0, 1) 사이의 값들을 갖도록 선형변환하였다. 시계열들의 일부를 신경망의 학습에서 제외하여 학습된 신경망의 예측력을 검증하기 위해 사용하였는데, 마지막 1년간의 관측값들을 검증자료로 사용하였다. 따라서, 월별자료는 12개, 분기자료는 4개의 자료가 신경망 학습에서 유보되었다. 학습멈춤 규칙은 MLP의 경우 최대 학습횟수가 50,000 Epochs에 이르거나 MLP의 RMSE (Root Mean Squared Error)가 0.01 보다 작을 경우가 사용되었으며, CCN의 경우에는 CCN의 학습시 학습정도의 파악을 위해 많이 사용되는 EI(Error Index)의 값이 0.05에 이르면 학습을 멈추었다. N_o 를 출력노드의 수, N_p 를 학습패턴의 수, \hat{Z}_t 를 CCN의 출력값, 즉 예측값이라고 할 때 EI는 식 (1)과 같이 계산된다. 또, MLP와 CCN 모두 RMSE나 EI가 100 Epochs 동안 변화가 없는 경우 학습을 멈추었으며 100 Epochs마다 학습상태를 평가하여, 학습에 따른 예측성능의 변화를 추적하였다.

ErrorIndex

$$= \sqrt{\frac{\sum (\hat{Z}_t - Z_t)^2}{(N_o \times N_p) \times \sqrt{N_o \times \sum \hat{Z}_t^2 - (\sum \hat{Z}_t)^2}}} \quad (1)$$

신경망의 출력값들은 학습자료의 생성에 사용되었던 선형변환의 역으로 재변환하여 시계열의 실제값들과 비교되었다. 예측성과를 측정하기 위하여 MAPE(Mean Absolute Percentage Error)가 사용되었는데, 예를 들어 예측기간 L 동안 얻어진 예

(b) NFN에 의한 잔차시계열



(a) 원시계열과 추세시계열의 비교

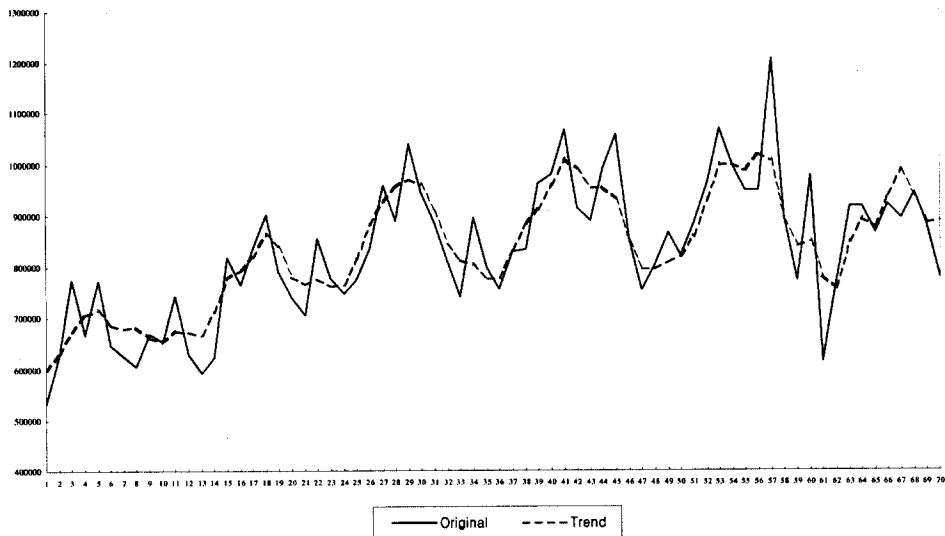


그림 4. Noise Filtering Network의 적용예 (CARINDS: m006).

측값의 정확도를 측정하기 위한 MAPE의 계산은 식 (2)와 같다.

$$MAPE = \frac{1}{L} \sum_{t=1}^L \frac{|\hat{Z}_{t+r} - Z_{t+r}|}{Z_{t+r}} \times 100 \quad (2)$$

본 연구에서 월별시계열은 $L=12$, 분기별시계열은 $L=4$ 가 사용되었다. 이는 Makridakis et al[33]의 분석에 사용되었던 예측기간과 차이가 있는 것으로, 확보한 M2 자료의 한계로 인한 것이다. 즉, M2에서는 월별자료의 경우 1987년 11월부터 1988년 12월까지와 1988년 11월부터 1989년 12월까지 두 번에 걸쳐서 14개의 예측값을 작성하도록 하였으나 본 연구에서

<표 1>에서의와 같이 1988년 10월까지의 자료들만 이용이 가능하였다. 따라서 본 연구에서는 1987년 10월까지의 자료는 신경망의 학습에 사용하였고, 나머지 일년 동안의 자료들에 대해 예측값을 구한 후 신경망의 예측성적을 분석하였다.

4.2. 실험결과

M2 경쟁자료에 대해 ARIMA, MLP 및 CCN의 예측성적을 비

교한 것이 <표 2>이다.

신경망의 학습결과인 시계열에의 적합도 및 검증자료에 대한 신경망의 예측성결과를 MAPE에 의해 측정한 상세 결과는 <부록 1>을 참조하기 바란다. <부록 1>과 <표 2>에서 괄호 안의 숫자는 m015(시계열명 : INTERSAL)의 경우 MAPE값이 학습과 예측에서 모두 이상적으로 매우 큰 값을 가지므로 이를 제외하고 평균과 표준편차를 구한 것이다.

<표 2>에서 신경망의 학습능력, 즉 시계열에의 적합도는 선형모형인 ARIMA에 비해 우수함을 알 수 있다. MLP와 ARIMA 사이에는 큰 차이가 없었지만 CCN의 적합능력은 매우 뛰어나다. 하지만 예측능력에 있어서는 MLP가 가장 좋았지만 통계적으로 유의한 결과는 아니었다. CCN이 시계열에의 뛰어난 적합능력을 보였음에도 예측성능에 있어서는 세 모형 중에서 가장 나빴다. 본 연구에서는 CCN의 학습을 EI에 의해 일률적으로 제어함으로써 시계열에 따라 과잉적합 문제가 발생한 것으로 추정된다.

이는 <표 3>에 나타낸 CCN의 학습결과 생성된 은닉노드의 수를 살펴보면 지나치게 많은 수의 은닉노드가 생성된 경우들을 볼 수 있는데 이들은 과잉적합 되었을 가능성이 크다. 이와 같이 CCN의 학습과정은 MLP의 경우보다도 제어하기가 쉽지 않지만 예측결과들의 표준편차는 ARIMA에 비해 작으므로 앞으로 계속적 연구가 수행될 경우 안정적인 예측모형으로 사용가능하다고 판단된다.

M2 경쟁자료 전체에 대해 신경망에 의한 시계열 분해분석을 적용하기에 앞서 몇 가지 대안들을 검토하였다. 다시 말해,

NFN에 의해 분리된 추세와 잔차시계열들에 대해 예측함수로 사용된 모형과 예측결과를 합치는 과정에서 어떤 모형의 결과를 사용하는가 하는 것이다.

<그림 5>에서 ARIMA는 NFN에 의해 분리된 추세 및 잔차 시계열에 대해 AUTOBOX 3.0을 사용하여 모형화한 후 예측결과를 얻은 것이며 나머지는 모두 MLP를 사용한 것이다. MLPBEST는 추세와 잔차시계열에 대해 각각 가장 적합이 잘된 MLP들을 사용한 것이고, MLPSUM은 분리된 두 시계열을 합쳐서 가장 적합이 잘 된 MLP들을 사용하여 예측한 것이다. MLPBEST와 MLPSUM 뒤의 숫자 1은 은닉층과 입력층의 노드수를 같게 사용한 것이고, 2는 은닉층의 수를 두 배로 한 것이다. 검토결과 은닉층과 입력층의 노드수를 같게 사용한 MLPSUM1방식을 채택하였다.

<표 4>는 NFN에 의해 분리된 시계열들에 대해 ARIMA 모형을 사용한 경우와 MLP를 사용한 경우(DAUNN)를 비교한 것이다. DAUNN의 예측결과를 얻기 위해 사용된 추세 및 잔차 시계열에 대한 MLP의 구조는 <부록 2>를 참조하기 바란다. DAUNN의 예측성능은 <표 3>의 결과와 함께 비교해 보아도 가장 우수하였으며, ARIMA의 경우도 분리된 시계열에 적용했을 경우 상당한 예측성능의 개선이 이루어짐을 볼 수 있다. 이는 NFN을 사용하여 시계열을 분리함으로써 시계열내의 잡음들을 잔차 시계열에 국한시켜서 예측모형의 과잉적합 가능성을 크게 줄인 결과이다. <표 4>에서도 괄호 안의 숫자는 m015의 예측결과를 제외한 것이다.

표 2. M2 경쟁자료에 대한 신경망의 예측성결과(MAPE)

	ARIMA모형		MLP		CCN	
	Fitting	Forecasting	Fitting	Forecasting	Fitting	Forecasting
평균	12.94 (10.36)	19.98 (14.08)	12.58 (9.90)	17.68 (13.02)	2.53 (2.40)	18.48 (15.00)
표준편차	15.96 (7.75)	34.43 (13.39)	16.39 (8.00)	26.84 (9.81)	2.15 (2.08)	21.72 (11.17)

표 3. CCN에 의해 생성된 은닉노드수

시계열번호	은닉노드수	시계열번호	은닉노드수	시계열번호	은닉노드수	시계열번호	은닉노드수	시계열번호	은닉노드수
m001	21	m007	5	m013	14	m019	18	m025	14
m002	33	m008	2	m014	5	m020	17	m026	12
m003	10	m009	5	m015	29	m021	23	m027	11
m004	13	m010	28	m016	13	m022	21	m028	35
m005	11	m011	2	m017	15	m023	33	m29	21
m006	14	m012	41	m018	23	m024	9		

5. 결 론

본 연구에서는 시계열 분석에 신경망을 활용하는 방법론들에 대해 정리한 후, 두 번째 마크리다키스(M2) 경쟁자료를 이용하여 신경망의 예측성적을 분석하였다. 다계층 퍼셉트론(MLP), Cascade Correlation Network(CCN) 및 본 연구에서 제안한 신경망에 의한 시계열의 분해분석 등 세 가지 신경망모형의 예측성적을 ARIMA모형의 경우와 비교분석하였다.

MLP와 CCN을 ARIMA와 비교했을 경우, 신경망들이 학습자료에 대한 적합능력에서는 매우 뛰어났지만, 학습에 주어지지 않은 시계열 관측값에 대한 예측에서는 유의한 차이를 보이지는 못했다. 이는 마크리다키스 경쟁의 분석결과와 일치하는 것으로 볼 수 있다. 즉, 잡음이 심한 시계열의 경우에는 정교한 예측방법들이 지수평활법과 같은 단순한 예측방법보다 더 좋은 예측성적을 보이지 못한다는 것이다. 따라서, 본 연구에서는 잡음이 많고 관측값의 수가 적은 시계열에 대해서도 신경

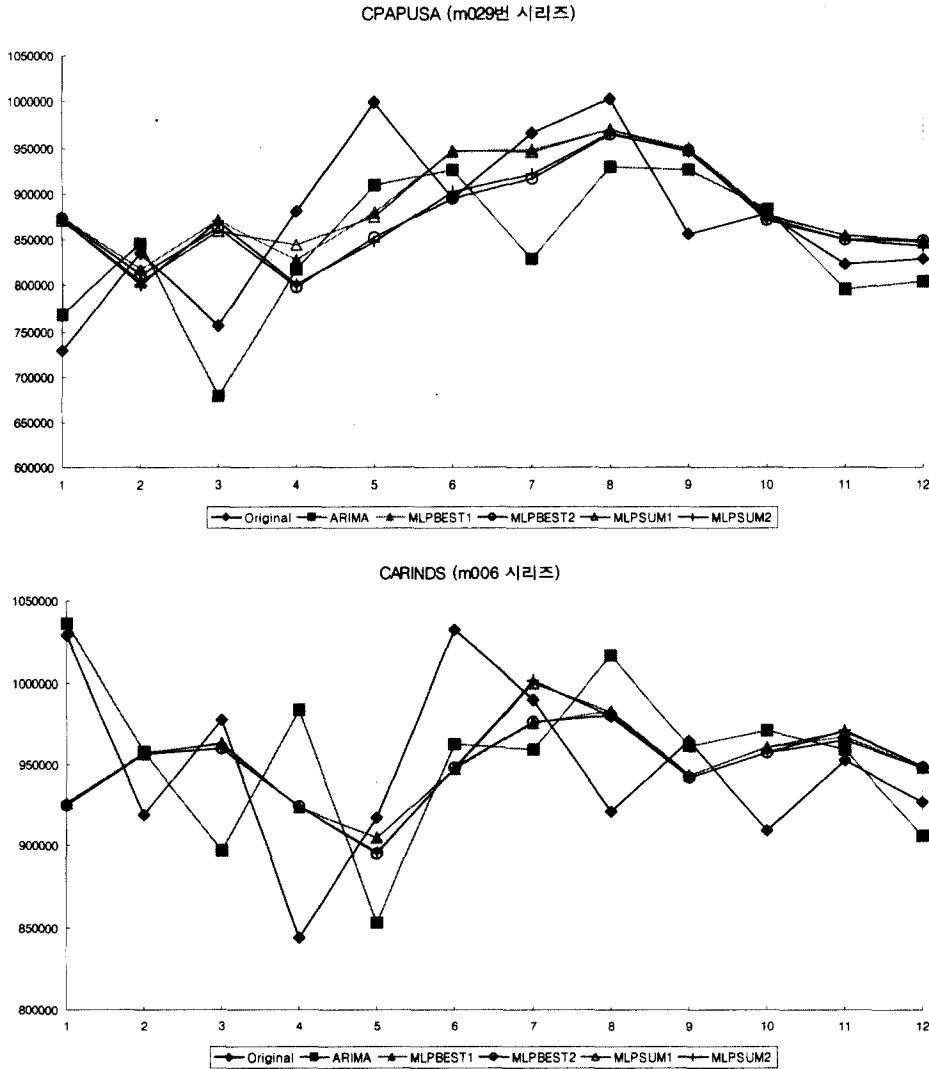


그림 5. 분해분석을 위한 대안들의 예측결과.

표 4. 신경망에 의한 분해분석결과(MAPE)

	ARIMA모형		DAUNN	
	Filling	Forecastomg	Fottomg	Forecastomg
평균	17.53 (14.64)	16.71 (13.28)	9.84 (6.64)	14.81 (10.77)
표준편차	17.24 (9.17)	24.31 (13.42)	12.46 (6.66)	21.38(8.81)

망을 이용한 시계열분석이 가능하도록 신경망을 이용한 시계열의 분해분석을 개발하였다. 신경망의 자기연상기억기능을 활용하여 주어진 시계열을 추세와 계절성만을 갖는 시계열과 잔차시계열로 분리함으로써 시계열내 임의변동으로 인한 영향을 잔차시계열로 한정하였다. 특히, 신경망을 이용한 분해분석은 전통적 분해분석방법에 비해 추세 및 계절성의 분리가 우수하였고, 이는 본 연구에서 비교 분석한 예측모형 중에서 가장 좋은 예측성과를 가져왔다. 또, 신경망에 의해 분해된 시계열에 대해서 ARIMA모형을 사용하여 예측한 경우에도 분해하기 전보다 상당한 예측성과의 개선이 있었다.

이상과 같이 신경망에 의한 분해분석이 가장 좋은 예측성과를 보인 것은 마크리다키스 경쟁자료와 같이 잡음이 심한 학습자료를 사용할 경우에는 신경망의 설계가 중요하다는 사실을 확인시킨 것이다. 이런 의미에서 보면 MLP와 CCN의 예측성과가 ARIMA에 비해 차이가 없었던 것은 주어진 시계열에 대한 철저한 분석을 바탕으로 신경망을 설계하지 않고 본 연구에서 설정한 방법을 모든 시계열에 일괄 적용한데도 원인이 있는 것으로 유추되며, 예측값들의 MAPE가 이상적으로 큰 시계열들에 대한 신경망의 설계를 바꿈으로써 예측성과의 개선 여지는 많다고 판단된다.

특히, CCN을 예측작업에 이용할 경우에도 CCN의 설계에 주의를 한다면 짧은 학습시간, 은닉노드수 결정의 불필요 등 CCN의 장점을 살릴 수 있다. 또, 최근 CCN에 관한 많은 연구가 이루어지고 있으므로 CCN의 과잉적합을 적절히 제어할 수 있는 단안이 개발될 것으로 기대된다.

마지막으로 앞으로의 연구방향에 대해 언급한다면 첫째 국내에서도 마크리다키스나 산타페자료와 같은 시계열 데이터베이스를 연구에 많이 활용할 필요가 있으며, 가능하다면 국내 자료에 의한 데이터베이스를 구축하여 보다 국내 상황에 적합한 시계열 분석방법을 찾아내는 연구를 할 수 있도록 하여야 한다. 두 번째는 신경망에 대한 분해분석과 관련한 것으로, 본 연구에서는 MLP를 자기연상기억장치로 사용하였으나, Hopfield Network이나 BAM 등과 같은 다른 신경망모형의 성능을 비교분석할 필요성이 있다.

또, MLP를 NFN으로 사용할 경우에도 입력 원도의 크기 및 학습횟수 등에 변화를 주는 추가적인 실험을 통해 보다 효율적인 NFN으로써 작동할 수 있도록 해 주는 학습 파라미터를 찾는 것이 필요하다.

마지막으로, 분해된 시계열에 대한 모형화 역시 본 연구에서는 MLP를 사용하였으나, 잔차시계열에 대해서는 마크리다키스의 연구결과와 같이 지수평활법을 사용한 결과와 비교분

석할 필요가 있다.

참고문헌

1. 박종엽, 한인구, "인공신경망을 이용한 종합주가지수 예측," 한국전문가시스템 '95 추계학술대회논문집, pp. 359-371, 1995.
2. 지원철, "신경망을 이용한 시계열분석: M1~Competition Data에 대한 예측성과 분석," 한국전문가시스템 학회지, Vol. 1, No. 1, pp. 135-148, 1995.
3. 지원철, 박시우, 신현정, 신홍섭, "Cascade-Correlation Network를 이용한 종합주가지수 예측," 한국경영과학회 대한산업공학회 '96 춘계 공동학술대회 논문집, pp. 745-748, 1996.
4. AFA Inc., *AUTOBOX 3.0 Manual*, Harboro, PA. 1991.
5. Ash, T. *Dynamic Node Creation in Back-propagation Networks, TR-8901*, Institute for Cognitive Science, UC San Diego, 1989.
6. Asoh, H. and Otsu, N., "Nonlinear data analysis and multilayer perceptrons," *Proceedings of IJCNN*, Vol. II, pp. 411-415, 1989.
7. Baum E. B. and Haussler, D., "What size net gives valid generalization," *Neural Computation*, Vol. 1, pp. 151-160, 1989.
8. Box, G. E. P. and Jenkins, G. M., *Time Series Analysis - Forecasting*, San Francisco: Holden-Day, 1976.
9. Casdagli, M., Eubank, S., Farmer, J.D. and Gibson, "J. state space reconstruction in the presence of noise," *Physica D*, Vol. 51, pp. 52-98, 1991.
10. Chatfield, C., "Neural networks: Forecasting breakthrough or passing fad?," *Int'l Journal of Forecasting*, Vol. 9, pp. 1-3, 1993.
11. Chauvin, Y., "A backpropagation algorithm with optimal use of hidden units," *In Advances in Neural Network Information Processing Systems*, Vol. 1, pp. 519-526, San Mateo CA: Morgan Kaufmann, 1989.
12. Chen, C. H., "Neural networks for financial market prediction," *Proc. of the 1994 IEEE Int'l Conf. on Neural Networks*, pp. 1199-1202, 1994.
13. Chu, C. H. and Widjaja, D., "Neural network system for forecasting method selection", *Decision Support Systems*, Vol. 12, pp. 13-24, 1994.
14. De Gooijer, J. G. and Kumar, K., "Some recent developments in non-linear time series modelling, testing and forecasting," *Int'l Journal of Forecasting*, Vol. 8, pp. 135-156, 1992.
15. Fahlman, S. E., *An Empirical Study of Learning Speed in Back-Propagation Networks*, CMU-CS-88-162, Pittsburgh, School of Computer Science, CMU, 1988.
16. Fahlman, S. E. and Lebiere, C., *The Cascade-Correlation Learning Architecture*, CMU-CS-90-100, Pittsburgh, School of Computer Science, Carnegie Mellon University, 1990.
17. Farmer, J. D. and Sidorowich, J. J., "Predicting chaotic time series," *Physical Review Letter*, Vol. 59, No. 8, pp. 845-848, 1987.
18. Fishwick, P. A. "Neural network models in simulation: A

- comparison with traditional modeling approaches," *Proc. of the 1989 Winter Simulation Conference*, pp. 702-710, 1989.
19. Foster, W. R., Collopy, F. and Ungar, L. H., "Neural network forecasting of short noisy time series," *Computer and Chemical Engineering*, Vol. 16, pp. 293-297, 1992.
 20. Geman, S. E., Bienenstock, E. and Doursat, R., "Neural networks and bias/variance dilemma," *Neural Computation*, Vol. 4, pp. 1-58, 1992.
 21. Granger, C. W. J. and Terasvirta, T., *Modeling Nonlinear Economic Relationships*. Oxford: Oxford University Press, 1992.
 22. Harvey, A. C., "A unified view of statistical forecasting procedures," *Journal of Forecasting*, Vol. 3, pp. 245-275, 1984.
 23. Hornik, K., Stinchcombe, M. and White, H., "Multilayer feedforward networks are universal approximators," *Neural Networks*, Vol. 2, pp. 359-366, 1989.
 24. Jhee, W. C. and Ro, H. B., "Decision support for ARMA model identification using hierarchically organized neural networks," *Proceedings of IEEE Int'l Conf. on Systems, Man, and Cybernetics*, pp. 1637-1644, 1991.
 25. Jhee, W. C., Lee, J. K. and Lee, K. C., "A neural network approach for the identification of Box-Jenkins model," *Network: Computation in Neural Systems*, Vol. 3, No. 3, pp. 323-339, 1992.
 26. Jhee, W. C. and Lee, J. K., "Performance of neural networks in managerial forecasting," *Int'l J. of Intelligent Systems in Accounting Finance & Management*, Vol. 2, No. 1, pp. 55-71, 1993.
 27. Jhee, W. C. and Shaw, M., "Time series prediction using minimally-structured neural networks: An empirical test," Trippi & Turban, (ed.) *Neural Networks in Finance and Investing*, 2nd, 1996. Also appears in *Proc. of World Congress on Neural Networks, 1994 Int'l Neural network Society Annual Meeting*, San Diego, CA., pp. 266-271.
 28. Karnin, E. D., "A simple procedure for pruning back-propagation trained neural networks," *IEEE Transactions on Neural Networks*, Vol. 1, No. 2, pp. 239-242, 1990.
 29. Lapedes, A. and Farber, R. *Nonlinear Signal Processing Using Neural Networks: Prediction and System Modeling*, Los Alamos National Laboratory Report, LA-UR-87 ~2662, 1987.
 30. Lebaron, B., "Nonlinear dynamics and stock returns," *Journal of Business*, Vol. 62, No. 3, pp. 311-337, 1989.
 31. Lee, J. K. and Jhee, W. C., "Two-staged neural network approach for the ARA model identification," *Decision Support Systems*, Vol. 11, No. 5, pp. 461-479, 1994.
 32. Makridakis, S, et al., "The accuracy of extrapolation (time series) methods: Results of a forecasting competition," *Journal of forecasting*, Vol. 1, No. 2, pp. 259-312, 1982.
 33. Makridakis, S, et al., "The M2-Competition: A real time judgmentally based forecasting study," *Int'l Journal of forecasting*, Vol. 9, pp. 5-22, 1993.
 34. Moody, J. and Darken, C., "Fast learning in networks of locally tuned processing units," *Neural Computations*, Vol. 1, No. 2, pp. 281-294, 1989.
 35. Prechelt, L., "Investigation of the cascor family of learning algorithm," *Neural Networks*, Vol. 10, No. 5, pp. 885-896, 1997.
 36. Ripley, B. D., "Statistical aspects of neural networks," *Proceedings of 1993 SemStat*, Chapman & Hall, London, 1993.
 37. Rojas, R., *Neural Networks: A Systematic Introduction*, Springer-Verlag, Berlin, 1996.
 38. Sharda, R. and Patil, R. B., "A connectionist approach to time series prediction: An empirical test," *Journal of Intelligent Manufacturing*, 1992.
 39. Stewart, G. A., "Prophetable software," *Byte*, April, pp. 97-101, 1998.
 40. Wasserman, P. D., *Advanced Methods in Neural Computing*, VNR, New York, 1993.
 41. Werbos, P. J., "Generalization of backpropagation with application to a recurrent gas market model," *Neural Networks*, Vol. 1, No. 4, pp. 339-356, 1988.
 42. Weigend, A. S., Rumelhart, D. E., and Huberman, B. A. "Backpropagation, weight-elimination and time series prediction," *In Touretzky (eds.) Advances in Neural information Processing Systems*, pp. 105-116, 1990.
 43. Weigend, A. S. and Gershenfeld, N. A., *Time Series Prediction: Forecasting the Future and Understanding the Past*. Proceedings Volume XV in the Santa Fe Institute Studies in the Sciences of Complexity, Addison-Wesley Publishing Company, 1994.
 44. White, H., "Economic prediction using neural networks: the case of IBM daily stock returns," *Int'l Joint Conference on Neural Networks*, pp. 451-458, 1988.

부록 1. M2 경쟁자료 예측모형들의 분석결과 (MAPE)

	ARIMA		MLP		CCN		DAUNN	
	Fitting	Forecastin g	Fitting	Forecastin g	Fitting	Forecastin g	Fitting	Forecastin g
1	16.78	10.65	16.27	19.73	1.86	12.94	5.14	5.97
2	21.2	16.9	19.07	14.79	2.34	14.81	18.69	10.99
3	12.17	17.6	0.92	13.43	0.73	14.72	2.89	12.94
4	8.98	12.81	9.47	11.35	0.71	14.89	3.6	8.95
5	8.63	10.14	9.37	11.19	0.63	14.5	3.14	5.88
6	8.77	6.01	4.36	8.09	0.49	13.19	2.19	3.04
7	0.79	1.04	1.59	4.85	1.01	2.6	0.54	4.52
8	0.45	0.37	2.89	5.43	1.82	4.34	7.49	2.73
9	1.24	0.85	1.84	1.13	2.15	0.9	0.79	1.61
10	1.95	2.04	2.07	5.41	1.68	3.64	0.87	5.91
11	0.38	0.78	4.92	6.3	2.07	5.49	1.12	2.7
12	10.01	13.4	10.05	12.44	6.39	15.61	5.41	8.01
13	25.12	25.51	23.37	19.24	4.31	27.85	16.12	15.64
14	18.31	24.26	11.68	36.84	7.91	38.64	23.34	32.25
15	85.89	185.43	87.37	147.96	6.15	116	99.61	127.77
16	17.61	17.66	16.47	15.37	2.37	18.9	18.34	13.23
17	8.93	8.42	8.49	9.35	1.4	14.74	11.72	9.13
18	4.64	8.46	5.15	6.43	0.86	7.6	1.58	5.3
19	25.39	48.02	24.17	25.54	5.13	37.78	17.37	30.84
20	17.8	32.36	12.44	35.18	3.39	24.99	9.15	20.86
21	17.34	36.42	22.48	17.39	4.7	25.97	8.4	24.08
22	17.38	37.3	21.38	27.96	5.88	38.88	6.22	22.79
23	17.56	35.99	25.64	26.93	4.75	26.63	11.68	22.74
24	2.38	2.52	1.89	2.95	0.5	4.3	0.77	3.04
25	2.7	5.82	3.32	8.7	0.63	9.72	1.7	4.94
26	5.75	4.67	3.54	4.77	0.65	8.97	2.24	4.5
27	6.87	5.5	4.61	5.31	0.92	7.61	2.16	4.27
28	5.48	3.14	5.31	4.3	1.06	4.13	1.68	8.74
29	4.77	5.59	4.65	4.28	0.9	5.72	1.52	6.01
평균	12.94 (10.36)	19.98 (14.08)	12.58 (9.90)	17.68 (13.02)	2.53 (2.40)	18.48 (15.00)	9.84 (6.64)	14.81 (10.77)
표준 편차	15.96 (7.75)	34.43 (13.39)	16.39 (8.00)	26.85 (9.81)	2.15 (2.08)	21.72 (11.17)	18.46 (6.66)	23.38 (8.81)

부록 2. 분해분석의 ARIMA모형화 결과

Series No.	Obs. Interval	Original Series	Trend Series	Residual Series
1	M	(2,1,0)	(4,0,1)	(6,0,0)
2	M	(2,1,0)	(8,0,0)	(2,1,0)
3	M	(1,0,12)	(5,13,12)	(2,0,0)
4	M	(13,0,0)	(14,3,0)	(0,0,0)
5	M	(13,0,0)	(6,12,0)	(14,0,0)
6	M	(1,0,12)	(14,3,0)	(0,0,0)
7	Q	(0,1,10)	(2,2,1)	(0,0,1)
8	Q	(2,4,4)	(6,2,5)	(5,1,4)
9	Q	(1,1,0)	(6,0,1)	(3,0,0)
10	Q	(1,1,2)	(3,3,1)	(1,0,0)
11	Q	(14,5,0)	(1,7,5)	(2,0,1)
12	Q	(1,0,0)	(4,0,0)	(2,2,0)
13	M	(13,0,0)	(16,0,12)	(0,0,0)
14	M	(14,12,0)	(3,1,12)	(3,12,12)
15	M	(4,0,0)	(4,0,0)	(0,0,0)
16	M	(3,12,12)	(3,12,12)	(2,3,0)
17	M	(0,12,12)	(16,0,0)	(3,2,12)
18	M	(3,1,1)	(2,18,1)	(3,1,0)
19	M	(1,1,0)	(0,0,1)	(0,0,0)
20	M	(3,3,12)	(2,1,2)	(1,12,0)
21	M	(2,1,0)	(8,1,1)	(1,14,0)
22	M	(2,1,0)	(1,2,1)	(2,15,0)
23	M	(2,1,0)	(1,2,1)	(3,0,0)
24	M	(13,2,1)	(14,0,1)	(15,0,0)
25	M	(12,1,0)	(6,2,1)	(12,0,0)
26	M	(1,12,12)	(15,2,13)	(16,0,0)
27	M	(15,3,0)	(3,12,13)	(12,4,0)
28	M	(2,12,12)	(15,6,13)	(5,12,1)
29	M	(17,12,0)	(5,14,12)	(13,7,0)