Journal of the Korean Data & Information Science Society
2019, **30**(3), 655–669

http://dx.doi.org/10.7465/jkdi.2019.30.3.655
한국데이터정보과학회지

XGBoost 모형을 활용한 코스피 200 주가지수 등락 예측에 관한 연구[†]

하대우[1] · 김영민[2] · 안재준[3]

[13]연세대학교 정보통계학과 · [2]순천향대학교 빅데이터공학과

접수 2019년 4월 21일, 수정 2019년 5월 16일, 게재확정 2019년 5월 16일

요 약

주식시장은 자본주의 경제 체제를 나타내는 대표적인 시장으로써 금융시장에서 중요한 경제적 기능을 수행한다. 또한, 주식시장은 기업뿐만 아니라 개인 투자자들에게 자본을 획득할 수 있는 유용한 수단으로 여겨지고 있다. 이러한 인식 속에서 주가의 흐름을 정확하게 예측하는 것은 현재까지도 중요한 연구 과제로 남아있다. 최근 기계학습을 활용한 주가예측에 대한 연구들이 활발하게 진행되고 있는 가운데, 본 연구에서는 다양한 분야에서 우수성을 입증하고 있는 XGBoost (extreme gradient boosting) 모델을 주가 등락 예측에 활용하고자 한다. XGBoost 모델의 유용성을 입증하기 위해 시계열 데이터 분석에 강점을 가지고 있다고 알려진 LSTM (long-short term memory) 신경망과 전통적으로 가장 널리 사용되었던 시계열 분석 기법인 자기회귀모형의 예측 결과들을 비교 및 분석하였다. 실증분석 결과 주가등락 예측에 있어서 XGBoost 모델의 유용성을 확인할 수 있다.

주요용어: 기계학습, 자기회귀모형, 주가예측, LSTM 신경망, XGBoost 모델.

1. 서론

주식시장은 자본주의 경제체제를 나타내는 대표적인 금융시장으로써 경제주체인 정부, 기업, 가계 등에 중요한 경제적 기능을 수행한다. 또한 경제주체들이 기업의 소유권 지분을 거래할 수 있어 자본의 접근성을 높일 수 있다는 점에서 중요한 역할을 수행하고 있다. 이러한 배경으로부터 주식시장은 개인의 자산운용과 더불어 기업은 필요한 자본을 조달할 수 있는 기회의 장으로 자리매김하였다 (Yoon과 Hwang, 2008). 따라서 주가의 흐름을 정확하게 예측하는 것은 투자자들뿐만 아니라 경제, 사회 등 다양한 분야에서 중요한 과제로 인식하고 있지만, 일반적으로 주가는 다양한 요인에 의해 결정되고 그 요인이 주가에 영향을 미치는 경로와 정도가 매우 다양하고 복잡하여 주가변동의 근원을 정확히 파악하는 문제는 쉽지 않은 것으로 알려져 있다 (Lee, 2008). 특히 금리, 환율, 국제유가, 타 국가의 주가지수 및 경제 상황과 같은 거시적, 미시적인 요인들로 인한 주가의 불안정성 및 비정상성 특성 때문에 주가를 정확하게 예측하는 것은 쉽지 않다 (McNelis, 2005). 그러나 최근 시계열 데이터에 기계학습 (machine

[†] 이 논문은 2017학년도 연세대학교 원주캠퍼스 미래선도연구사업의 지원을 받아 작성된 것임(2017-52-0072).

[1] (26493) 강원도 원주시 연세대길 1, 연세대학교 정보통계학과, 석사과정.
[2] (31538) 충청남도 아산시 신창면 순천향로 22, 순천향대학교 빅데이터공학과, 조교수.
[3] 교신저자: (26493) 강원도 원주시 연세대길 1, 연세대학교 정보통계학과, 부교수.
E-mail: ahn2615@yonsei.ac.kr

learning)을 접목시킨 다양한 연구들이 진행되면서 높은 예측 성능들을 도출함으로써 주식 시장에서 기계학습의 유용성에 대한 가능성이 제시되고 있다.

Pai와 Lin (2005)는 ARIMA (autoregressive integrated moving average) 모델과 기계학습 모델 중 하나인 SVM (support vector machine)을 결합한 hybrid ARIMA 모델을 제안하였으며, 그 결과 제안모델이 단일모형보다 예측오차가 줄어드는 것을 확인하였다. 또한 Kim과 Shin (2007)은 하이브리드 인공신경망 (artificial neural networks) 모델을 제안하여 KOSPI 200 주가를 예측하였다. 그들은 ATNNs (the adaptive time delay neural networks)와 TDNNs (the time delay neural networks)의 아키텍처를 활용하였으며, 유전자알고리즘 (genetic algorithm)을 통해 모델을 결합하는 하이브리드 모델을 제안하였다. Yoon 등 (2017)은 유전자알고리즘을 인공신경망과 결합하여 유전자-신경망 모형을 제시하였고, 이를 단기 KOSPI 주가 예측에 적용하여 기존의 역전파 신경망 모형보다 유전자-신경망 모델이 유용함을 확인하였다.

또한, 최근에는 딥러닝 알고리즘과 같이 복잡성의 강도를 높힌 모형을 기반으로 하여 주가를 예측하고자 하는 연구들이 활발히 진행되고 있다. Selvin 등 (2017)은 RNN (recursive neural networks), LSTM (long-short term memory), CNN (convolutional neural networks)을 이용하여 주가를 예측하였으며 전통적인 시계열모형인 ARIMA 보다 우수한 성과를 도출하였다. Chou 등 (2018)는 SVR (support vector regression)모델의 복잡성과 효율성을 한층 향상시킨 LSSVR (least squares support vector regression) 모형 기반의 MetaFA-LSSVR 모델을 제안하였으며, NASDAQ, S&P 500, BIST 100 지수에 적용한 결과 우수한 예측성과를 보였다. Hwang (2018)은 오토인코더를 통해 변수를 추출하여 예측 정확도를 향상시켰으며, 기존의 단일 신경망보다 개선된 예측 정확도를 보인 심층 신경망 모델을 제안하였다. 이처럼 최근에는 딥러닝 알고리즘이 빠르게 발전하면서 고도의 비선형 모델링이 가능해지면서 주가를 예측하고자 하는 연구들이 활발히 진행되고 있다.

그럼에도 불구하고 앞서 소개한 연구모형들은 계산의 복잡성으로 인해 파라미터 조절에 대한 어려움이 존재한다는 한계점이 존재한다. 본 연구에서는 이러한 한계점을 보완할 수 있는 XGBoost (extreme gradient boosting) 모델을 활용하여 코스피 200 주가지수 등락을 예측하고자 한다. XGBoost 모형은 빠른 실행속도와 우수한 분류 성능 때문에 다양한 분야에서 연구되고 있다 (Guo 등, 2019). 실제로 XGBoost 모형은 training loss를 최소화하여 과적합을 방지할 수 있을 뿐만 아니라 병렬 및 분산 처리 기반으로 되어있어 기존의 gradient boosting 모델보다 학습 및 분류 속도가 빠르다는 강점이 있다고 알려져 있다 (Nobre와 Neves, 2019). 따라서 본 연구에서는 분류 성능과 분석 속도에 강점을 보이는 XGBoost 모형과 순환신경망이 가지고 있는 장기의존성 문제점을 보완한 LSTM 신경망 모델의 주가 예측성능을 비교하여 주식시장 분석에 있어서 XGBoost 모형의 효율성을 보이고자 한다.

본 논문의 구성은 다음과 같다. 2절에서는 본 연구에서 사용되는 분석 모델에 대한 이론적인 내용을 설명한다. 3절에서는 KOSPI 200 데이터 및 기술적 지표에 대한 설명과 실험 방법에 대하여 설명하고 LSTM 신경망 및 XGBoost 모형의 예측 결과를 비교한다. 4절에서는 연구 결과에 대해서 요약하고 향후 진행할 필요가 있다고 판단되는 추가 연구 내용을 제시한다.

2. 연구 배경

2.1. 예측모형의 성능평가 지표

본 연구에서는 주가 등락예측을 위한 분류 모델의 성능 평가를 위해 실제 주가의 등락과 예측 주가의 등락을 행렬로 나타내는 혼동행렬 (confusion matrix)과 민감도와 특이도 간의 관계를 그래프로 표현한 지표인 ROC 커브 (receiver operating characteristic curve)를 기반으로 예측성과를 비교한다. 혼동행

렬을 기반으로 정확도 (accuracy), 민감도 (sensitivity), 그리고 특이도 (specificity)를 도출하여 모형의 예측 성능을 평가하게 된다. Table 1은 혼동행렬의 예를 보여주고 있다.

Table 2.1 Example of confusion matrix used in this study

		Prediction	
		Down	Up
Actual	Down	TN	FP
	Up	FN	TP

Table 2.1에서 TN은 true negative를 의미한다. 이는 곧 실제 주가가 하락하고 모델 또한 주가를 하락으로 예측한 것이다. FN은 false negative를 의미하고 실제 주가가 상승하는 반면 모델은 하락으로 예측한 경우를 뜻한다. FP는 false positive로써 실제 주가는 하락하는 반면 모델은 주가를 상승으로 예측하는 경우를 뜻한다. TP는 true positive를 의미하며 실제 주가가 상승하고 모델 또한 주가의 상승을 예측하는 경우이다.

식 (2.1) ∼ (2.3)은 혼돈행렬을 바탕으로 도출되는 예측모형의 성과 지표들을 나타낸다. 정확도는 예측모델이 예상한 주가의 등락과 실제 해당 시점 주가의 등락이 일치한 확률을 뜻한다. 민감도는 실제 주가가 상승했을 때, 예측모델이 해당 시점의 주가가 상승할 것이라고 예측할 확률을 뜻한다. 반대로 특이도는 실제 주가가 하락했을 때, 예측모델이 해당 시점의 주가가 하락할 것이라고 예측할 확률을 뜻한다.

$$\text{Accuracy} = \frac{\text{TN} + \text{TP}}{\text{TN} + \text{FP} + \text{FN} + \text{TP}}, \tag{2.1}$$

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{2.2}$$

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}}. \tag{2.3}$$

ROC 커브는 머신러닝 분야에서 모델의 분류 성능 파악을 위해 자주 활용되는 지표이며, ROC 커브를 통해 모든 분류 임계값에서 분류 모델의 성능을 확인할 수 있다(Bradley와 Andrew, 1997). ROC 그래프의 x축은 1-특이도로 설정하고, y축을 민감도로 설정하여 ROC 커브의 면적을 계산하고 분류 성능을 평가한다. ROC 그래프 위에 그려지는 곡선의 면적이 넓을수록 모델의 성능이 좋다고 판단한다. 이때 ROC 그래프의 면적을 AUC (area under ROC curve)라고 표현한다. AUC는 ROC 커브의 면적을 파악하기 쉽게 나타낸 수치이며, 전체적인 민감도와 특이도의 상관관계를 보여주는 지표이다. AUC는 0.5와 1사이의 값을 가지며 1에 가까운 값을 가질수록 모델의 분류 성능이 좋다고 판단한다.

2.2. 자기회귀모형 (autoregressive model: AR 모형)

자기회귀모형은 현재의 시계열 자료가 과거 자료의 값들로부터 설명된다는 대표적인 시계열 분석 모형이다.

$$Y_t = \alpha_0 + \alpha_1 Y_{t-1} + \alpha_2 Y_{t-2} + \cdots + \alpha_p Y_{t-p} + \epsilon_t. \tag{2.4}$$

식 (2.4)에서 $\alpha_1, \alpha_2, \cdots, \alpha_p$는 자기상관계수를 의미한다. ϵ_t는 백색잡음 (white noise)을 의미한다. 현재 시점 t의 시계열 값 (y_t)은 $t-1$ 시점으로부터 α_1만큼, $t-2$ 시점으로부터 α_2만큼 영향을 받는다고 해석할 수 있다. 과거 p시점까지의 값들이 y_t에 영향을 준다고 가정할 경우, 자기회귀 모형이라고 부르며 $AR(p)$모형으로 표현된다.

2.3. LSTM (long-short term memory) 신경망

STM 신경망은 순환신경망의 한 종류로서 현 시점을 포함한 과거의 정보를 이용하여 미래 시점의 값을 예측하는 원리로 설계되었으며, 기존의 순환신경망이 갖고 있는 장기 의존성 문제를 해결한 알고리즘이다 (Hochreiter와 Schmidhuber, 1997).

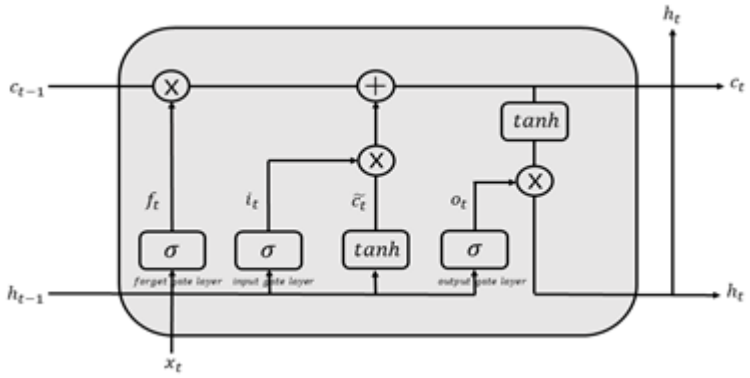


Figure 2.1 Hidden layer structure of LSTM at time t

Figure 2.1은 LSTM에서 t시점의 은닉층 모듈을 나타내고 있다. C_t는 셀 스테이트(cell state)로 t시점의 정보를 지니고 있는데, t시점의 은닉층에서는 $t-1$시점의 정보 C_{t-1}를 입력받아 정보 선별과정(forget process)과 새로운 자료로부터 새로운 정보를 추출하여 다시 업데이트하는 과정 (update process)을 거쳐 t시점의 정보인 C_t를 도출하게 된다. t시점의 셀 스테이트인 C_t로부터 t시점의 결과값을 도출하고, $t+1$시점에 정보를 제공함으로써 LSTM은 과거정보의 지속성을 고려할 수 있게 된다.

LSTM은 크게 네 가지의 과정을 거친다. 첫 번째는 셀 스테이트에서 버리고자하는 정보를 선택하는 과정이다. 이 과정은 식 (2.5)에서 나타내고 있으며, 포겟 게이트 레이어 (forget gate layer)라는 시그모이드 레이어에서 이루어진다. 이 게이트는 f_t로 표현할 수 있고, h_{t-1}과 x_t를 입력값으로 가지며 1을 출력하면 이 값을 유지하고, 0을 출력하면 이 값을 제거한다. 이때 식 (2.5)에서 W_f와 b_f는 LSTM 신경망의 매개변수를 의미하며 σ는 시그모이드 활성화 함수를 의미한다.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f). \tag{2.5}$$

두 번째는 새로운 정보를 셀 스테이트에 저장할지 결정하는 과정이다. 이는 다시 두 과정으로 이루어진다. 우선 인풋 게이트 레이어 (input gate layer)에서 어떤 값을 입력할지 결정한다. 다음으로 tanh 레이어에서의 $\tilde{C}_t$값은 셀 스테이트에 더해질 수 있는 새로운 후보값을 만들어낸다. 최종적으로 두 가지 과정을 거쳐 나온 값을 합쳐 다음 단계에 영향을 준다. 이 두 과정은 식 (2.6)과 식 (2.7)에서 보여주고 있다.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i), \tag{2.6}$$

$$\tilde{C} = tanh(W_C \cdot [h_{t-1}, x_t] + b_C). \tag{2.7}$$

세 번째는 이전 단계의 셀 스테이트 (C_{t-1})를 t시점의 셀 스테이트 (C_t)로 업데이트하는 과정이다. 첫 번째 단계에서 출력한 이전 셀 스테이트를 f_t로 곱하여 버리고자 하는 정보를 버린다. 그 다음, 두 번째 단계에서 출력한 $i_t \cdot \tilde{C}_t$를 $\oplus$과정을 거쳐 더해 새로운 후보 값이 기존 값에 영향을 주게 된다. 이 과정은 식 (2.8)에서 보여주고 있다.

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t. \tag{2.8}$$

마지막으로 어떤 값을 출력할지 결정하는 과정이다. 이는 식 (2.9)와 식 (2.10)에서나타내고 있는데 우선 아웃풋 게이트 레이어 (output gate layer)를 통해 어떤 값을 출력할지 결정한다. 다음으로 tanh레이어에서 -1과 1 사이의 값을 출력한다. 그 다음 원하는 값만 출력값 (h_t)으로 반영하기 위해 아웃풋 게이트 레이어의 출력값과 tanh레이어의 출력값을 곱한다.

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o), \tag{2.9}$$

$$h_t = o_t \cdot tanh(C_t). \tag{2.10}$$

위의 총 네 가지의 과정을 거쳐 t시점의 셀 스테이트를 통해 t시점의 결과값을 도출하고 다음 $t+1$시점에 정보를 제공하여 과거 시점의 정보들을 지속적으로 유지한다.

2.4. XGBoost (extreme gradient boosting) 모형

XGBoost 모형은 약한 분류기를 순차적으로 개선해나감으로써 보다 강력한 분류기를 생성하는 트리 모형에 그래디언트 부스팅 (gradient boosting)기법을 적용한 앙상블 알고리즘 중의 하나이다 (Chen과 Guestrin, 2016).

$$\hat{y_i} = \sum_{k=1}^{K} f_k(x_i), \quad f_k \in F. \tag{2.11}$$

식 (2.11)은 트리의 앙상블 모델을 나타내는 식으로서 K는 트리의 개수, F는 모든 가능한 CART (classification and regression trees)의 집합을 의미한다. f_k는 각 독립된 트리와 각 잎의 가중치에 대응된다. 각 잎의 점수를 합산, 비교하여 최종 예측을 한다.

$$Obj = \sum_{i}^{n} l(y_i, \hat{y_i}) + \sum_{k=1}^{K} \Omega(f_k). \tag{2.12}$$

식 (2.12)은 XGBoost 모델을 나타내는 식이다. 첫 번째 $l(\hat{y_i}, y_i)$는 손실함수로서 예측값과 실제값의 차이를 나타낸다. 두 번째 $\Omega(f_k)$는 정규화 항으로 모델의 복잡도를 조절하여 과적합을 방지한다.

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda||w||^2. \tag{2.13}$$

식 (2.13)는 정규화 항을 다르게 표현한 식이다. γT는 트리의 잎 개수를 의미하고, $\frac{1}{2}\lambda||w||^2$는 잎의 점수를 의미한다. 다시 말해 잎의 개수와 점수가 모델의 복잡도를 결정한다는 것을 의미한다.

$$Obj(t) = \sum_{i}^{n} l(y_i, \hat{y}^{(t-1)} + f_t(x_i)) + \Omega(f_t). \tag{2.14}$$

식 (2.14)에서 $\hat{y}^{(t-1)}$을 $t-1$시점의 예측값이라고 했을 때, 식 (2.12)은 식 (2.14)처럼 다시 표현할 수 있다. 식 (2.14)에서 f_t를 조절하면서 정규화 항과 손실함수를 최소화하여 오차를 줄여나간다.

$$Obj(t) = \sum_{i}^{n} [l(y_i, \hat{y_i}^{(t-1)}) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)] + \Omega(f_t), \tag{2.15}$$

$$Obj = \sum_{i}^{n} [g_i f_i(x_i) + \frac{1}{2} h_i f_t^2(x_i)] + \Omega(f_t). \tag{2.16}$$

식 (2.15)는 목적함수가 복잡해지는 것을 방지하기 위해 테일러 급수를 사용하여 목적함수를 근사화한 식이다. 위 식에서 g_i는 손실함수를 $t-1$시점에서 1차 미분, h_i는 손실함수를 $t-1$시점에서 2차 미분한 값으로, 식 (2.16)은 식 (2.15)를 미분 형태로 표현한 식이다.

$$Obj = \sum_{j=1}^{T} [(\sum_{i \in I_j} g_i) w_j + \frac{1}{2} (\sum_{i \in I_j} h_i + \lambda) w_j^2] + \gamma T, \tag{2.17}$$

$$w_j^* = -\frac{G_j}{H_j + \lambda}. \tag{2.18}$$

식 (2.17)은 전체 과정을 목적함수에 적용하여 w_j를 기준으로 정렬한 결과이다. 식 (2.17)에서 I_j는 잎 노드인 j의 모든 데이터 개수를 의미한다. G_j와 H_j를 각각 g_j, h_j의 합으로 가정했을 때, 최적의 가중치 w_j^*는 식 (2.18)과 같이 표현될 수 있다.

$$Obj = -\frac{1}{2} \sum_{j=1}^{T} \frac{G_j^2}{H_j + \lambda} + \gamma T. \tag{2.19}$$

식 (2.19)는 하나의 트리 구조에 대한 최적의 목적함수를 의미한다. 따라서 이 식을 통해 트리의 구조가 얼마나 좋은지 평가하는 기준이 된다.

$$Gain = \frac{G_L^2}{H_{L+}\lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{(G_L + G_R)^2}{H_L + H_R + \lambda} - \gamma. \tag{2.20}$$

식 (2.20)은 한 트리의 특정 깊이에서 가지를 쳤을 때 얻는 정보획득량을 뜻한다. 이때 임의로 기준을 정하고 정보획득량이 최대가 되도록 가치를 치는 트리를 생성한다. 이러한 과정을 반복하여 정보획득량이 가장 큰 트리를 조합하여 부스팅을 진행한다. 그 결과로 T개의 트리가 조합된 최적의 분류 모델을 얻는다.

3. 실증 분석

3.1. 실험데이터 및 변수 소개

본 연구는 한국종합주가지수 200 (KOSPI 200) 등락 예측을 위해 통계적 모형인 자기회귀모형과 최근 기계학습 분야에서 주목받고 있는 LSTM, 그리고 XGBoost 모형을 활용한다. 실험에 사용된 KOSPI 200 주가지수의 데이터는 2010년 1월 29일부터 2017년 12월 28일까지이며 일별 데이터를 사용하였다. 모형 구축에 사용된 학습 기간은 2010년 1월 29일부터 2015년 8월 7일까지이며, 테스트 기간은 2015년 8월 10일부터 2017년 12월 28일까지이다. 즉, 예측모델을 구축하기 위해 총 데이터의 70% (1,371개)를 학습 데이터로 활용하였고, 구축된 모델의 성능을 검증하기 위해 나머지 30% (587개)는 테스트 데이터로 사용하였다.

LSTM 신경망과 XGBoost를 활용한 등락 예측모형을 구축하기 위해 사용된 입력변수(또는 독립변수)는 대표적인 기술적 지표인 SMA (simple moving average), EMA (exponential moving average), stochastic D%, stochastic K%, RSI (relative strength index), MACD (moving average convergence and divergence), 이격도 (disparity)이며 Table 3.1에서 보여주고 있다 (Shin 등, 2017). 그리고 출력변수 (또는 종속변수)는 연속형 자료인 KOSPI 200 주가지수 종가 (Close)이다. 여기서 단순이동평균 (SMA)은 특정 기간 동안 주가가 가지고 있는 방향성을 수치화시킨 대표적인 기술적 지표이며, 특정일 (t)에서 n일간의 이동평균을 의미하며 종가를 바탕으로 계산된다. 지수이동평균 (EMA)는 시간에 따라 가중치를 부여하는 방법으로 최근 시점에 더 높은 가중치를 부여하는 이동평균 방법이다. stochastic 기술적 지표는 일정 기간 동안 종가를 바탕으로 주가의 상승, 하락 강도를 나타내는 단기적 지표로 알려져 있다. 또한, stochastic K%의 Low는 m-기간 동안의 가장 낮은 가격을, High는 m-기간 동안 가장 높은 가격을 의미한다. RSI 는 현재의 주가 추세 강도를 백분율로 표현되어 상대적 추세를 측정할 수 있는 기술적 지표로 알려져 있다. 여기서 AU_t^n는 n일 동안 주가 상승 폭의 합계 평균을 의미하고, AD_t^n는 n일 동안 주가 하락 폭의 합계 평균을 의미한다. MACD 기술적 지표는 단기이동평균 값과 장기이동평균 값과의 차이를 이용하여 주가의 추세를 파악할 수 있는 기술적 지표이다. 마지막으로 이격도는 주가와 이동평균 값과의 차이를 측정하는 기술적 지표이다.

Table 3.1 Technical indicators used for the analysis

Technical indicators	Formula
SMA	$\frac{C_t+C_{t+1}+\cdots+C_{t+n}}{n}$
EMA	$Close_i \times EP + Close_{i-1} \times (1-EP)$
Stochastic K%	$\frac{Close_t - Low(m)}{High(m)-Low(m)} \times 100$
Stochastic D%	$\frac{\sum_{i=0}^{n-1} K_{t-i}}{10}\%$
RSI	$\frac{AU^{n}t}{AU^{n}t+AD^{n}t} \times 100$
MACD	$\sum_{i=t-9}^{t}(EMA(12)_t - EMA(26)_t)$
Disparity	$\frac{Close_t}{SMA(n)} \times 100$

3.2. 분석 방법 및 실증분석 결과

KOSPI 200 주가지수 등락을 예측하기 위해 본 연구에서는 일별 종가를 출력 변수로 사용하였으며, LSTM 신경망과 XGBoost 모형 구축을 위한 입력변수들은 Table 3.1에서 언급된 7가지 기술적 지표들을 사용하였다. 또한 자기회귀모형의 경우 연속형 출력 변수 값 예측을 위한 모형임을 감안하여 t 시점의 추정 지수가 $(t-1)$ 시점의 지수보다 크면 1 (상승), 작으면 0 (하락)으로 변환한 뒤 실제 지수 등락과 비교하여 예측 정확도를 계산하였다.

또한, 시계열 예측이라는 본 연구의 특성에 맞추어 의료, 날씨, 금융 분야와 같은 다양한 시계열 분야에서 모형의 높은 예측 성능을 위해 널리 사용되는 슬라이딩 윈도우 (sliding window) 방법론을 적용하였다 (Kim 등, 2018). 슬라이딩 윈도우 방법론은 과거 데이터를 누적하여 다음 시점의 값을 예측하는 방법이다 (Yahyia 등, 2015). 이때 누적된 과거 데이터를 윈도우 (window)로 정의하며, 윈도우를 일정 간격으로 이동시키면서 다음 시점의 값을 예측한다. Figure 3.1은 본 연구에서 적용된 슬라이딩 윈도우 방법을 시각적으로 보여주고 있으며, 윈도우의 크기 (window size)인 h를 5로 설정한 경우이다. $h = 5$라면 t 시점부터 $t + 4$ 시점까지의 데이터를 누적 데이터 (input)로 설정하여 $t + 5$ 시점의 값 (target)을 예측한다. 그다음, 가장 오래된 t 시점의 데이터를 제외하고 최근인 $t + 5$ 시점의 데이터를 누적시켜 윈도우의 크기를 5로 유지하면서 다음 $t + 6$ 시점의 값을 예측한다. 위 과정을 반복하면서 윈도우가 학습데이터를 모두 사용할 때까지 이동한다.

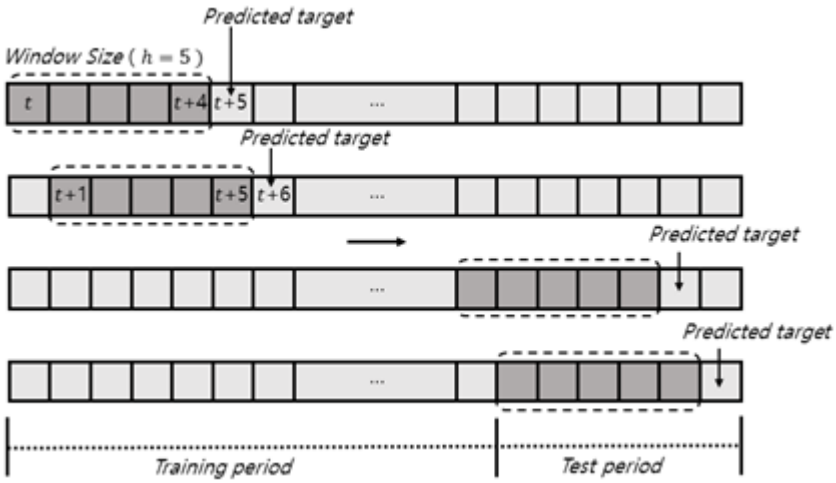


Figure 3.1 Sliding window scheme when the window size is set to 5

3.2.1. 자기회귀모형 분석 결과

Table 3.2은 각 window size 마다 ($h = 1, 5, 10, 20$ 일) 자기회귀 모형을 이용한 예측모형의 정확도를 나타내주고 있다. 자기회귀모형의 경우 $h = 1$ 이면 AR(1) 모형이며, $h = 5$ 이면 AR(5) 모형에 해당된다고 할 수 있다. 또한 자기회귀모형으로 예측 모형 구축 시 기술적 지표는 고려되지 않는다.

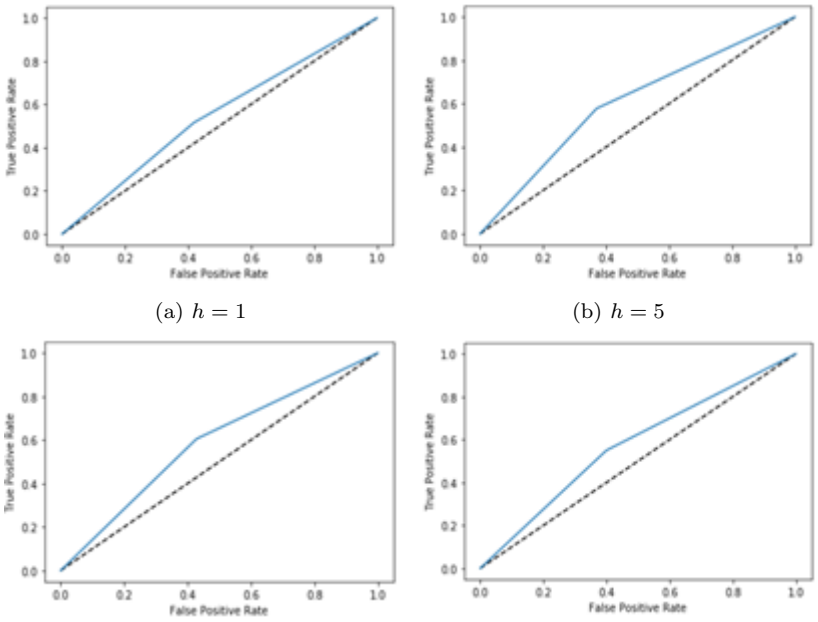
테스트기간 동안 $h = 5$ 일 때 예측 정확도는 60.17%로 가장 높게 측정되었다. 즉, $t - 5$ 시점까지의 정보를 사용하였을 때 t 시점의 KOSPI 200 주가지수의 등락을 약 60%로 예측 가능함을 의미한다. 민감도의 경우는 $h = 20$ 일 때 73.33%, 특이도의 경우는 $h = 1$ 일 때 58.11%로 가장 높게 나왔다. 또한, 전반적으로 특이도보다 민감도가 높게 나옴을 볼 수 있다. 이는 주가의 상승을 하락보다 더 잘 예측한다는 의미를 가지게 되는데, 일반적인 주식예측 연구의 결과와 비슷한 양상을 보이는 결과라고 할 수 있다. 그리고 window size가 증가할수록 민감도가 증가하며, 특이도는 감소하는 경향을 보이고 있다. 이는 이전 시점의 정보가 증가할수록 주가 상승과 관련한 예측에 도움을 주지만 주가 하락 예측에는 오히려 부정적인 영향을 미친다고 해석할 수 있다.

Figure 3.2는 자기회귀모형의 window size에 따른 ROC (receiver operating characteristic) 곡선을

Table 3.2 Predictive performances (%) of AR model during test period

Window size	Accuracy	Sensitivity	Specificity
$h = 1$	54.51	51.55	58.11
$h = 5$	60.17	64.91	55.74
$h = 10$	59.32	71.88	44.44
$h = 20$	56.67	73.33	40.00

나타낸 그래프이다. $h = 1$인 경우 ROC 곡선의 아래 면적에 해당하는 AUC (area under the curve) 값은 0.54, $h = 5$인 경우는 0.58, $h = 10$인 경우는 0.59 그리고 $h = 20$일 때는 0.58로, $h = 10$일 때 가장 높은 AUC 값을 보였다. 하지만 전체적으로 window size의 변화와 무관하게 AUC 값이 비슷한 것을 확인하였으며, 낮은 AUC 값들을 토대로 자기회귀모형의 경우 주가 등락 예측모형으로 적합하지 않음을 알 수 있다.



(a) $h = 1$ (b) $h = 5$

(c) $h = 10$ (d) $h = 20$

Figure 3.2 ROC curves of AR models for each window size (h)

3.2.2. LSTM 신경망 분석 결과

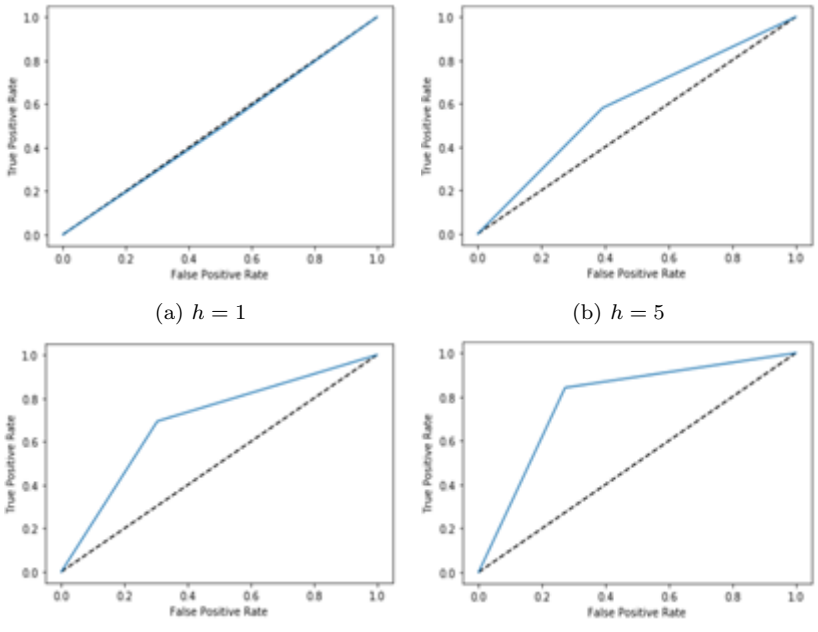
LSTM 신경망을 이용한 주가예측 모형을 구축하기 위해 먼저 입력변수들의 값을 [0, 1]로 표준화하였다. 또한, 본 실험에 적용된 LSTM 신경망의 파라미터는 epoch 수를 150으로 고정하였고, batch size는 10, 활성화 함수는 Relu, 그리고 optimizer는 adam으로 적용하였으며, 각 window size 마다 모두 동일하게 적용하였다. Table 3.3은 LSTM 신경망을 이용한 주가예측 결과를 보여주고 있다. 정확도의 경우 $h = 20$ 일 때 80%로 가장 높은 정확도를 보였다. 또한, 민감도와 특이도 모두 정확도와 마찬가지로

$h = 20$ 일 때 84.21%와 72.73%로 가장 높은 수치를 보였다. 그리고 window size가 늘어날수록 예측모형의 성과 지표들이 향상되는 경향을 보이고 있는데, 이는 LSTM 신경망이 기존 순환신경망의 장기의존성을 해결할 수 있는 알고리즘임을 고려한다면 당연한 결과라고 판단된다. LSTM 신경망은 포겟 게이트 레이어를 통해 예측을 위한 이전 시점의 정보들을 선택하기에 window size가 늘어날수록 주가 예측을 위한 영향력 있는 정보들이 모형에 선별적으로 추가될 수 있기 때문이다.

Table 3.3 Predictive performances (%) of LSTM model during test period

Window size	Accuracy	Sensitivity	Specificity
$h = 1$	48.72	52.48	44.15
$h = 5$	59.32	58.06	60.71
$h = 10$	69.49	69.44	69.56
$h = 20$	80.00	84.21	72.73

Figure 3.3은 LSTM 신경망을 이용한 예측모형의 ROC 곡선들을 나타낸 그래프이다. $h = 1$인 경우 ROC 곡선의 AUC 값은 0.49, $h = 5$인 경우는 0.59, $h = 10$인 경우는 0.69 그리고 $h = 20$일 때는 0.78로 도출되었다. 전반적으로 window size가 늘어남에 따라 AUC 값이 증가하고 있다. 이는 Table 3.3의 결과와 같은 맥락으로 해석 가능하며, 특히 $h = 20$인 경우 주가예측에 있어 유용한 모형임을 알 수 있다.



(a) $h = 1$ (b) $h = 5$

(c) $h = 10$ (d) $h = 20$

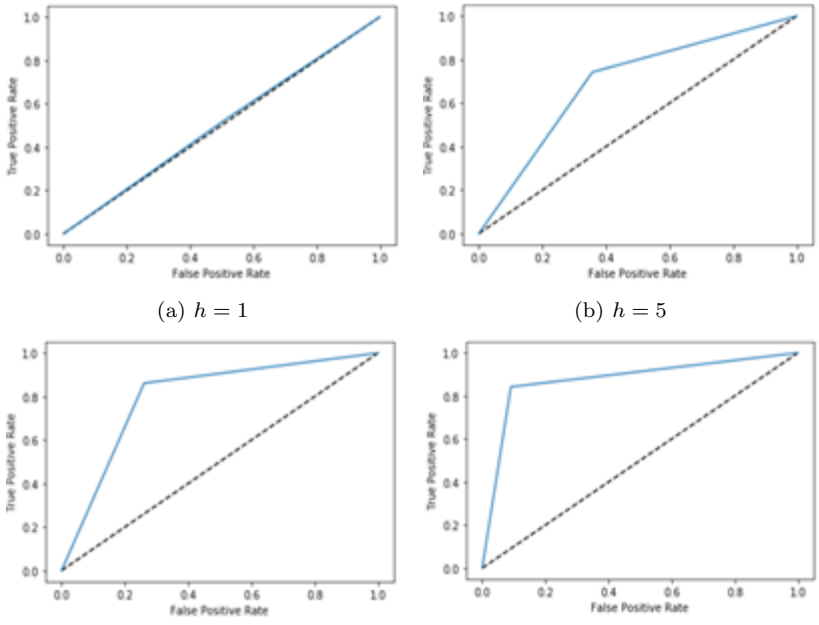
Figure 3.3 ROC curves of LSTM models for each window size (h)

3.2.3. XGBoost 분석 결과

LSTM 신경망과 마찬가지로 XGBoost 모형 적용을 위해 입력변수들의 값을 [0, 1]로 표준화하였다. XGBoost 모형의 파라미터는 학습률 (learning rate) 0.05, Max depth는 5, N estimators는 300개를 사용하였으며, window size와 상관없이 모두 동일하게 지정하였다. Table 3.4는 XGBoost를 이용한 주가 예측 결과를 보여주고 있다. 그 결과 정확도의 경우, $h = 20$일 때 86.67%로 가장 높은 정확도를 보였다. 민감도의 경우 $h = 10$일 때 86.11%로 가장 높게 나왔으며 특이도는 $h = 20$일 때 90.91%로 가장 높게 나왔다. 전반적으로 window size가 증가함에 따라 예측성과 지표들도 향상됨을 알 수 있다. 또한 자기회귀모형의 결과와 마찬가지로 민감도가 특이도보다 다소 높게 도출됨에 따라 주가의 상승을 하락보다 더 잘 예측한다는 것을 알 수 있다. 다만 $h = 20$인 경우 매우 높은 특이도를 보여주고 있는데, 실제 투자자들에게 주가의 하락 방향에 대한 확률 높은 정보를 제공할 수 있다는 측면에서 매우 의미가 있다고 할 수 있다.

Table 3.4 Predictive performances (%) of XGBoost model during test period

Window size	Accuracy	Sensitivity	Specificity
$h = 1$	50.94	52.79	48.68
$h = 5$	69.49	74.19	64.28
$h = 10$	81.36	86.11	73.91
$h = 20$	86.67	84.21	90.91



(a) $h = 1$ (b) $h = 5$

(c) $h = 10$ (d) $h = 20$

Figure 3.4 ROC curves of XGBoost models for each window size (h)

Figure 3.4는 XGBoost모델의 window size에 따른 ROC 곡선을 보여주고 있다. $h = 1$인 경우 AUC

값은 0.51, $h = 5$인 경우는 0.69, $h = 10$인 경우는 0.80, 그리고 $h = 20$일 때는 0.87로 이 중 가장 높은 AUC 값을 보였다. 전반적으로 window size가 커짐에 따라 AUC 값이 증가하는 것을 알 수 있으며, LSTM 모형과 마찬가지로 $h = 20$인 경우 가장 좋은 예측모형을 구축할 수 있다.

3.2.4. 모형간 예측 성과 비교

Table 3.5는 본 연구에서 이용한 자기회귀모형, LSTM 신경망, 그리고 XGBoost 모형의 테스트 기간 동안 정확도, 민감도, 특이도, 그리고 AUC 값들을 window size ($h = 1$, 5, 10, 20일)에 따라 비교한 표이다. 전통적인 시계열 모형인 자기회귀모형은 다른 기계학습 비교모형들보다 예측성과가 다소 저조한 것으로 나타났다. 이는 높은 복잡성과 비선형적인 특성을 가진 주식시장의 등락예측을 위해 전통적인 시계열모형보다 기계학습 모형들이 더 적합한 모형임을 보여주는 결과이다. 또한, 전반적으로 XGBoost 모델의 예측성과가 LSTM 신경망의 예측성과보다 높은 것을 확인할 수 있었다. LSTM 신경망과 XGBoost 모형 모두 window size가 증가함에 따라 정확도가 높아짐을 확인할 수 있었는데, XGBoost 모형은 LSTM 신경망보다 정확도뿐만 아니라 민감도, 특이도 모두 예측성과가 개선됨을 확인할 수 있었다. 이를 통해 XGBoost 모형이 LSTM 신경망보다 상승과 하락 방향성 예측에 있어 더 나은 결과를 도출함을 확인할 수 있었다.

Table 3.5 Prediction performance (%) and AUC of all analysis methods during test period

window	Autoregressive model				LSTM				XGBoost			
size	Acc	Sen	Spec	AUC	Acc	Sen	Spec	AUC	Acc	Sen	Spec	AUC
$h = 1$	54.51	51.55	58.11	0.54	48.72	52.48	44.15	0.49	50.94	52.79	48.68	0.51
$h = 5$	60.17	64.91	55.74	0.58	59.32	58.06	60.71	0.59	69.49	74.19	64.28	0.69
$h = 10$	59.32	71.88	44.44	0.59	69.49	69.44	69.56	0.69	81.36	86.11	73.91	0.80
$h = 20$	56.67	73.33	40.00	0.58	80.00	84.21	72.73	0.78	86.67	84.21	90.91	0.87

* Acc: accuracy, Sen: sensitivity, Spec: specificity

4. 요약 및 결론

본 연구는 주가예측에 있어서 통계적 기반의 시계열 분석모형인 자기회귀모형과 기존의 순환신경망의 문제점을 보완한 LSTM 신경망, 그리고 CART 기반의 부스팅 앙상블 알고리즘인 XGBoost 모형의 활용 가능성에 대하여 비교 분석하였다. 이를 위해 KOSPI 200 주가지수 일별 데이터를 바탕으로 실증 분석하여 각 모형들의 예측성과를 비교하였다. 분석 결과 자기회귀모형 보다 LSTM 신경망과 XGBoost 모형의 예측성과가 더 좋은 것으로 나타났다. XGBoost 모형과 LSTM 신경망을 이용한 주가예측의 경우 window size가 증가함에 따라 정확도, 민감도 및 특이도가 모두 증가하는 추세를 가지는 것을 확인할 수 있었는데, 이는 기계학습 적용 시 단기간보다는 장기간의 정보를 이용하여 주가를 예측하는 것이 더욱 효과적임을 말해준다. 또한, 본 연구에서 고려한 세 가지 예측모형 중 XGBoost모델이 주가 예측에 있어서 가장 적합한 모형으로 선정되었다. 이를 통해 기존 연구들에서 시계열 데이터 예측에 있어 좋은 성과를 보인다고 알려진 LSTM 신경망과 비교하여 XGBoost 모형 또한 충분히 유용한 적용 모형이 될 수 있음을 확인할 수 있었다. 따라서 XGBoost 모형을 주가 데이터에 적용하여 여러 하이퍼파라미터들을 잘 도출할 수 있다면, 주식시장 분석에 있어 효율적인 모형으로 평가받을 수 있을 것이다.

본 연구의 주된 목적은 분류문제에 있어서 강력한 성능을 보이는 XGboost 모형을 주식시장 분석에 적용하여 그 유용성을 검증하는데 있다. 실증분석을 통해 도출된 결과를 통해 주식가격 예측에 있어서 XGBoost 모형의 유용성을 알 수 있었다. 그럼에도 불구하고 본 연구가 가지고 있는 한계점이 존재한

다. 첫째, 제한된 기술적 지표를 고려했다는 점, 둘째, window size 마다 같은 파라미터를 설정했다는 점, 셋째, KOSPI 200 지수만을 대상으로 실증분석이 진행된 점 등이다. 또한, 비정상성을 보이는 주가의 특성을 고려했을 때, 로그수익률을 사용한다면 자기회귀모형의 경우는 KOSPI 200지수를 사용한 본 연구의 결과보다 높은 예측 성능을 보일 것으로 판단된다. 향후 연구에서는 이러한 한계점들이 보완된다면 본 연구에서 검증하였던 XGBoost 모형의 유용성을 더욱 강건하게 뒷받침할 수 있을 것이라 생각된다.

References

Basak, S., Saha, S, Kar., Saha, S., Khaidem, L. and Dey, S. R. (2019). Predicting the direction of stock market prices using tree-based classifiers. *The North American Journal of Economics and Finance*, **47**, 552-567.

BenYahmed, Y., Bakar, A. A., RazakHamdan, A., Ahmed, A. and Abdullah, S. M. S. (2015). Adaptive sliding window algorithm for weather data segmentaion. *Journal of Theoretical and Applied Information Technology*, **80**, 322-333.

Bradley, A. P. (1997). The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Queensland University of Technology*, **30**, 1145-1159.

Chatzis, S. P., Siakoulis, V., Petropoulos, A., Stavroulakis, E. and Vlachogiannakis, N. (2018). Forecasting stock market crisis events using deep and statistical machine learning techniques. *Expert Systems with Applications*, **112**, 353-371.

Chen, T. and Guestrin, C. (2016). XGBoost: A scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785-794.

Chou, J. S. and Nguyen, T. K. (2018). Forward forecast of stock price using sliding-window metaheuristic-optimized machine-learning regression. *IEEE Transactions on Industrial Informatics*, **14**, 3132-3142.

Guo, J., Yang, L., Bie, R., Yu, J., Gao, Y., Shen, Y. and Kos, A. (2019). An XGBoost-based physical fitness evaluation model using advanced feature selection and Bayesian hyper-parameter optimization for wearable running monitoring. *Computer Networks*, **151**, 166-180.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, **9**, 1735-1780.

Hwang, H. (2018). Daily stock price forecasting using deep neural network model. *Journal of the Korea Convergence Society*, **9**, 39-44.

Jung, J. H. and Min, D. K. (2013). The study of foreign exchange trading revenue model using decision tree and gradient boosting. *Journal of the Korean Data & Information Science Society*, **24**, 161-170.

Kazem, A., Sharifi, E., Hussain, F, K., Saberi, M. and Hussain, O. K. (2013). Support vector regression with chaos-based firefly algorithm for stock market price forecasting. *Applied Soft Computing*, **13**, 947-958.

Kim, H. J. and Shin, K. S. (2007). A hybrid approach based on neural networks and genetic algorithms for detecting temporal patterns in stock markets. *Applied Soft Computing*, **7**, 569-576.

Kim, Y. C., Lee, H. J., Kim, J. W. and Ahn, J. J. (2018). A study on machine learning-based KRW/USD exchange rate prediction model using swap point determinants: Focused on optimal structure finding in multi layer perceptron. *Journal of the Korean Data & Information Science Society*, **29**, 203-216.

Lee, H. (2008). A combination model of multiple artificial intelligence techniques based on genetic algorithms for the prediction of Korean stock price index (KOSPI). *Entrue Journal of Information Technology*, **7**, 33-43.

Lee, W. (2017). A deep learning analysis of the KOSPI's directions. *Journal of the Korean Data & Information Science Society*, **28**, 287-295.

McNelis, P. D. (2005). *Neural networks in finance: Gaining the predictive edge in the market*, Elsevier Inc, New York.

Nobre, J. and Neves, R. F. (2019). Combining principal component analysis, discrete wavelet transform and XGBoost to trade in the financial markets. *Expert Systems with Applications*, **125**, 181-194.

Pai, P. F. and Lin, C. S. (2005). A hybird ARIMA and support vector machines model in stock price forecasting. *Omega*, **33**, 497-505.

Selvin, S., R, Vinayakumar, R., Gopalakrishnan, E.A., Menon, V. K. and Soman, K. P. (2017). Stock price prediction using LSTM, RNN and CNN-sliding window model. *2017 International Conference on Advances in Computing, Communications and Informatics*, 1643-1647.

Shin, D. H., Choi, K. H. and Kim, C. B. (2017). Deep learning model for prediction rate improvement of stock price using RNN and LSTM. *The Journal of Korean Institute of Information Technology*, **15**, 9-16.

Yoon, J. H. and Hwang, G. H. (2008). The effect of macroeconomic factors on Korean stock prices. *East and Central Asia Economic and Business Association*, **19**, 63-82.

Yoon, Y. C., Jo, N. R. and Lee, S. D. (2017). Forecasting algorithm using an improved genetic algorithm based on backpropagation neural network model. *Journal of the Korean Data & Information Science Society*, **28**, 1327-1336.

Journal of the Korean Data & Information Science Society
2019, **30**(3), 655–669

http://dx.doi.org/10.7465/jkdi.2019.30.3.655
한국데이터정보과학회지

A study on KOSPI 200 direction forecasting using XGBoost model[†]

Dae Woo Hah[1] · Young Min Kim[2] · Jae Joon Ahn[3]

[13]Department of Information and Statistics, Yonsei University
[2]Department of Bigdata Engineering, Soonchunhyang University

Received 21 April 2019, revised 16 May 2019, accepted 16 May 2019

Abstract

The stock market is a representative market representing the capitalist economic system and performs important economic functions in the financial market. The stock market also plays a role of acquiring capital for both individual and corporate investors. Accurately predicting the stock prices remains an important research task. In recent years, studies on stock price prediction using machine learning have progressed. In this study, we will use the XGBoost (extreme gradient boosting) model, which has been used in various fields recently and proved its excellence to predict stock price fluctuation. In order to demonstrate the superiority of the XGBoost model, we compared and analyzed the results of the LSTM (long-short term memory) neural network which showed good performance in the previous studies and the autoregressive model which is a conventional time series analysis technique. The empirical analysis shows that the XGBoost model is competitive in predicting stock price movements.

Keywords: Autoregressive model, LSTM neural network, machine learning, Stock price prediction, XGBoost model.

[†] This work was supported by the Yonsei University Wonju Campus Future-Leading Research Initiative of 2017 (2017-52-0072).
[1] Graduate student, Department of Information & Statistics, Yonsei University, Wonju 26493, Korea.
[2] Assistant professor, Department of Bigdata Engineering, Soonchunhyang University, Shinchang-myeon, Asan-si, Chungcheongnam-do 31538, Korea.
[3] Corresponding author: Associate professor, Department of Information & Statistics, Yonsei University, Wonju 26493, Korea. E-mail: ahn2615@yonsei.ac.kr