



회귀 강화학습을 이용한 포트폴리오 자산 배분의 성능 향상 연구

A Study on Performance Improvement of Portfolio Asset Allocation Using Recurrent Reinforcement Learning

강문주 · 안준규 · 이주홍[†]

Moon-Ju Kang, Jungyu Ahn and Ju-Hong Lee[†]

인하대학교 컴퓨터공학부

Department of Computer Engineering, Inha University

요약

최근, 자산 매매 및 포트폴리오 운용 분야에 인공지능을 활용한 사례 및 연구들이 활발히 진행되고 있다. 본 논문에서는 기존에 제시된 회귀 강화학습을 기반으로 한 포트폴리오 운용 모델의 성능 향상을 위해 포트폴리오를 구성하는 자산들의 예측값과 인공적으로 생성한 데이터를 사용하는 방법을 연구한다. 기존의 회귀 강화학습과 예측값을 함께 적용한 경우, 인공 생성 데이터를 함께 적용한 경우가 회귀 강화학습만을 사용하였을 때의 성능과 비교 분석하여 성능 향상에 도움이 됨을 보이고자 한다. 실험 결과, 회귀 강화학습에 과거 자산의 데이터만 사용한 경우보다 자산들의 예측값과 인공 생성 데이터를 함께 사용하였을 때 포트폴리오의 성능이 향상되었음을 알 수 있었다.

키워드 : 회귀 강화학습, 포트폴리오, 자산 예측, 인공 데이터

Abstract

Recently, researches using artificial intelligence in asset trading and portfolio have been actively conducted. In this paper, we study how to use the predicted values of the assets and artificially generated data to improve the performance of the portfolio management model based on the existing recurrent reinforcement learning(RRL). In the case of applying the RRL and the predicted value, it is shown that the artificial generation data is comparatively analyzed to improve the performance. Experimental results show that the performance of the portfolio is improved when the predicted value of assets and the artificial generation data are used together rather than the case of using the data of past assets only for RRL.

Key Words : Recurrent Reinforcement Learning, Portfolio, Asset prediction, Artificial data

Received: May. 3, 2018

Revised: Aug. 5, 2018

Accepted: Aug. 8, 2018

[†]Corresponding authors

juhong@inha.ac.kr

1. 서론

최근, 인공지능의 기술적 발전을 통해 뛰어난 성과를 거두어 다양한 분야에 적용되고 있다. 특히, 금융 분야에서 인공지능을 적용한 산업이 빠르게 발전하고 있다. 인공지능이 학습한 알고리즘을 이용해 투자 조언, 투자 결정 및 자산 운용을 할 수 있게 되었다. 인공지능이 적용되는 금융 분야의 세부적인 영역으로는 포트폴리오 최적화, 신용 등급평가, 주식 투자, 자산 예측 등이 있다. 그중 포트폴리오는 투자의 안정성 확보와 수익 창출이라는 목표를 위해 중요한 의사결정이 필요하다[14].

기존의 포트폴리오 알고리즘으로는 Markowitz[1]의 Mean-Variance 모델, 선형계획법, 비선형계획법 등이 있고, 인공지능을 활용한 방법으로는 인공신경망, 강화학습 등의 방법들이 있다[2]. 이들 방법 중에서 회귀강화학습 방법이 최근에 많은 관심을 받고 활발히 연구되고 있다[3,4,5,6,7]. 그러나 이들 연구들은 오직 자산들의 과거 데이터만을 사용하였을 뿐 포트폴리오의 성능 향상에 도움을 줄 수 있는 다른 요소들에 대한 적용이 부족하다. 본 논문에서는 회귀강화학습을 사용한 포트폴리오의 성능을 향상시키기 위해서 과거 자산 데이터 외에 자산 예측과 인공생성 데이터를 사용하는 방법을 제시한다.

이 논문은 2017년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(2017R1D1A1B03028929).

This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

자산 예측은 포트폴리오 성능 향상에 도움을 줄 가능성이 높다. 그에 대한 근거 논문은 다음과 같다. Jain[8]는 자산 예측은 이익을 얻는 중요한 역할을 한다고 주장하였다. Mohapatra[9]는 시장의 변동성으로 인해 예측의 경과가 100% 정확하지 않더라도 투자에 도움이 될 수 있다고 주장했다. 수익률이 높고 안정적인 종목들로 포트폴리오를 구성하는 것과 함께 좋은 예측력의 자산 예측 모델을 이용하여 매매하면 포트폴리오를 통해 추구하는 목표 달성에 근접할 수 있다[10,11].

또한 인공적으로 생성된 데이터를 사용하는 것이 도움이 될 수 있다. 실제 관측된 데이터들만을 사용한 훈련은 포트폴리오의 목표 달성을 이루기에는 부족할 수 있다. 주식 데이터가 초 단위, 일 단위로 많은 데이터 포인트들을 가지고 있지만, 데이터 포인트들의 집합은 단지 하나의 트렌드만을 나타내기 때문이다. 훈련 데이터의 부족은 견고한 모델 구축에 문제가 될 수 있다. 따라서 실제 데이터와 유사한 트렌드를 가지면서도 다양한 변동성을 지닌 인공 데이터를 생성해 학습에 함께 사용한다면 견고한 포트폴리오 모델 학습에 도움을 줄 수 있을 것이다.

본 논문은 포트폴리오를 위한 회귀 강화학습 모델의 성능 향상을 위해, 자산의 예측값과 인공 생성 데이터의 활용이 효과적일 것이라는 가설을 세웠고 실험을 통해 입증하려 한다. 회귀 강화학습 모델은 LSTM으로 구현해 사용했다. 실제 실험시의 자산의 예측값은 운용 기간 동안의 예측 정확도에 따른 자산 가격의 상승, 하락에 대한 가상 예측값을 생성해 사용했다. 인공 생성 데이터는 가우시안 프로세스를 사용했다.

본 논문의 구성은 다음과 같다. 2장은 본 논문과 관련된 연구들을 소개한다. 3장은 본 논문이 제안하는 모델이 서술되어 있다. 4장은 본 논문에서 진행한 실험이 서술되어 있다. 5장은 본 논문에 대한 결론이 서술되어 있다.

2. 관련 연구

Markowitz[1]는 포트폴리오를 최적화하는 평균-분산(Mean-Variance)모델을 소개하면서 포트폴리오 이론을 체계화하였다. 마코위츠 모델은 모든 투자기회 중에서 최적의 수익률, 리스크 조합을 가진 투자기회를 결정하는 이론으로, 각 종목의 종목 간의 과거 자료, 평균 수익률, 분산만을 사용해 분산 투자 하는 이론이다. 위험의 정도인 종목 간의 분산을 최소화, 최소 기대수익률 달성, 모든 사용 가능한 금액을 투자하는 제약조건 세 가지를 가진 비선형계획 모델이다[1,11,13].

Moody[3]는 회귀 강화학습을 사용하여 포트폴리오의 자산 할당 및 거래 시스템을 최적화하는 방법을 제시했다. 또한, Moody와

Saffell[4]은 회귀 강화학습과 Q-Learning을 실제 데이터를 사용하여 비교 실험을 하였고 Q-Learning보다 회귀 강화학습이 더 좋은 결과를 보였다고 소개하였다.

Yue Deng[5]이 제시한 모델은 딥러닝과 강화 학습의 두 가지 학습 개념에서 영감을 얻어 구현되었다. 제시된 모델에서 딥러닝 부분은 유익한 기능 학습을 위한 역동적인 시장 상태를 자동으로 감지한다. 그런 다음 강화학습 부분은 딥러닝을 통해 추출된 정보와 상호 작용하고 알려지지 않은 환경에서 최종 보상을 축적하기 위해 거래 의사 결정을 내린다. 학습 시스템은 심층 구조와 반복 구조를 모두 나타내는 복잡한 신경망으로 구현했다.

Saud Almahdi[6]는 매매 시그널과 자산 배분 비중을 얻기 위해 칼마 지수(Calmar Ratio)를 이용한 회귀 강화학습 모델을 제안했다. 실험은 자주 거래되는 상장 된 펀드로 구성된 포트폴리오를 사용하여, Expected Maximum Drawdown 기반의 목적 함수인 칼마 지수가 이전에 제안된 회귀 강화학습의 목적 함수와 비교하여 좀더 우수한 성능을 산출한 결과를 발표하였다.

Lu, Daivid W[7]는 회귀 강화학습을 LSTM으로 구현한 모델을 제시하였다. LSTM은 기존에 제시된 RNN 보다 일반적으로 좋은 성능을 보이며 회귀 강화학습의 훈련방식을 BPIT 학습방법으로 사용하여 만족할 수 있음을 보였다.

Kraus[15]는 반복적 신경망의 한 종류인 RNN은 기울기 값이 사라지는 문제와 문맥 종속성이 짧음 때문에 실제 문제에 적용이 어렵지만, LSTM은 입력 신호들 사이의 매우 긴 의존성을 갖지며 기울기 값이 사라지는 문제를 방지하는 포켓 게이트가 있어 RNN 이 가지는 문제를 해결할 수 있어 금융 결정 지원에 관한 연구에서 적합하다고 주장했다.

3. 제안 모델

3.1 자산 예측값 데이터 생성

예측 정확도에 따른 자산 예측 정보가 회귀 강화학습에 적용시 효과적인지를 실험하기 위해서 일정한 예측 정확도에 따른 자산 예측 데이터를 인공적으로 생성하였다. 예측 데이터는 운용 기간

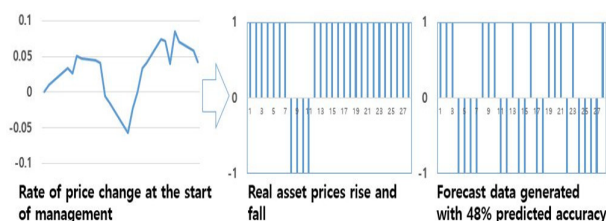


그림 1. 48% 예측 정확도 데이터 생성 예시

Fig. 1. Example of 48% prediction accuracy data generation

동안에 운용 시작 시점 기준으로 상승 예측은 1, 하락 예측은 -1로 표현하였다. 예측 정확도는 38%부터 64%까지 2% 단위로 나누어 적용하였다. 생성된 예측 데이터는 회귀 강화학습의 입력에 추가했다. 그림 1은 48% 예측 정확도 데이터 생성 예시이다.

3.2 가우시안 프로세스를 이용한 데이터 생성

가우시안 프로세스는 각 시간에서 관측된 자산의 가격으로 이루어진 훈련 데이터와 공분산 함수 커널을 사용하여 함수에 대한 하나의 확률 분포를 정의한다. 본 논문에서는 제곱 지수커널을 사용하였고, 변동성을 부여하기 위하여 노이즈 모델을 사용하였다. 커널의 수식은 수식(1)과 같다.

$$k(x_p, x_q) = \sigma_f^2 \exp(-\frac{1}{2l^2} |x_p - x_q|^2) + \sigma_n^2 \delta_{pq} \quad (1)$$

σ_n^2 은 훈련 데이터 집합에 대한 불확실성과 관련된 매개변수이다. σ_n^2 값을 조정해 원본 데이터의 트렌드는 따르지만 각 시간별로 차이가 있는 인공 생성 데이터를 생성했다. 그림 2는 인공 생성 데이터의 생성 과정 예시이다.

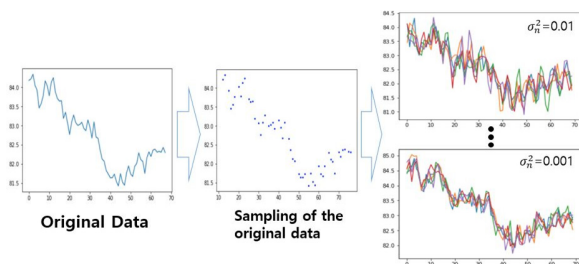


그림 2 인공 생성 데이터 생성 과정

Fig. 2. Artificial generated data creation process

3.3 회귀 강화학습 모델 설명

포트폴리오를 위한 회귀 강화학습의 목적은 환경(주식시장)과 상호작용을 통해 목적 함수(샤프지수)를 최대화되도록 행동(포트폴리오 자산 배분 비중)을 하는 모델을 학습시키는 것이다. 회귀 강화학습의 주요 특징으로는 이전 자산 배분 비중에 관한 정보들을 전달받아 현시점의 입력과 상호작용하여 자산 배분 비중을 출력한다. 또한, 모델이 입력으로 받는 상태와 최대화하려는 목적 함수의 설정이 자유로워 다양하게 정의 할 수 있다.

본 논문에서는 회귀 강화학습의 모델을 Unfold된 LSTM을 사용하여 구현하였다. LSTM은 회귀 강화학습과 같이 이전 포트폴리오 운용에 대한 정보를 LSTM의 Hidden State와 Cell State를 통해서 이전의 정보를 전달받아 현재 시점에서 입력으로 받은 상태와 상호작용하여 행동을 결정한다.

회귀 강화학습 모델의 훈련은 다음과 같은 순서로 진행된다.

Unfold된 LSTM은 각 시점에 대응되고, t 시점의 LSTM으로부터 자산 배분 비중 $\overrightarrow{F_t}(\theta)$ 를 얻는다. $\overrightarrow{F_t}(\theta)$ 를 통해 포트폴리오 수익률 R_t 를 얻는다. T 시점까지의 R_t 로 목적함수 $U_T(R_1, R_2, \dots, R_T)$ 를 구한다. 최종적으로 U_T 를 최대화하도록 LSTM의 내부 가중치 θ 를 조정한다. 수식 (2)는 U_T 를 최대화하는 미분 수식이다. 수식(2)를 정확히 계산하고 최적화하기 위해서 LSTM의 BPTT(Backpropagation Through Time)학습방법을 사용했다[12].

$$\frac{dU_T(\theta)}{d\theta} = \sum_{t=1}^T \frac{dU_T}{dR_t} \left\{ \frac{dR_t}{dF_t} \frac{dF_t}{d\theta} + \frac{dR_t}{dF_{t-1}} \frac{dF_{t-1}}{d\theta} \right\} \quad (2)$$

본 논문에서 U_T 는 대표적인 포트폴리오 성능 척도인 샤프 지수를 사용했다. 그림 3은 LSTM으로 구현한 회귀 강화학습 모델 구조이다.

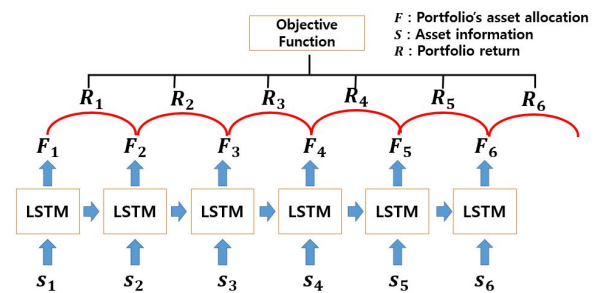


그림 3. LSTM으로 구현한 회귀 강화학습 모델 구조

Fig. 3. RRL model structure implemented by LSTM

3.4 상태(State), 행동(Action), 보상(Reward) 정의

회귀 강화학습을 이용한 포트폴리오 운용 모델 혹은 에이전트 (Agent)는 상태, 행동, 보상에 대한 정의가 필요하고, 아래와 같이 정의하였다.

기호1) 에이전트의 액션 $\overrightarrow{F_t}$

$\overrightarrow{F_t}$: 시간 t 시점에서 포트폴리오의 m 개의 개별 자산들에 대한 자산 배분 비중 벡터.

$$\overrightarrow{F_t} = (F_t^1, F_t^2, \dots, F_t^m) \quad (3)$$

F_t^i : 시간 t 시점에서 포트폴리오의 i 번째 개별자산에 대한 자산 배분 비중.

$$F_t^i \geq 0, \sum_{i=1}^m F_t^i = 1 \quad (4)$$

기호2) 에이전트의 상태 $\overrightarrow{S_t}$ 의

$\overrightarrow{S_t}$: 시간 t 시점에서 포트폴리오 운용 에이전트의 상태 입력 벡터.

$$\vec{S}_t = (\vec{a}_t^1, \vec{a}_t^2, \dots, \vec{a}_t^m) \quad (5)$$

\vec{a}_t^i, \vec{a}_t^i 는 예측값 사용 유무에 따라 두 가지 경우로 나뉜다.

-예측값 사용: 시간 t 시점에서 i 번째 개별자산의 과거 l 일 동안의 자산가격 일 단위 수익률 벡터와 미래 n 일 동안의 예측값 벡터,

$$\vec{a}_t^i = (\vec{r}_t^i, \vec{j}_{t:n}^i) \quad (6)$$

-예측값 미사용: 시간 t 시점에서 i 번째 개별자산의 과거 l 일 동안의 자산가격 일 단위 수익률 벡터,

$$\vec{a}_t^i = (\vec{r}_t^i) \quad (7)$$

$\vec{r}_{t:l}^i$: 시간 t 시점에서 과거 l 일 동안의 과거 일 단위 수익률 벡터,

$$\vec{r}_{t:l}^i = (r_{t-l+1}^i, r_{t-l+2}^i, \dots, r_t^i) \quad (8)$$

r_t^i : 시간 t 시점에서 i 번째 개별자산의 일 단위 수익률,

$$r_t^i = \frac{z_t^i}{z_{t-1}^i} - 1 \quad (9)$$

z_t^i : 시간 t 시점에서 i 번째 개별자산의 자산가격

$\vec{j}_{t:n}^i$: i 번째 개별자산의 시간 t 시점에서 미래 $t+n$ 구간의 예측 정확도에 따른 예측값 벡터, 각 예측값들은 상승 예측 시 1, 하락 예측 시 -1로 설정,

$$\vec{j}_{t:n}^i = (j_{t+1}^i, j_{t+2}^i, \dots, j_{t+n}^i) \quad (10)$$

기호3) 에이전트의 보상(Reward)과 목적함수 $U(\theta)$ 정의

$U(\theta)$: 총 운용 기간 T 시간 동안 에이전트의 행동으로 인해 발생한 포트폴리오의 수익률에 대한 샤프 지수,

$$U(\theta) = \frac{E(R_1, R_2, \dots, R_T)}{\sqrt{\text{Var}(R_1, R_2, \dots, R_T)}} \quad (11)$$

R_t : 시간 t 시점에서 포트폴리오의 수익률,

$$R_t = \sum_{i=1}^m W_t^i \cdot r_{t+1}^i \quad (12)$$

W_t^i : 시간 t 시점에서 자산의 가격변화에 따라 변경된 자산 배분 비중

4. 실험

4.1 실험 데이터

본 논문은 지수데이터 7개, 국내 주식 데이터 8개, 해외주식 데이터 10개, 총 25개의 데이터를 사용했다. 5개의 자산들의 집합을 설정해 5개의 포트폴리오를 구성하였고 모든 실험은 5개의 포트폴리오의 성능의 평균을 이용해 비교하였다. 5개의 포트폴리오는 표 1과 같이 구성되어있다. 데이터는 모두 일 단위 데이터를 사용했다. 훈련 데이터의 기간은 2012년 10월 17일부터 2014년 1월 3일까지로 설정했고, 테스트 데이터의 기간은 2014년 1월 6일부터 2015년 3월 26일까지로 설정하였다. 운용 일자 는 20일로 고정하였다. 표 2는 각 자산들의 테스트 기간 샤프 지수이고, 표 3은 5개의 포트폴리오의 테스트 기간 평균 샤프 지수이다.

표 1. 5개의 포트폴리오 집합

Table 1. Five Portfolio Set

Portfolio set 1	S&P500	KOSPI200	Samsung	AVA	BMJ
Portfolio set 2	NASDAQ	GOLD	Sk hynix	WTI	LG
Portfolio set 3	US High Yield	D/W	Posco	CHK	ETFC
Portfolio set 4	MS	NAVER	FCX	CPB	DGX
Portfolio set 5	CJCGV	CPK	APA	KIA	JYP

표 2. 포트폴리오 구성 자산들의 평균 샤프지수

Table 2. Average Sharpe ratio of Portfolio's Assets

Portfolio set	Average Sharpe ratio
Portfolio set 1	0.049152
Portfolio set 2	0.023568
Portfolio set 3	0.022871
Portfolio set 4	0.038378
Portfolio set 5	0.005824

표 3. 각 자산의 샤프 지수

Table 3. Sharp ratio of each asset

Asset Name	Asset's Sharpe Ratio	Asset Name	Asset's Sharpe Ratio
S&P500	0.074705	LG	0.079383
NASDAQ	0.078354	POSCO	-0.01419
KOSPI200	0.000641	SAMSUNG	0.031009
US High Yield	0.057435	SK Hynix	0.040773
GOLD	-0.059	NAVER	0.022871
WTI	-0.02167	KIA	0.005339
Dollar/Won	0.020966	JYP	-0.05148
CHK	-0.00364	CJ CGV	0.018301
FCX	-0.08891	BMJ	0.057119
ETFC	0.05378	AVA	0.082289
DGX	0.112974	MS	0.063043
CPK	0.084346	APA	-0.02739
CPB	0.081915		

4.2 실험 환경

실험 환경은 Intel Xeon 3.50Ghz CPU, 128G DRAM과 NVIDIA GTX 1080을 사용하여 진행했다. 실험 프로그램은 Python과 Tensorflow를 사용했다.

4.3 최적의 Unfold수 실험

LSTM으로 구현한 회귀 강화학습의 최적의 Unfold수를 찾는 실험을 수행했다. Unfold만을 변수로 두고, 상태의 길이는 현시점부터 과거 60일로 고정해 실험했다. Unfold수는 3,5,8,10,12로 실험했다. 실험결과 Unfold 수가 3에서 가장 높은 성능을 보였으며, 결과는 그림 4와 같다.

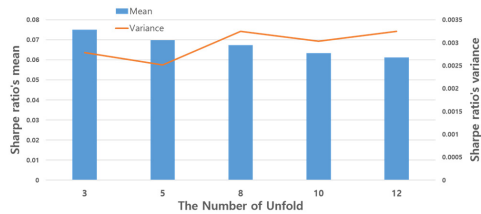


그림 4. Unfold 수에 따른 실험 결과

Fig. 4. Experimental results according to Unfold number

4.4 최적의 상태의 길이 실험

4.3을 통해 Unfold수를 3으로 고정한 다음, 최적의 상태 길이를 찾기 위해 현 시점에서 과거 20일, 40일, 60일, 80일, 100일, 120일로 설정하여 실험했다. 실험결과 20일에서 가장 좋은 성능을 보였다. 결과는 그림 5와 같다.

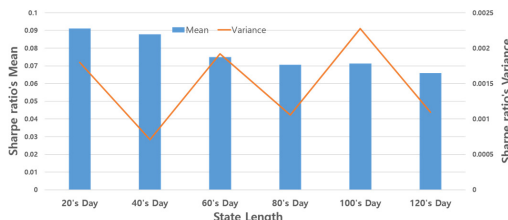


그림 5. 상태 길이에 따른 실험 결과

Fig. 5. Experimental results according to length of State

4.5 회귀 강화학습과 다른 알고리즘의 성능 비교실험

회귀 강화학습의 기본적인 성능을 분석하기 위하여, 4.3과 4.4에서 얻은 최적의 Unfold수와 상태의 길이를 적용하여 Markowitz 모델과 1/N 포트폴리오와 비교실험 하였다. 실험결과 회귀 강화학습이 가장 좋은 성능을 보였다. 결과는 그림 6과 같다.

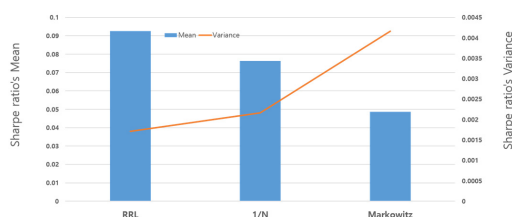


그림 6. 다른 알고리즘과의 성능 비교 실험

Fig. 6. Performance comparison experiment with other algorithms

4.6 예측값 적용 실험

4.6 에션 예측값을 적용한 회귀 강화학습 모델의 성능을 실험했다. Unfold수는 3, 상태의 길이는 20일, 예측 정확도는 38%~64%로 2%단위로 나누어 실험했다. 그림7에서는 과거 자산데이터만을 사용한 회귀 강화학습의 샤프지수 대비 증감률을 그래프로 표현하였다. 실험결과 예측 정확도가 높을 때는 물론이고, 낮은 예측 정확도에서도 유의미한 상승을 보였다.

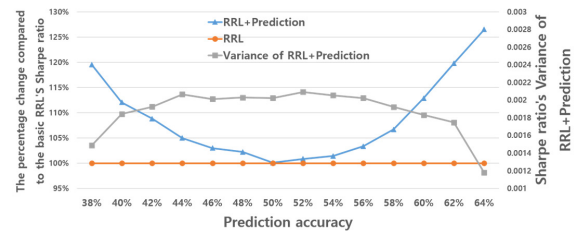


그림 7. 예측값 적용에 대한 실험

Fig. 7. Experiments on the application of predictive values

4.7 인공 생성 데이터 적용 실험

가우시안 프로세스의 파라미터 σ_n^2 을 여러 단계로 나누어 원본 데이터와 함께 학습시켰다. 원본 데이터와 인공 생성 데이터의 비율은 1:1이고, 인공 생성 데이터를 적용한 모든 경우에서 성능향상을 보였다. 실험결과는 과거 자산 데이터만을 이용한 회귀 강화학습의 샤프지수 대비 증감률을 그림8에서 그래프로 표현하였다.

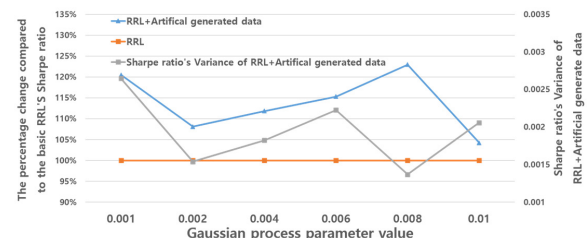


그림 8. 인공 생성 데이터 적용에 대한 실험

Fig. 8. Experiments on the application of artificial generated data

4.8 인공 생성 데이터 조합의 성능 실험

표 4. 인공 생성 데이터 조합에 따른 성능 표

Table 4. Performance Table according to Artificial generated data combination

Gaussian process parameter value 1	0.01	120% / 0.0029	121% / 0.0008	121% / 0.0018	115% / 0.0026	111% / 0.0019	
	0.008	119% / 0.0011	133% / 0.0016	125% / 0.0012	116% / 0.0025		
	0.006	127% / 0.0016	122% / 0.0029	110% / 0.0017			
	0.004	120% / 0.0027	116% / 0.0023				
	0.002	112% / 0.0008					
	0.001						
	σ_n^2	0.001	0.002	0.004	0.006	0.008	0.01
Gaussian process parameter value 2							

4.7에서는 원본 데이터와 인공 생성 데이터의 비율이 1:1이었지만, 4.9에서는 원본 데이터와 2개의 다른 파라미터의 인공 생성 데이터의 조합으로 실험을 진행했다. 각각의 비율을 1:1로 설정하여 진행했고 실험결과는 표4에서 과거 자산 데이터만을 이용한 회귀 강화학습의 샤프지수 대비 증감률과 분산을 나타낸다.

4.9 인공 생성 데이터 비율 증가에 따른 성능 실험

4.7과 4.8에서 최고의 성능을 보인 경우에 대해서 원본 데이터와 인공 생성 데이터의 비율을 조정한 실험을 진행했다. 최고의 성능을 보인 경우는 원본 데이터, 첫 번째 $\sigma_n^2=0.002$ 두 번째 $\sigma_n^2=0.008$ 일 때이다. 이 경우에 대해 인공 생성 데이터의 비율 증가가 성능 향상에 어떠한 영향을 미치는지 실험하였다. 표5는 인공 생성 데이터의 비율을 나타내고 있고 실험결과는 그림9와 같다. 그림9는 과거 자산 데이터만을 이용한 회귀 강화학습의 샤프지수 대비 증감률을 나타내었다.

표 5. 인공 생성 데이터 비율 표

Table 5. Artificial generated data ratio table

	Set1	Set 2	Set 3	Set 4	Set 5	Set 6	Set 7	Set 8	Set 9	Set10
Original data	1	1	1	1	1	1	1	1	1	1
$\sigma_n^2 = 0.002$	1	2	3	4	5	6	7	8	9	10
$\sigma_n^2 = 0.008$	1	2	3	4	5	6	7	8	9	10

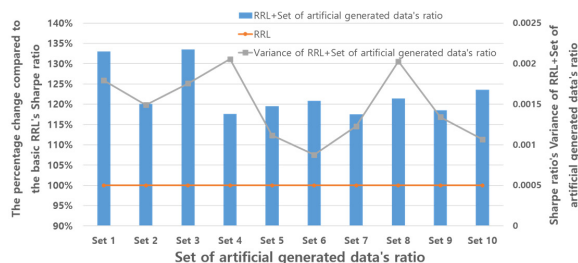


그림 9. 인공 생성 데이터 비율에 따른 실험결과

Fig. 9. Experimental results according to artificial generated data ratio

5. 결론

본 논문은 포트폴리오를 위한 회귀 강화학습의 성능을 향상시키기 위해 예측값과 인공 생성데이터 적용을 제안하였다. 실제 예측값은 주어진 정확도에 따라 운용 기간 동안의 상승과 하락을 인공적으로 생성해 사용하였고, 인공 생성 데이터는 가우시안 프로세스를 이용하여 사용하였다. 실험결과를 통해 예측값과 인공 생성 데이터의 사용이 성능 향상에 크게 도움이 되었고 최대 약 34%의 성능 향상을 확인하였다.

향후 연구에서는 포트폴리오 성능향상을 위해 회귀 강화학습의

입력과 목적함수를 다양하게 변경한 실험과 실제 자산을 예측하는 모델과 인공 생성 데이터를 함께 적용해 볼 계획이다.

References

- [1] Markowitz, H., "Portfolio selection," *Journal of Finance* 7, 77-91,(1992).
- [2] SeungKyu Hwang, HyungJoon Lim, ShiYong Yoo, "Simulation on the Optimal Asset Allocation with Expected Returns Estimates," *KAPP* 11,1: 27-57,(2009).
- [3] Moody J et al, "Performance function and reinforcement Learning for trading systems and portfolios," *Journal of Forecasting*,(1997).
- [4] Moody, J., & Saffell, M., "Learning to trade via direct reinforcement," *IEEE Transaction on Neural Networks*,(2001).
- [5] Deng, Y., Bao, F., Kong, Y., Ren, Z., and Dai, Q., "Deep direct reinforcement learning for financial signal representation and trading," *IEEE Transactions on Neural Neural Networks and Learning Systems*,(2016).
- [6] Almahdi, S., & Yang, S. Y. "An adaptive portfolio trading system: A risk-return portfolio optimization using recurrent reinforcement learning with expected maximum drawdown," *Expert Systems with Applications*, 87, 267-279,(2017).
- [7] Lu, David W. "Agent Inspired Trading Using Recurrent Reinforcement Learning and LSTM Neural Networks," *arXiv preprint arXiv:1707.07338*,(2017).
- [8] Jain, Vikalp Ravi, Manisha Gupta, and Raj Mohan Singh, "Analysis and Prediction of Individual Stock Prices of Financial Sector Companies in NIFTY50," *International Journal of Information Engineering and Electronic Business* 10,2: 33,(2018).
- [9] Mohapatra, Avilasa, et al, "Applications of neural network based methods on stock market prediction: survey," *International Journal of Engineering and Technology(UAE)* 7,26: 71-76,(2018).
- [10] Kanghee Park, Hyunjung Shin, "Stock Trading Model using Portfolio Optimization and Forecasting Stock Price Movement," *KIIE* 39,6: 535-545,(2013).
- [11] Guresen, Erkam, Gulgun Kayakutlu, and Tugrul U. Daim, "Using artificial neural network models in stock market index prediction," *Expert Systems with Applications* 38,8: 10389-10397,(2011).
- [12] Moody, J., Wu, L., Liao, Y., & Saffell, M., "Performance functions and reinforcement learning for trading systems and portfolios," *Science*, 17, 441-470,(1988).

- [13] Jooyoung, Park., Jinho, Jeong., Kyungwook, Park., "An Investigation on Dynamic Portfolio Selection Problems Utilizing Stochastic Receding Horizon Approach.", *Journal of The Korean Institute of Intelligent Systems*, Vol. 22, No. 3, pp.386-393, (2012).
- [14] Sungcheol, Park., Sungwoong, Kim., Heungsik, Choi., "Selection Model of System Trading Strategies using SVM.", *Journal of The Korean Institute of Intelligent Systems*, Vol. 20, No. 2, pp.59-71, (2014).
- [15] Kraus, Mathias, and Stefan Feuerriegel. "Decision support from financial disclosures with deep neural networks and transfer learning." *Decision Support Systems* 104 : 38-48, (2017).

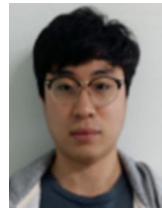
저자 소개



강문주(Moon-Ju Kang)

2016년 : 계명대학교 의용공학과 졸업(학사)
2018년 : 인하대학교 컴퓨터공학과 졸업(석사)

관심분야 : 강화학습, 딥 러닝, 금융공학
Phone : +83-32-860-7453
E-mail : sktel1020@nate.com



안준규(Jungyu Ahn)

2014년 : 인하대학교 컴퓨터공학 졸업(학사)
2018년 : 인하대학교 컴퓨터공학 졸업(석사)
2018년~현재 : 인하대학교 컴퓨터공학과
박사과정

관심분야 : 시계열분석, 데이터마이닝
Phone : +83-32-860-7453
E-mail : ahnjunguy320@gmail.com



이주홍(Ju-Hong Lee)

1983년 : 서울대학교 전자계산기공학 졸업(학사)
2013년 : 서울대학교 컴퓨터공학과 졸업(석사)
2001년 : 한국과학기술원 컴퓨터공학 졸업(박사)
2001년~현재 : 인하대학교 공과대학
컴퓨터공학과 교수

관심분야 : 머신러닝, 데이터마이닝, 시계열분석
Phone : +83-32-860-7453
E-mail : juhong@inha.ac.kr