Journal of Corporation and Innovation
Volume 44, Number 3, Fall, 2021 : pp.3~18

Received 2021. 08. 23
Revised 2021. 09. 23
Accepted 2021. 09. 23

A Study on Korean Exchange Rate Prediction Using Machine Learning*

Jae-Whak Roh**

< ABSTRACT >

In this study, the prediction ability on exchange rate was compared with that of machine learning and non-machine learning methodologies. Two experiments are performed. In experiment 1, exchange rate for the year of 2019 was forecasted with the training data from 2001 to 2018. In experiment 2, exchange rate for the year of 2018 and 2019 were forecasted with the training data from 2001 to 2017. In both Experiment 1 and Experiment 2, it was shown that the predictive ability of the machine learning series was superior to that of the non-machine learning series such as ARIMA.AUTO or ETS methods. In particular, in both experiments, the MLP(multilayer perceptron) showed a very good forecast-ability , and both TSFKNN, which extended KNN in a time series, and NNETAR, which extended a neural network in a time series, showed similar good abilities. In the traditional non-machine learning methods, the characteristics of the data were not sufficiently identified, and thus showed a low level of predictive ability.

Key Words : Exchange Rate, Machine-Learning, Uni-variate Analysis, Time-Series, Forecasting

* This study was supported by research fund from Hansung University in 2021.
** Professor, Department of International Trade, Hansung University (jwroh@hansung.ac.kr)

기업과혁신연구
제44권 제3호, 2021 : pp.3~18

논문투고일 2021. 08. 23
심사완료일 2021. 09. 23
게재확정일 2021. 09. 23

머신러닝을 이용한 우리나라 환율 예측 연구*

노 재 확**

< 국문초록 >

본 연구에서 환율의 예측 능력에 대하여 머신러닝 계열의 방법론과 비-머신러닝 계열의 방법론의 예측 능력을 상고 비교하고자 하였다. 데이터는 2001년부터 2018년을 학습의 기간으로 삼아 2019년을 테스트 하는 실험1과 2001년부터 2017년을 학습 기간으로 삼고 2018년과 2019년을 예측하는 실험2로 나누어 실험을 하였다. 실험1과 실험2 모두에서 머신러닝 계열의 예측이 비-머신러닝 계열의 예측보다 MSE측면에서 우수함을 보였다. 특히 두 실험 모두에서 다층퍼셉트론(MLP)이 매우 우수한 능력을 보였고, KNN을 시계열로 확장을 한 TSFKNN과 신경망을 시계열로 확장을 한 NNETAR 두 가지 모두 유사한 능력을 보였다. 전통적인 비-머신러닝 계열에서는 충분히 데이터에 대한 특성이 파악이 되지 않아 낮은 수준의 예측 능력을 보이는 것으로 판단이 된다.

주제어: 환율, 머신러닝, 단일변량분석, 시계열, 예측

< 목 차 >

* 이 논문은 2021학년도 한성대학교 학술연구비의 지원을 받아 연구되었음.
** 한성대학교 사회과학대학 국제무역트랙 교수 (jwroh@hansung.ac.kr)

Ⅰ. 서론

환율을 보다 정확히 예측하는 일은 매우 중요하다. 환율 예측에 관한 내용은 많은 연구가 발표 되어 왔고 최근에는 점차 머신러닝의 기법을 이용하는 연구도 점차 생겨나고 있다. 이 연구도 이러한 변화에 따라 환율을 예측하기 위한 적정한 방법을 찾으려는 목적으로 시작이 되었다.

본 연구는 지금까지 시계열의 단일 환율 변량 예측으로 많이 사용하였던 ARIMA모형이나 공간지수평활법(ETS)에 대비하여 대표적 머신러닝 방법론을 이용하여 환율을 예측하고 이를 상호 비교하여 하는 것을 본 연구의 목적으로 삼았다. 이러한 작업의 시도는 정확한 환율의 예측은 무역 정책의 입장에서 경상수지의 예측 및 수출, 수입량의 예측, 외환 재정의 운용의 측면에서 매주 중요한 일이 아닐 수 없기 때문이다.

환율의 예측을 위하여 결정해야 하는 첫 번째 이슈는 단일변량으로서 환율을 예측할 것인가 아니면 다변량을 이용하여 예측할 것인가의 문제이다. 또 다른 관련된 이슈는 어떤 모델 또는 어떤 방법론을 이용할 것인가를 결정해야 한다.

단변량으로 환율을 바라보는 것에 비하여 다변량으로 바라보는 것은 장단점을 가지고 있다. 단변량으로 보는 것은 고려해야 하는 과거 데이터가 환율 자체의 래그를 사용하므로 예측이 용이한 것이 가장 큰 장점이다. 반면 다변량으로 파악을 한다면 환율을 결정해야 하는 많은 요인을 고려하여야 하고 서로간의 관계가 규명이 되어야 한다는 전제를 가지게 된다.

이러한 점에서 본 연구에서는 시계열에서 GDP 예측을 목적으로 하여 폭넓은 다변량과 단변량의 예측 비교를 한 Iwok and Okpe(2016)의 연구에 주목을 하였다. 이들의 경우 GDP의 예측에서 다변량 분석과 단변량 분석의 기법을 모두 사용하여 예측을 비교하였고 결과 단변량의 예측의 우수성을 보여 주었다. 이들은 ARIMA뿐만 아니라 다변량으로 접근하는 Vector Error Correction Model(VECM)과 Vector Autoregressive(VAR) 등을 비교 검토하였다. 결과 예측력에서 단변량이 갖는 우수성을 증명하였다.

그러나 다른 연구인 Gabriel(2015)의 경우 다른 결과를 보여 주었다. Gabriel은 수 많은 변수의 영향을 받는 변수에 대하여 단변량으로 분석하는 것은 다소 무리가 있다는 주장이다. 그의 연구는 다변량 추정을 통하여 투자, 수입, 소비 변수 등에 대하여 추정을 하고 추정의 잔차를 비교하여 다변량 추정의 방식이 우수함을 보여 주었다. 즉, 단변량과 다변량의 선택은 주제에 따라 선택의 문제로 일단 정리가 된다.

다음으로 결정할 문제는 예측 모형을 어떻게 구성할 것인가의 문제이다. 환율에 관한 내용은 전통적으로 많이 연구되어 왔다. 주로 시계열의 분석의 대상으로 연구 되었으며 주로 VAR 또는 VECM등으로 연구되어 왔다. 그러나 본 연구는 전통적 방법이 아니고 최근에 많이 연구되는 머신러닝 또는 기계학습의 방법을 통하여 환율의 예측을 시도하려고 한다. 최근 머신러닝을 이용하는 연구는 점차 늘어나고 있으며 환율에 대하여 서로 주안점이 다르지만 머신러닝 계열의 방법론을 이용하는 연구가 발생하고 있다

환율에 대한 전통적 분석법은 신축가격모형이나, 경직가격모형, 종합모형, 또는 랜덤워크 등의 방법론을 이용하여 외화 가격의 방법으로 접근하여 왔다. 그러나 본 연구에서는 이런 방법론 보다는 머신러닝의 최근의 기법을 활용하고자 한다.

국제경제 분야에서 환율과 연구에서 머신러닝의 기법이 채택된 연구는 점차 늘어나는 추세이다. 대표적으로 서종덕(2016), 김영철 외3인(2018), 한정아(2019), 이재득(2021)등으로 정리가 된다. 이들이 주로 연구하는 머신러닝의 기법은 다층퍼셉트론(MLP), 신경망분석(Neural Network), 랜덤포레스트(Random Forest) 등의 방법론을 많이 이용하고 있다.

본 연구에서는 아직 우리나라에서는 환율의 예측에 시도된 적이 없는 TSFKNN, NNETAR, MLP등의 머신러닝의 방법을 사용하여 환율을 예측한다. TSFKNN은 KNN계열의 방법론이면서 시계열을 분석할 수 있도록 개발이 된 머신러닝 알고리즘이다. NNETAR은 신경망계열의 방법론이면서 시계열에 적합하도록 개발이 된 것이다. 그리고 가장 많이 사용되는 다층퍼셉트론(MLP)도 같이 사용하고자 한다. 이와 같은 머신러닝 계열의 방법론을 이용하여 예측하고, 이에 대비하여 전통적 환율의 예측 방법이었던 공간지수평활법(ETS)와 ARIMA의 방법을 이용하여 결과를 상호 비교 하고자 한다.

Ⅱ. 예측 알고리즘의 이론적 고찰

1. TSFKNN 알고리즘 및 활용

TSFKNN은 KNN알고리즘을 시계열의 예측에 활용하는 알고리즘을 지칭한다. KNN 알고리즘은 비모수계열의 매우 중요한 머신러닝 알고리즘이며 다양한 분야에서 활용되고 있는 알고리즘이다. 예를 들어 회귀분석에도 확대되어 사용되기도(Martínez et al., 2018)하고, 기술적으로는 영상보간(image interpolation)이나, 시각인식에 활용되기도 한다. Zhang, et al. (2017)은 앙상블기술을 이용하여 다면형(multidimensional) k-NN 모형을 주식의 예측에 활용하기도 하였다. 시계열 분야에서는 Lora, et al. (2007)은 가중 k-NN 회귀를 이용하여 전기사용 가격 예측에 활용하였으며, Ahmed, et al. (2010)도 k-NN을 이용한 예측을 하였으나 이들의 연구는 단지 한 단위 앞을 예측하는 것에 머물렀다.

TSFKNN에서 단일변량으로 예측하는 구조는 필요한 래그의 숫자와 밀접한 관계가 있으며 과거의 래그를 행태변수(features)로 활용하게 된다. 만약 $t = \{x_1, x_2, x_3, x_4, x_5, \ldots, x_{132}, x_{133}\}$ 월별 시계열 자료가 있다고 가정을 하자. 12개의 데이터가 11년에 걸쳐 있으므로 데이터의 개수는 132개이다. 주어진 데이터를 가지고 우리는 다음 달의 값을 예측을 통하여 다음 단계의 값을 알려고 한다. 이런 타겟(target)을 설명하는 형태변수(features)는 12개의 과거 래그(lag)가 필요하다고 가정하자. x_{13}을 알기위하여 과거 12개의 래그 변수인 $(x_1, x_2, \ldots, x_{12})$을 알아야 한다. 동일하게 x_{133}을 예측하기 위하여 과거 래그 변수인 $(x_{121}, x_{122}, \ldots, x_{132})$을 알아야 한다. 다시 말하면, 예측 타겟을 알기 위한 형태(features)변수로 과거 12개의 래그(lag)가 필요하게 된다. 이때 새로운 타겟을 설명하기 위한 변수가 바로 형태변수(features)가 된다.

이제 이를 일반적으로 표시해 보면, 먼저 인스턴스가 존재한다. $(f_1^i, f_2^i, f_3^i, \ldots, f_n^i)$의 i번째 인스턴스가 있다고 하면, 타겟 벡터에 m개의 속성이 $(t_1^i, t_2^i, t_3^i, \ldots, t_n^i)$으로 표시된다. 만약 예측해야 하는 새로운 인스턴스에 대하여 이들의 형태변수(features)는 $(q_1, q_2, q_3, \ldots, q_n)$가 주어졌고, 타겟을 구해야 한다면 이때 k-nn 알고리즘에서 활용하는 유클리디언(Euclidean)거리 측정 공식을 이용하여

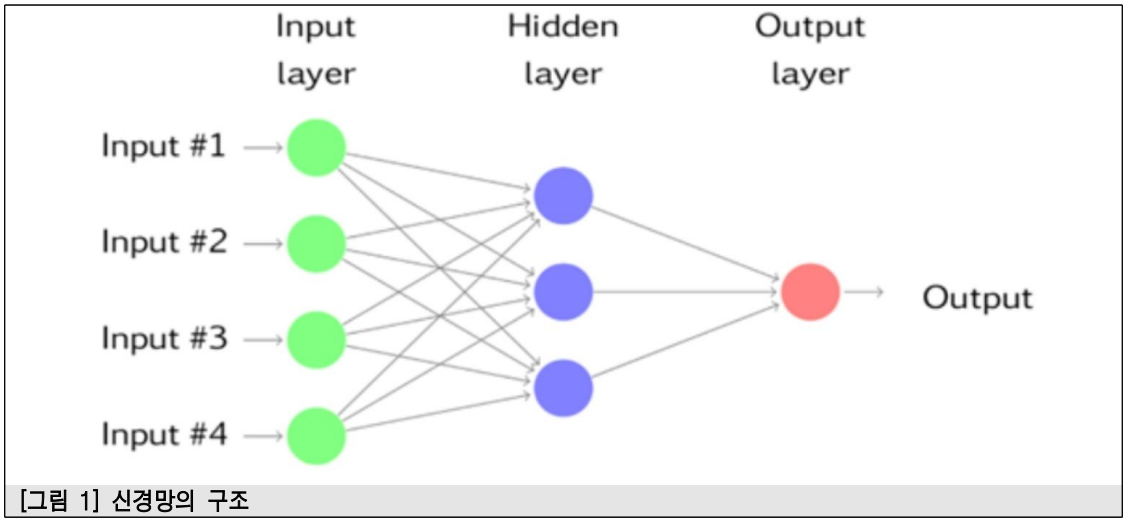
$$\sqrt{\sum_{x=1}^{n} \left(f_x^i - q_x\right)^2} \quad \cdots\cdots (1)$$

을 이용하여 분류하게 된다. 즉, 예측을 위한 새로운 인스턴스가 정해지면 최근접이웃(nearest neighbors)의

개수 만큼을 구하여 활용하게 된다. 예를 들어 최근접이웃이 $(t^1, t^2, t^3, ..., t^k)$의 벡터의 꼴이라면 새로운 인스턴스의 값은 최근접이웃 모두를 활용한 $\sum_{i=1}^{k} \frac{t^i}{k}$을 가 된다. 즉, 최근접이웃의 개수만큼을 활용하여 새로운 인스턴스를 구하게 된다.

2. NNETAR 알고리즘 및 활용

인공신경망(Neural Network)을 나타내는 가장 기초적 형태는 아래의 그림처럼 투입층에 투입(Input)이 발생하고 이를 결과층(Output layer)의 결과(Output)로 구성이 된다. 각 투입층과 은닉층(Hidden Layer)사이를 나타내는 선으로 각 선의 가중치(Weight)를 표시하게 된다.



[그림 1] 신경망의 구조

은닉노드 z_j에 주는 영향은 아래와 같이 입력노드x_i에 w_i가중치가 조정된 합으로 은닉층에 영향을 준다. 이를 수식으로 나타내면 다음과 같다.

$$z_j = b_j + \sum_{i=1}^{4} w_i x_i \quad \text{.....(2)}$$

은닉층에서는 활성화함수를 사용하게 된다. 예들 들어 sigmoid함수를 사용한다면 아래처럼 표현이 된다.

$$S(z) = \frac{1}{1+e^{-z}} \quad \text{.....(3)}$$

이때에 파라미터 b_1, b_2, b_3, b_4 등과 w_1, w_2, w_3, w_4 등으로 표시되는 가중치는 데이터로부터 학습하게 된다. 가중치가 너무 커지는 것을 막기 위해 일반적으로 제한을 가하는데 이를 붕괴파라미터(decay parameter)이라고 부르며 일반적으로 0.1을 넘지 않도록 한다.

시계열을 이용하여 신경망 자기회귀 (Neural Network Autoregression)로 예측하는 경우 래그가 매우 중요한 요인으로 작용을 한다. 한 단계 앞의 값을 예측하기 위하여 시계열에서 자기의 래그 변수가 입력으로 사용하게 되면 이를 신경망자기회귀(Neural Network Autoregression)또는 NNAR모형이라고 부른다. 이를 NNAR(p,k)로 표시하고 자기회귀의 수준이 p이며 은닉층에 있는 은닉노드의 숫자가 k임을 나타낸다. 예를 들어 NNAR(9,5)라면 $(y_{t-1,}\ y_{t-2,}\ ,\cdots,\ y_{t-9,})$를 이용하여 예측하게 되고 일반화하면 p개의 래그인 $y_{t-1}y_{t-2},\ldots,y_{t-p,}$ 를 이용하여 $y_{t,}$ 를 예측한다는 의미가 된다.

계절성이 있는 경우에는 그 구조가 조금 더 복잡해진다. 계절성이 있는 경우 표기는 $NNAR(p,P,k)m$ 표기하게 되면 $(y_{t-1}y_{t-2},\ldots,y_{t-p,}\ y_{t-m}y_{t-2m,}\ \ldots,\ y_{t-Pm})$을 입력으로 사용하고 이때 k는 앞과 동일하게 은닉노드의 숫자를 의미하고, m은 계절성을 나타낸다. 따라서 $y_{t-1}y_{t-2},\ldots,y_{t-p,}$ 의 p개의 자기 래그와 계절성을 설명하는 m의 p개의 자기회귀 $(y_{t-m}y_{t-2m,}\ \ldots,\ y_{t-Pm})$ 모두 y_t 를 예측하는 자료로 활용하게 된다.

본 연구에서 사용하는 NNETAR( )에서는 $NNAR(p,P,k)m$에 맞추어 추정하게 된다. 실험자가 p(소문자), P(대문자)를 명시하지 않아도 데이터의 특성을 자동적으로 찾아주는 알고리즘이다. 이런 점에서 NNETAR의 편리성이 있다. 비-계절성 데이터에 대해서는 AIC기준으로 최적의 래그를 선택하게 된다. 그리고 계절성이 있는 데이터에 대해서는 디폴트 값은 계절성 래그를 먼저 P=1로 상정을 하고 자기회귀수준을 찾는 방식을 택하게 된다. 만약 적정 은닉노드의 숫자가 지정이 되지 않을 경우 k=(p+P=1)/2의 공식을 적용하여 찾게 된다.

3. NNFOR의 MLP(Multilayer Perceptron for Time Series Forecasting) 알고리즘 및 활용

본 연구에서 사용하는 NNFOR의 순방향 (feed forward) 다층퍼셉트론 (MLP)의 소개한 연구는 Crone and Kourentzes (2010)이다. 다층퍼셉트론의 방법론의 핵심은 반복적으로 신경망 필터를 작동시키는 것이다. 이렇게 해서 그림을 통하여 데이터간의 거리를 분석하는 방법과 시계열의 자료의 분석과 혼합하는 방식이다.

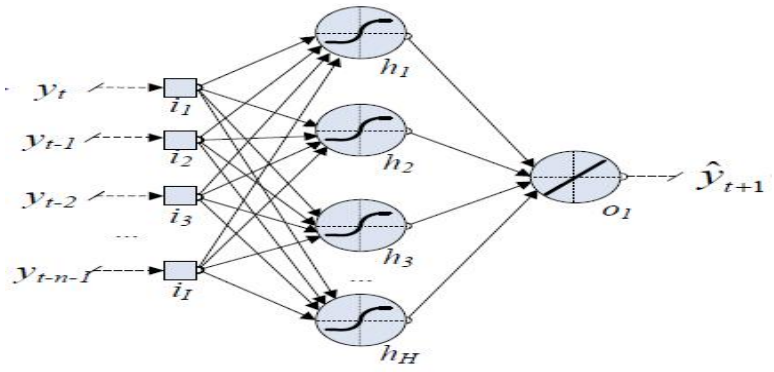
다음과 같이 독립변수 X와 종속변수 Y가 있다고 하자. 이를 예측 함수로 나타내면 $\hat{y}= f(X,\ Y)$ 로 표시할 수 있다. 우리가 구하고자 하는 $\hat{y}$를 위하여 독립변수 X, 종속변수 Y의 관계식이 함수f의 형태로 정의가 되어야 한다. 함수가 파악이 되면 $\hat{y}_{t+1}$을 예측하기 위하여 과거 래그 벡터 $(y_t,\ y_{t-1},\ \ldots,y_{t-n+1})$가 설명변수로 활용된다. 이를 식으로 표현하면$\hat{y}_{t+1} = f(y_t,\ y_{t-1},\ \ldots,y_{t-n+1})$ 가 된다.

MLP의 편리성은 데이터의 자기회귀 수준과 이동평균의 수준을 파악하는 것은 매우 까다로운 작업인데 이 과정을 알고리즘이 스스로 찾아 준다는 점이다. 그리고 이렇게 파악이 된 자기회귀 수준과 이동평균의 정보를 이용하여 예측까지 실현시킨다는 점에서 MLP의 우수성이 있다.

파악이 된 추정식을 바탕으로 h시점의 앞을 예측할 경우, MLP의 알고리즘은 전통적으로 순방향 (feed-forward)구조에 바탕을 둔 다층퍼셉트론(Multi-Layer Perceptron)을 이용한다. 만약 h구간 앞을 예측하는 경우 $\hat{y}_{t+h}$로 표시하고, 간단하기 위하여 만약 n=p라고 동일 구간을 가정한다면 $(y_t,\ y_{t-1},\ \ldots,y_{t-n+1})$의 데이터를 활용해야 함을 의미한다. 즉, t, t-1, t-2,…, t-n+1의 데이터를 활용하게 된다.

이를 식으로 정리하면 아래처럼 정리할 수 있다.

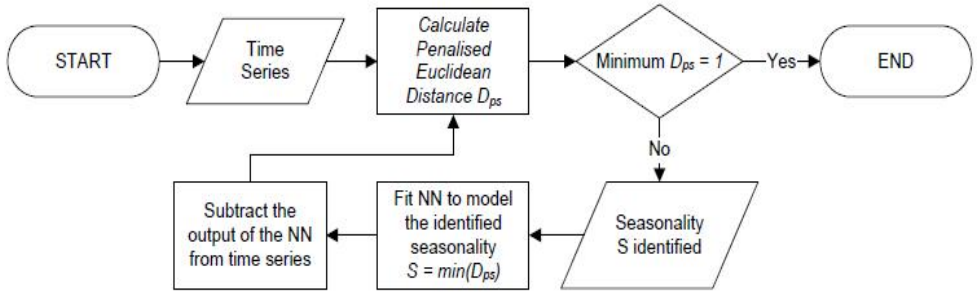
$$f(Y,w)= \beta_0 + \sum_{h=1}^{H} \beta_h g\left(\gamma_{oi} + \Sigma_{i=1}^{I} \gamma_{hi} y_i\right) \quad \text{.....(4)}$$



[그림 2] MLP의 시계열 예측의 구조

주) Sven F. Crone and Nikolaos Kourentzes (2010)

각 신경망은 파라미터w와 종속변수 Y로 표시가 되는데 이는 $f(Y,w)= \beta_0 + \sum_{h=1}^{H} \beta_h g(\ )$ 을 기본적으로 각 미래예측구간값의 선형조합의 합으로 구한다. 이때 각각 구간의 예측값은 예측을 하는 각 h의 단계마다 $\gamma_{oi} + \sum_{i=1}^{I} \gamma_{hi} y_i$ 의 NN의 구조가 활용이 된다. [그림2] 에서 보듯이 각 은닉층은 활성화함수로 변환된 값을 가중치 hi로 사용하고 이런 값을 모아서 최종 한 구간의 예상 출력의 $\hat{y}_{t+1}$가 계산이 됨을 보여주고 있다. 이 작업의 과정을 보면 먼저 데이터를 입력 받는다. 초기 데이터간의 거리를 1로 주고 시작한 다음, 데이터의 계절성을 파악하고 결과값을 보고 이를 수정하는 방식을 계속하면서 최적의 값을 찾게 되는 과정을 설명하고 있다.



[그림 3] MLP의 처리과정

4. 지수평활상태공간모형(ETS)

본 연구에서는 ETS를 지수평활상태공간모형으로 해석한다. 여기에서 상태를 파악하고 파악된 상태 후의 데이터 진화의 프로세스(process)가 지수평활법(exponential smoothing)을 따른다는 의미가 된다. 지수평활상태공간모형은 먼저 ETS는 오차(error), 추세(trend), 그리고 계절성(seasonal)로 일단 상태를 정의하게 된다.

ETS가 예측하는 구조를 설명하기 위하여 5가지의 추세의 분석을 전제로 한다. 먼저 N(추세없음), A(덧셈형 추세), Ad(완화된 덧셈형), M(곱셈형), Md(완화된 곱셈형)으로 나눌 수 있고 계절적 요인도 N(없음), A(덧셈형), 곱셈형(M)으로 구분을 한다. 여기서 구체적 이해를 위하여 예를 들면 다음과 같다. 아래에서 l_t는 데이터의 레벨을 나타낸다. 먼저 $l_t = \alpha y_t + (1-\alpha) l_{t-1}$는 N(관계없음)형이다. 이는 특별한 성장도(b_t)와 과거레벨 (l_{t-1})의 관계가 설정이 되어 있지 않다 의미이다. $l_t = \alpha y_t + (1-\alpha)(l_{t-1} + b_t)$로 표시하는 것은 덧셈형인데 $(l_{t-1} + b_t)$가 덧셈의 관계이다. $l_t = \alpha y_t + (1-\alpha)(l_{t-1} + \phi b_{t-1})$은 완화된 덧셈형인데, $(l_{t-1} + \phi b_{t-1})$ 에서 ϕ만큼 관계가 완화될 수 있음을 표시하고 있다. 다음은 곱셈형인데 $l_t = \alpha y_t + (1-\alpha)(l_{t-1} b_t)$곱으로 나타난다. 마지막으로 $l_t = \alpha y_t + (1-\alpha)(l_{t-1} b_{t-1}^{\phi})$완화된 곱셈형은 기본적으로 곱셈형이기는 하지만 영향이 완화되어 나타난다. 이렇게 하여 추세의 분류를 5가지로 하고 계절성의 분류를 3가지로 한다면 모두 15가지의 조합으로 현재의 상태(state)를 먼저 판단하게 된다. 이 점에서 모형의 상태가 강조되는 이유이다.

상태를 판단한 뒤에 상태 백터는 $x_t = (\ l_t,\ b_t,\ s_t,\ s_{t-1}, \cdots, s_{t-m+1}\ )'$로 표현이 가능하고 y_t와 상태백터x_{t-1} 간의 관계와 현재의 상태백터 x_t와 과거상태벡터x_{t-1}의 관계를 규명하는 비선형함수를 풀게 된다.

$$y_t = w(x_{t-1}) + r(x_{t-1})\varepsilon_t$$
$$x_t = f(x_{t-1}) + g(x_{t-1})\varepsilon_t \quad \text{.....(5)}$$

$w,\ r,\ f,\ g$이들의 함수가 각각 지수평활(exponential smoothing)의 방법을 활용하게 된다. 이때 $\{\varepsilon_t\}$는 가우시안 백색프로세스를 따르고, 평균은 0이고 σ^2분산은 이다. 구체적으로 예들 들어 Makridakis et al. (1998), 의 모형에 따라 지수평활(exponential smoothing)을 설명하면 아래에서 보는 바와 같이 상태를 나타내는 l_t와 b_t 그리고 s_t 가 정해지는 함수가 정해지고 나면 이들과 관련하여 바로 $\hat{y}_{1+h|t}$ 가 결정이 된다. 각 레벨과 성장함수 그리고 계절성간에는 이런 식의 표현이 가능하다.

$$l_t = \alpha P_t + (1-\alpha) Q_t$$
$$b_t = \beta R_t + (\phi - \beta) b_t$$
$$s_t = \gamma T_t + (1-\gamma) s_{t-m} \quad \text{.....(6)}$$

여기에서 l_t는 시리즈의 레벨을 나타낸다. b_t는 t시점에서 성장률(growth)를 나타낸다. 그리고 s_t는 계절적

요인을 나타낸다. 마지막으로 m은 시즌의 개수를 의미한다. 이런 상태가 진보하여 $\hat{y}_{1+h|t}$가 결정이 되게 된다.

5. 자기회귀누적이동평균(AutoRegressive Integrated Moving Average)

시계열의 분석에서 데이터를 분석하여 자기회귀 수준을 나타내는 p와 평균이동 q의 수준의 결정은 어려운 문제이다. 이런 단점을 극복하고 자동으로 p, q를 찾아주는 것이 자동이동평균누적자기회귀의 방법이다.

자동으로 자기회귀의 수준과 이동평균의 수준을 정하여 주는 알고리즘은 처음 Hannan and Rissanen (1982)이 제안한 방법이 된다. 이들의 방법에서는 먼저 장기간에 적용되는 자기회귀 수준을 먼저 결정하고 다음에 표준적 회귀분석을 통하여 가능한 모델에 대한 우도(likelihood)를 계산하는 방법을 이용한다.

초기의 제안이 Gómez (1998)는 이런 방법을 조금 더 발전시켜 승수적(multiplicative)계절 ARIMA에 적용하는 방법을 제안하게 된다. 이후 Gómez and Maravall (1998), Makridakis and Hibon (2000) 그리고 Autobox를 만든 Reilly(2000) 등의 발전을 하게 된다.

먼저, 비계절성$ARIMA(p,d,q)$는 다음과 같이 (7)로 표현된다.

$$\Phi(B)(1-B^d)y_t = c+\theta(B)\varepsilon_t \quad \text{.....(7)}$$

여기에서 $\{\varepsilon_t\}$는 백색프로세스를 따른다. 따라서 평균은 0이고 분산σ^2이다. 여기에서 B는 후방이동연산자(backshift operator)이다. $\phi(z)$와 $\theta(z)$는 p와 q로 이루어진 다항식이며 각각 p와 q의 수준으로 나타난다.

이때 누적의 수준을 나타내는 c가 0가 아니라면 식은 조금 더 복잡해져서 아래 (8)을 따른다.

$$\Phi(B^m)\phi(B)(1-B^m)^D(1-B)^d y_t = c+\Theta(B^m)\theta(B)\varepsilon_t. \quad \text{.....(8)}$$

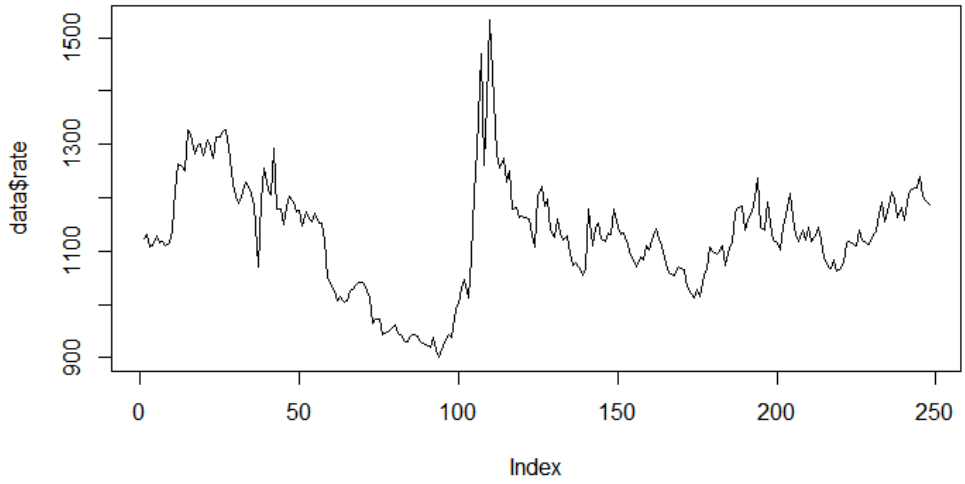
우리는 데이터가 주어질 경우 적절한 추정은 올바른, p, P, q, Q, d, D를 선택하는 것을 의미한다. 만약 d, D가 주어진다면 $AIC=-2\log(L)+2(p+q+P+Q+K)$를 기준으로 적정한 p와 P그리고 Q와 q를 선택하는 것을 의미한다.

Ⅲ. 데이터 시뮬레이션

1. 데이터

데이터는 e-나라지표에서 2000년 1월부터 2018년 12월까지의 우리나라의 환율 데이터를 구하였다. 이렇게 구한 데이터를 머신러닝의 학습용으로 2017년12월까지의 데이터를 사용하고 예측의 실험을 위하여 2018년 12개의 데이터는 테스트용으로 남겨두었다. 따라서 학습용으로 216개의 데이터가 사용되었고, 테스트용으로 12개의 데이터가 활용이 되었다. 이렇게 하여 본 연구에서 사용하는 다수의 실험을 통하여 어떤 모형이 가장 테스트용의 데이터 예측에서 우수성을 보여주는가를 확인하기 위함이다.

전체의 데이터를 그림으로 표시하면 다음과 같이 표시가 된다.



[그림 4] 우리나라 환율의 움직임

데이터를 구한 후에 데이터의 비안정서(non-stationary)를 체크하기 위하여 adf 테스트를 실시하였다. 결과 Dickey-Fuller = -3.0183, Lag order = 6, p-value = 0.1472의 결과를 보여 주었다. 이는 데이터가 안정적이라는 의미가 된다. 따라서 앞으로 진행되는 실험은 데이터의 안정성을 전제로 진행한다.

2. 시뮬레이션 결과

1) 2019년 1개년 예측의 결과

앞에서 제시한 여러 방법을 이용하여 2019년 환율에 대한 예측을 실시하였다. 실제의 실현된 데이터는 우상향의 모습을 보여주는 데이터이다.

TSFKNN을 실행하는 경우 주의해야 할 점은 최근접이웃의 값을 지정해야 한다는 점이다. 이를 위하여 12을 지정해 주었다. 보통의 경우 데이터 개수의 제곱근을 지정한다는 일반적 룰에 따라 12를 지정해 주었다. 결과 fit.TSFKNN의 근접이웃은 157 170 38 137 185 179 139 171 141 166 172 138으로 알고리즘은 보여주었다. 우리의 미래 예측의 값은 이 최근접 이웃의 값을 구한 뒤 이들을 평균한 값으로 구해진다. NNETAR에서 데이터를 지정하고 나면 알고리즘은 모델의 형태를 알려준다. 결과 NNAR(4,2)의 형태를 결과값으로 제시하였다. 이것은 데이터에서 4개의 자기회귀가 있다고 알고리즘은 판단한 것이며 은닉노드의 숫자를 2개를 사용했음을 나타낸다. 즉 4-2-1의 네트워크의 모습을 보여 준 것이다. 4개의 래그의 숫자가 입력으로 활용이 되고 은닉층은 2개 그리고 마지막으로 결과값의 형태로 구성된 네트워크의 형태를 결과로 산출한 것이다.

이에 반하여 MLP에서 찾은 모형은 다소 달랐다. MLP에서는 모든 4개의 래그를 사용하지 않고 1,3,4의 세개만의 래그를 사용하였다. 그리고 은닉 노드의 개수도 5개를 이용하였다. 따라서 3-5-1의 형태의 네트워크의 모습을 보여 주었다.

〈표 1〉 각 모형의 2019년 예측 결과 비교

모델명	MSE(mean squared error)
TSFKNN	20.24
NNETAR	20.3869
MLP	11.91
ETS	36.79
AUTO.ARIMA	19.85

각 예측의 결과 2019년을 가장 우수하게 예측한 모형은 MLP방식이다. 평균제곱오차(mse)값을 11.91을 보여 주었다. 다음으로는 KNN의 방법론을 활용하여 시계열에 적용되는 TSFKNN의 경우 20.24를 보여 주었다. 시계열에 적용하는 신경망 분석법인 NNETAR의 경우 매우 유사하게 20.36을 보여 주었다. 전반적으로 우리가 채택한 머신러닝의 방법은 매우 우수하게 작동하였다.

반면, 전통적 방법으로 분류한 AUTO.ARIMA방식으로 mse값을 19.85를 보여 주어 머신러닝에 비하여 뒤지지 않음을 보였다. 하지만 상태지수평활법의 경우 예측력이 가장 떨어지는 모형으로는 36.79를 보여 주었다.

〈표 2〉 각 모형의 12개월 예측값

	y.test	fc.tsfkn	fc.tar	fc.mlp	fc.ets	fc.aa
t+1	1067.9	1092.033	1076.222	1071.827	1071.941	1074.506
t+2	1082.8	1085.633	1082.773	1070.080	1071.941	1078.200
t+3	1063.5	1096.058	1090.534	1071.236	1071.941	1081.605
t+4	1068.0	1101.992	1099.939	1075.115	1071.941	1084.744
t+5	1077.7	1116.067	1109.158	1079.913	1071.941	1087.637
t+6	1114.5	1101.500	1117.843	1089.487	1071.941	1090.305
t+7	1118.7	1107.408	1125.812	1101.060	1071.941	1092.764
t+8	1112.9	1110.150	1132.606	1111.514	1071.941	1095.031
t+9	1109.3	1122.367	1137.336	1118.650	1071.941	1097.121
t+10	1139.6	1128.050	1139.345	1120.852	1071.941	1099.048
t+11	1121.2	1121.483	1139.869	1121.624	1071.941	1100.824
t+12	1115.7	1121.175	1140.051	1121.375	1071.941	1102.462

이를 그림으로 표시하면 다음과 같다. 그림으로 보면 검은색으로 표시된 것이 실제 2019년 환율의 값을 나타낸다. 가장 우수한 예측력을 보인 MLP의 경우 실제값을 기준으로 볼 경우 상반기에는 다소 실제의 값과 거리가 있어나 하반기의 경우 실제의 값과 매우 유사함을 볼 수 있다.

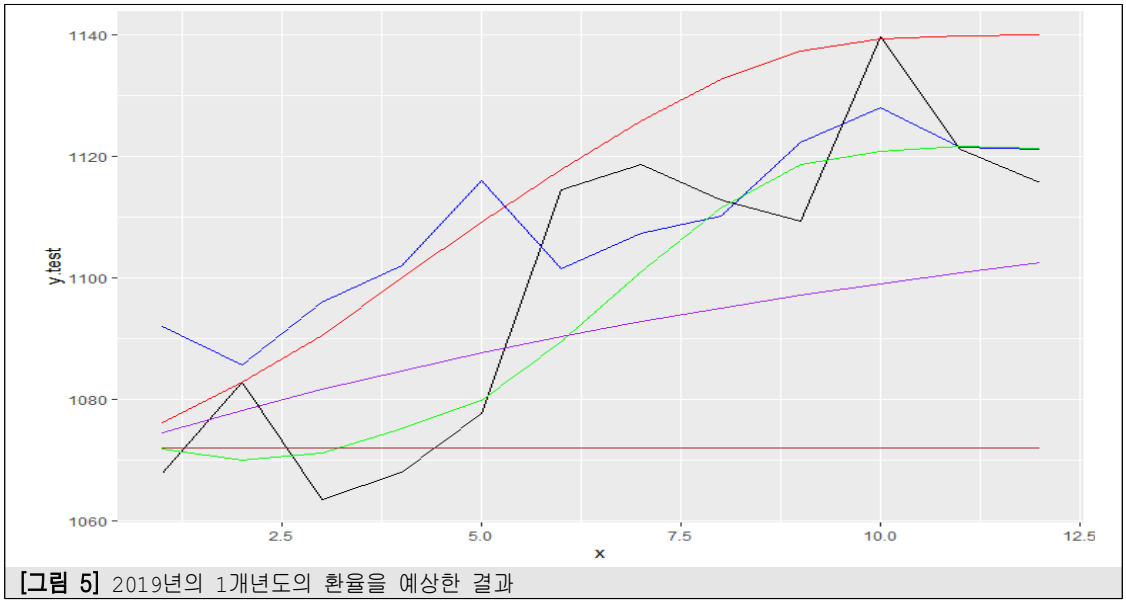
TSFKNN의 경우도 역시 우수한 예측력 보였다. 하지만 MLP와 상반기의 예측을 비교하면 MLP의 경우 상반기를 다소 낮게 평가한 반면, TSFKNN의 경우 상반기를 다소 높이 평가한 특징을 보이고 있다. 그러나 전체의 성능은 5가지 중에서 두 번째로 우수함을 보였다.

NNETAR의 경우 전체적으로 높게 예측하였음을 볼 수 있다. 그러나 우리가 채택한 머신러닝 계열의 세 가지 모두 매우 우수한 예측력을 보임을 알 수 있다.

반면, AUTO.ARIMA 의 경우와 지수평활의 방법에서는 다소 직선의 모습이 관측이 되어 머신러닝에서 보여 주는 유동성이 미흡함을 보여 주고 있다.

2) 2018, 2019년 2개년 예측의 결과

2001년부터 2017년까지를 머신학습기간으로 하고 2018년과 2019년을 테스트 기간으로 선정한 예측의 결과이다. 각 예측의 결과 2개년을 가장 우수하게 예측한 모형은 MLP방식이다. 평균제곱오차(mse)값을 54.28을 보여 주었다. 앞의 경우와 달리 2개년을 예측하는 경우 NNETAR이 TSFKNN보다 우수하게 작동을 하였다. 그러나 이 둘의 성과는 크게 다르지 않다. 눈에 띄는 변화는 AUTO.ARIMA의 성과가 가장 나빠졌다는 점과 ETS의 결과도 상대적으로 좋지 않다.



[그림 5] 2019년의 1개년도의 환율을 예상한 결과

주) 실제선은 검은색, TSFKNN은 푸른색, NNETAR은 붉은색, MLP는 초록색, ETS는 브라운색, AUTO.ARIMA은 보라색으로 표현이 되어 있다.

결국 머신러닝 계열의 성과가 비-머신계열의 성과보다 우수하다고 결론지을 수 있다.

<표 3> 각 모형의 2018, 2019년 예측 결과 비교

모델명	MSE(mean squared error)
TSFKNN	66.87
NNETAR	66.59
MLP	54.28
ETS	97.14
AUTO.ARIMA	99.948

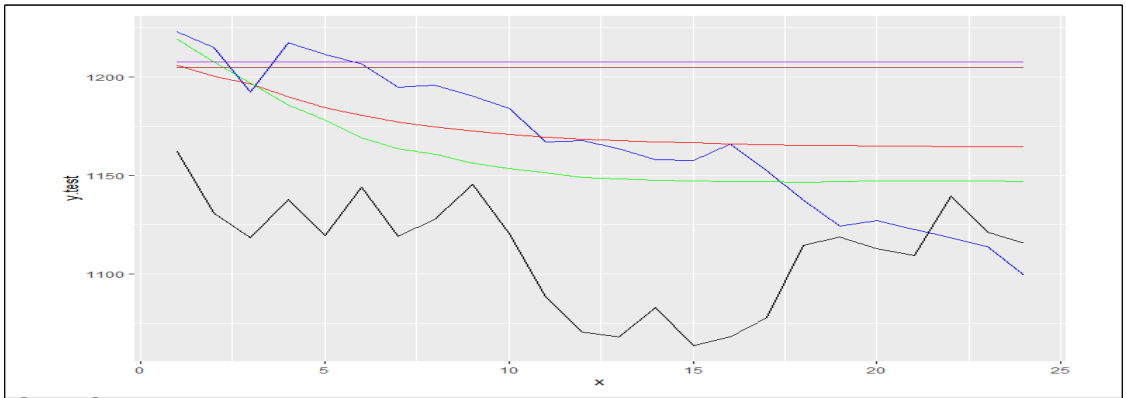
이렇게 비머신계열의 결과를 보는 AUTO.ARIMA의 경우도 추정의 결과를 검토해 보면 ARIMA(0,1,0)로 데이터를 파악하여 예측의 결과가 좋을 수 없고 ETS의 경우도 추정의 성과를 보면 데이터를 ETS(M,N,N)으로 파악을 하고 있어 예측의 결과가 좋을 수 없는 것으로 판단이 된다. 따라서 아래서 보듯이 24개의 예측이 나빠진 결과를 알 수 있다.

아래의 <표>4에서 2018년과 2019년 24개월의 예측 값을 각 종류의 예측방법별로 표를 만들어 두었다.

〈표 4〉 각 모형의 2018, 2019, 24개월 예측값

	y.test	fc.tsfknn	fc.tar	fc.mlp	fc.ets	fc.aa
t+1	1162.1	1222.900	1205.754	1218.893	1204.782	1207.7
t+2	1130.7	1214.792	1200.146	1207.443	1204.782	1207.7
t+3	1118.4	1192.375	1196.524	1196.918	1204.782	1207.7
t+4	1137.9	1217.167	1189.738	1185.522	1204.782	1207.7
t+5	1119.5	1211.292	1184.305	1177.974	1204.782	1207.7
t+6	1144.1	1206.600	1180.336	1169.108	1204.782	1207.7
t+7	1119.0	1194.575	1177.005	1163.592	1204.782	1207.7
t+8	1127.8	1195.758	1174.403	1160.598	1204.782	1207.7
t+9	1145.4	1190.267	1172.388	1156.068	1204.782	1207.7
t+10	1120.4	1183.858	1170.779	1153.431	1204.782	1207.7
t+11	1088.2	1166.917	1169.497	1151.397	1204.782	1207.7
t+12	1070.5	1167.608	1168.473	1148.841	1204.782	1207.7
t+13	1067.9	1163.633	1167.649	1148.086	1204.782	1207.7
t+14	1082.8	1157.808	1166.983	1147.539	1204.782	1207.7
t+15	1063.5	1157.442	1166.445	1147.147	1204.782	1207.7
t+16	1068.0	1165.925	1166.007	1146.871	1204.782	1207.7
t+17	1077.7	1152.242	1165.652	1146.678	1204.782	1207.7
t+18	1114.5	1137.442	1165.362	1146.544	1204.782	1207.7
t+19	1118.7	1124.175	1165.126	1146.889	1204.782	1207.7
t+20	1112.9	1127.133	1164.933	1147.240	1204.782	1207.7

위에 나타난 <표>4의 결과를 다시 그림으로 표시하면 다음과 같이 [그림6] 으로 표시할 수 있다. 가장 아래쪽의 선이 실제 2018년과 2019년의 그림을 나타낸다. 가장 가까운 그림으로는 바로 위의 선으로 나타난 MLP임을 알 수 있다. TSFKNN과 NNETAR이 바로 위를 나타내고 있다.



[그림 6] 2018년과 2019년 2개년을 예상한 결과

주) 실제선은 검은색, TSFKNN은 푸른색, NNETAR은 붉은색, MLP는 초록색, ETS는 브라운색, AUTO.ARIMA은 보라색으로 표현이 되어 있다.

Ⅳ. 결론

우리 연구의 목적은 환율을 예측하는 것에 두었다. 환율을 예측하는 방법은 다변량 분석과 단변량 분석이 있을 수 있으나 이는 연구자의 목적에 따라 채택될 수 있음을 검토한 후에, 본 연구에서는 단일 변량으로 환율을 예측하기로 하였다. 또한 환율의 예측을 위하여 전통적인 시계열 분석의 다변량 분석 방법인 VAR이나 VECM 등보다 최근에 연구가 시작되는 머신러닝의 방법론을 이용하여 예측하기로 하였다.

최근에 이용되는 시계열의 예측을 위하여 사용하는 방법 중에서 단일 변량의 환율 예측에 적정한 3가지 방법을 선택하였다. KNN계열 중에서 시계열의 예측이 가능한 TSFKNN을 이용하였고, 신경망 계열에서는 NNETAR을 이용하였다. 그리고 다층퍼셉트론(MLP)도 이용하였다. 이들을 이용하여 환율에 대한 예측의 도구로 삼았다.

데이터는 우리나라의 2001년부터 2018까지를 학습데이터로 활용하고 2019년을 테스트 데이터로 하는 실험1과 2001년부터 2017년까지를 학습데이터로 활용하고 2017년과 2018년 매월 환율을 테스트로 사용하는 두개의 실험을 진행하였다.

실험의 결과 두 가지 모두에서 머신러닝계열의 예측 능력이 우수함을 보였다. 특히 다층퍼셉트론의 결과가 NNETAR과 TSFKNN의 결과보다 우수하게 나타났다. 반면 AUTO.ARIMA의 경우 실험1에서는 다소 잘 작동하였으나 실험2에서 작동이 잘 되지 못함을 보였고, 지수평활법을 사용하는 ETS 역시 실험1과 실험2에서 데이터의 특성 파악이 충분하지 못하여 예측 능력이 떨어짐을 보였다.

본 연구의 기여점은 우리나라 환율의 예측에서 지금까지 사용하지 않는 머신러닝의 방법을 이용하여 전통적으로 단변량 예측에 많이 사용하는 ARIMA기법이나, ETS 기법과 비교를 하여 머신러닝 계열의 예측법이 약 20-30% 예측이 우수함을 보인 것이라고 평가된다. 이러한 진보된 예측을 바탕으로 우리나라의 무역수지의 계산, 수출, 수입의 계산, 재정 운용의 예측 등에서 많은 기여가 있을 것으로 예상된다.

본 연구가 앞으로 보완해야 할 점은 환율의 예측은 다변량 및 다변량의 연구도 활발한 만큼 단변량과 다변량까지도 연구의 범위를 넓힐 필요가 있다.

참고문헌

1. 구희조(1997). “장기 구매력평가와 환율예측 : 원/달러”,무역학회지, 제22권 제3호, pp.17-34.
2. 김영철, 외(2018). “스왑포인트 결정요소를 이용한 머신러닝 기반의 원/달러 환율 예측 모형에 관한 연구, 한국데이터정보과학회지, 제29권 제1호, pp.203-216.
3. 서종덕(2016), “데이터 마이닝 기법을 이용한 환율예측 GARCH와 결합된 랜덤 포레스트 모형” 산업경제연구, 제29권 제5호, pp.1607-1628.
4. 오문석, 이상근(2000), “환율결정모형의 원 / 달러환율 예측력 비교”, 경영학연구, 제29권 제4호, pp.711-722.
5. 이상래(2005),“환율예측모형의 비교연구”, 한국과학기술원.
6. 이재득(2021), “2021,투자와 수출 및 환율의 고용에 대한 의사결정 나무, 랜덤 포레스트와 그래디언트 부스팅 머신러닝 모형 예측”, 무역학회지, 제46권 제2호, pp.281-299.
7. 한정아, 안창호(2019), “시계열 모형을 이용한 원/달러 환율 예측모형 비교연구”, 융복합지식학회논문지, 제7권 제4호, pp.69-78.
8. 한태경 2010),“환율예측을 위한 합성모형 연구 : 시계열분석방법과 기계학습방법을 이용한 합성 모델”, KAIST 석사학위논문.
9. Ahmed, N. K., et. al.,(2010), ″An Empirical Comparison of Machine Learning Models for Time Series Forecasting” Econometric Reviews, 29(5-6), pp.594-621.
10. Crone, S. and Kourentzes, N.,(2010), “Feature selection for time series prediction - a combined filter and wrapper approach for neural networks”, Neurocomputing, 73(10 – 12), pp.1923-1936.
11. Gabriel, P. K., (2015), “On the application of Multivariate Times Series Models”, Journal of Physical Science and Technology. 8(10), pp.51-62.
12. Hannan EJ, Rissanen J ,(1982), “Recursive Estimation of Mixed Autoregressive-Moving Average Order.” Biometrika, 69(1), pp.81 – 94.
13. Hyndman, R.J., Akram, Md., and Archibald, B., (2008), ″The admissible parameter space for exponential smoothing models″. Annals of Statistical Mathematics, 60(2), pp.407 – 426.
14. Hyndman, R.J., Koehler, A.B., Snyder, R.D., and Grose, S. (2002), ″A state space framework for automatic forecasting using exponential smoothing methods″, International J. Forecasting, 18(3), pp.439 – 454.
15. Iwok, A., Okpe, A., (2016), “A Comparative Study between Univariate and Multivariate Linear Stationary Time Series Models”, American Journal of Mathematics and Statistics, 6(5): pp.203-212.
16. Kim, S. H., (2000), “Establishment of Optimal Artificial Neural Network Model and Exchange Rate Prediction Performance Analysis” Journal of Money and Finance, 14, pp.57-85.
17. Kim. J. H. (2001). “A comparative study on the Won/Dollar exchange rate forecasting performance of the artificial neural network model and ARIMA model, The Graduate School of Sogang University, Seoul.
18. Lee. H. J (2016). “Time series models based on relationship between won/dollar and won/yen exchange rate” Journal of the Korean Data & Information Science Society, 27, pp.1547-1555.
19. Lora et. al.(2007)″Electricity Market Price Forecasting Based on.
20. Makridakis S, et. al. (1982). “The Accuracy of Extrapolation (Time Series) Methods: Results of a Forecasting

Competition." Journal of Forecasting, 1, pp.111-153.
21. Martínez et. al.,(2019), "Time Series Forecasting with KNN in R: the tsfknn Package", R Journal, 9(2), pp.229-242.
22. Reilly D (2000),"The Autobox System." International Journal of Forecasting, 16(4), pp.531-533.
23. Weighted Nearest Neighbors Techniques", IEEE TRANSACTIONS ON POWER SYSTEMS, 22(3).
24. Zhang, G., Patuwo, B. Eddy and Hu, M. Y.(1988), "Forecasting with artificial neural networks:: The state of the art, International Journal of Forecasting, 14(1), pp.35-62.