University of Qom
Engineering Management and Soft Computing
JEMSC.QOM
Online ISSN: 2538-2675

Portfolio Optimization using Deep Reinforcement Learning

Somayeh Namdari-Birgani[1], Amir Hossein Seddighi[2] and Saber Molla-Alizadeh-Zavardehi[3]

1. Ph.D. Student, Department of management, Masjed Soleyman Branch, Islamic Azad University, Masjed Soleyman, Iran. Email: Somayehnamdari90@gmail.com
2. Corresponding Author, Assistant Prof., Department of management, Masjed Soleiman Branch, Islamic Azad University, Masjed Soleiman, Iran; Information Technology Research Department, Iranian Research Institute for Information Science and Technology (IranDoc), Tehran, Iran. Email: Seddighi@irandoc.ac.ir
3. Assistant Prof., Department of Industrial Engineering, Masjed Soleiman Branch, Islamic Azad University, Masjed Soleyman, Iran. Email: Saber.alizadeh@gmail.com

Article Info

Article type:
Research Article

Article history:
Received 11 8 2024
Received in revised form 20 10 2024
Accepted 18 2 2025
Published online 18 3 2025

Keywords:
Portfolio Optimization,
Reinforcement Learning,
Artificial Intelligence,
Q Deep Reinforcement Learning,
Dow Jones Industrial Average.

ABSTRACT

This research aims to train an intelligent trader by using artificial intelligence concepts that can help to make optimal decisions for investing in the stock portfolio. For this purpose, a method based on Q deep reinforcement learning is presented for portfolio optimization. In this method, the policy network and the target policy network are used to learn the actions, and the learning network and the target network are used to estimate the optimal Q. The data related to the companies constituting the Dow Jones Industrial Average (DJIA) from March 2008 to October 2021 are used to evaluate the proposed method. Moreover, the performance of the proposed method is compared with conventional investment strategies and two deep reinforcement learning algorithms, Proximal Policy Optimization (PPO) and Soft Actor-Critic (SAC). The results indicate that the proposed method has the best performance on the test data with a total profit of 35.6% compared to other investigated methods. On the other hand, the Sharpe ratio of the proposed method is the highest value, which implies this strategy performs better in balancing profit and risk.

Cite this article: Namdari-Birgani, S. & Others, (2024)., Portfolio Optimization using Deep Reinforcement Learning. *Enginiiring Management and Soft Computing*, *10* (2). 1-22. **DOI**: https://doi.org/10.22091/jemsc.2025.11158.1192

© The Author(s)
Publisher: University of Qom
DOI: https://doi.org/10.22091/jemsc.2025.11158.1192

JEMSC.QOM
شاپا الکترونیکی: ۲۶۷۵-۲۵۳۸
مدیریت مهندسی و رایانش نرم
دانشگاه قم

بهینه‌سازی سبد سهام با استفاده از یادگیری تقویتی عمیق

سمیه نامداری بیرگانی[۱]، امیر حسین صدیقی[۲]✉ و صابر ملاعلیزاده زواردهی[۳]

۱. دانشجوی دکتری مدیریت صنعتی گرایش مالی، گروه مدیریت، واحد مسجدسلیمان، دانشگاه آزاد اسلامی، مسجدسلیمان، ایران . Somayehnamdari90@gmail.com

۲. نویسنده مسئول، استادیار، گروه مدیریت، واحد مسجدسلیمان، دانشگاه آزاد اسلامی، مسجدسلیمان، ایران؛ پژوهشکده فناوری اطلاعات، پژوهشگاه علوم و فناوری اطلاعات ایران (ایرانداک)، تهران، ایران. Seddighi@irandoc.ac.ir

۳. استادیار، گروه صنایع، واحد مسجدسلیمان، دانشگاه آزاد اسلامی، مسجدسلیمان، ایران. Saber.alizadeh@gmail.com

اطلاعات مقاله

نوع مقاله: مقاله پژوهشی

تاریخ دریافت: ۱۴۰۳/۰۳/۱۹

تاریخ بازنگری: ۱۴۰۳/۰۶/۱۴

تاریخ پذیرش: ۱۴۰۳/۰۶/۱۵

تاریخ انتشار: ۱۴۰۳/۰۶/۳۱

کلیدواژه‌ها:
بهینه‌سازی سبد سهام،
یادگیری تقویتی،
هوش مصنوعی،
یادگیری تقویتی عمیق Q،
شاخص داو جونز

چکیده

پژوهش حاضر قصد دارد تا با استفاده از مفاهیم هوش مصنوعی، معامله‌گر هوشمندی را آموزش دهد که بتواند به تصمیم‌گیری بهینه برای سرمایه‌گذاری در سبد سهام کمک کند. بدین منظور روشی مبتنی بر یادگیری تقویتی عمیق Q برای بهینه‌سازی سبد سهام پیشنهاد خواهیم داد. در این روش از شبکه سیاست و شبکه سیاست هدف برای یادگیری اقدام‌ها و از شبکه یادگیری و شبکه هدف برای برآورد Q بهینه بهره گرفته می‌شود. برای ارزیابی عملکرد روش پیشنهادی از داده‌های مربوط به شرکت‌های تشکیل‌دهنده شاخص داو جونز (DJIA) از مارس ۲۰۰۸ تا اکتبر ۲۰۲۱ استفاده می‌گردد. بعلاوه عملکرد روش پیشنهادی با استراتژی‌های مرسوم سرمایه‌گذاری و دو الگوریتم یادگیری تقویتی عمیق، بهینه‌سازی سیاست پروکسیمال (PPO) و بازیگر-منتقد نرم (SAC) مقایسه می‌شود. نتایج این بررسی‌ها حاکی از آن است که روش پیشنهادی بر روی دادگان آزمون با مجموع بازده ۳۵.۶ درصدی در مقایسه با سایر روش‌های بررسی شده بهترین عملکرد را دارد. از سوی دیگر نسبت شارپ در روش پیشنهادی بیشترین مقدار است که نشانگر آن است که این استراتژی در متعادل‌سازی بین سود و ریسک عملکرد بهتری دارد.

استناد: نامداری بیرگانی، سمیه و همکاران. (۱۴۰۳). «بهینه‌سازی سبد سهام با استفاده از یادگیری تقویتی عمیق». *مدیریت مهندسی و رایانش نرم، دوره ۱۰* (۲). صص: ۲۲-۱. https://doi.org/10.22091/jemsc.2025.11158.1192

ناشر: دانشگاه قم

© نویسندگان.

۱) مقدمه

سرمایه‌گذاری و انباشت سرمایه در تحول اقتصادی کشورها به ویژه ایران نقش بسزایی دارد. اهمیت و نقش مؤثر این عامل را می‌توان به طور برجسته‌ای در کشورهایی با نظام سرمایه‌داری مشاهده نمود. بدون تردید بورس یکی از جایگاه‌های مناسب برای جذب سرمایه‌های کوچک و استفاده از آن‌ها در سطح کلان برای رشد شرکت‌ها است. افراد با سرمایه‌گذاری در بورس به دنبال رشد سرمایه شخصی و دستیابی به سود مورد انتظار خود هستند (Tolouie Eshlaghy & Haghdoust, 2007).

یکی از راه‌های سرمایه‌گذاری در بورس، پرتفوی یا سبد سهام است که شامل مجموعه‌ای از دارایی‌های غیرنقدی مانند سهام شرکت‌های مختلف می‌شود که در اختیار یک سرمایه‌گذار یا شرکت کارگزاری قرار دارد. یکی از مهم‌ترین موارد در مدیریت سبد سهام، تصمیم‌گیری بهینه درباره خرید یا فروش هر یک از سهام موجود در پرتفوی با توجه به پیش‌بینی سرمایه‌گذار از سوددهی یا زیان‌دهی هر سهم در طول دوره سرمایه‌گذاری است (Soleymani & Paquet, 2021).

معاملات با استفاده از سبد سهام را می‌توان یکی از مهم‌ترین روش‌های سرمایه‌گذاری در صنعت مالی دانست. با توجه به شرایط متغیر بازار مالی، معامله‌گران تلاش می‌کنند تا استراتژی‌های معاملاتی خود را به نحوی تنظیم کنند که بتوانند سرمایه خود را به دارایی‌های مالی مناسب اختصاص دهند. چالش اصلی در اینجا بهینه‌سازی سبد سهام به منظور حداکثرسازی ثروت پولی معامله‌گر با استفاده از اختصاص سرمایه به سبدی از دارایی‌ها در طول یک دوره زمانی است. برای حل این مساله استراتژی‌های مختلفی معرفی شده است که یکی از شناخته‌شده‌ترین آن‌ها نظریه سبد سهام مدرن[1] است که توسط مارکویتز پیشنهاد شد (Markowitz, 1991).

در سال‌های اخیر، استفاده از روش یادگیری تقویتی[2] برای حل مسائل در زمینه‌های مختلف از جمله در حوزه مالی بیشتر شده است (Kabbani & Duman, 2022; Alfonso-Sánchez et al. 2024; Vergara & Kristjanpoller, 2024). در روش‌های مبتنی بر یادگیری تقویتی، یک عامل یادگیرنده می‌تواند با انجام اقدامات مختلف تجاری و تجدید نظر در سیاست اقدام تجاری خود، یک محیط پیچیده مالی را درک کند و به دنبال آن استراتژی معاملاتی خود را بر اساس این تجربیات بهینه نماید. علاوه بر این، مزیت مهم این روش این است که عوامل یادگیرنده می‌توانند استراتژی‌های معاملاتی خود را بر اساس تجربیات خود در روزهای معاملاتی آینده به‌روز کنند. بدین ترتیب به جای حفظ استراتژی‌های معاملاتی به دست آمده از داده‌های تاریخی، عوامل یادگیرنده می‌توانند استراتژی‌های خود را با شرایط معاملاتی در هر روز تطبیق دهند (Wang et al., 2017). یادگیری Q یکی از رویکردهای یادگیری تقویتی برای یادگیری ارزش اتخاذ یک اقدام مشخص (برای مثال فروش یا خرید یک یا چند سهام) در یک حالت یا موقعیت خاص (اطلاعات اخیر درباره قیمت و معامله سهام‌ها) است. این روش به مدل‌بندی محیط بازار نیاز ندارد و نوعا بر اساس ساخت جدولی از مقادیر Q (کیفیت یک اقدام در یک حالت) بنا نهاده شده است (Li, 2023). یادگیری تقویتی عمیق از

[1] Modern portfolio theory
[2] Reinforcement Learning

شبکه‌های عمیق برای حل مسائل با استفاده از چارچوب یادگیری تقویتی بهره می‌گیرد. برای نمونه یادگیری عمیق Q به عنوان یکی از روش‌های مبتنی بر یادگیری تقویتی عمیق الگوریتمی است که درآن از شبکه عصبی برای تخمین مقادیر Q در روش یادگیری Q استفاده می‌شود (Varga et al., 2022). یادگیری تقویتی در عمل نتایج قابل توجهی را در مقایسه با سایر روش‌ها در مدیریت سبد سهام از خود نشان داده است (Ngo et al., 2023). بنابراین با توجه به آنچه گفته شد این مقاله تلاش دارد تا با استفاده از یادگیری تقویتی عمیق Q روشی برای بهینه‌سازی سبد سهام ارائه دهد. سپس کارایی این روش با استراتژی‌های مرسوم سرمایه‌گذاری مقایسه خواهد شد.

مهمترین نوآوری‌های این مقاله در دو بخش است. اول اینکه روش یادگیری تقویتی عمیق پیشنهادی برای حل مساله بهینه‌سازی سبد سهام سفارشی‌سازی شده است. به عبارت دیگر فضای حالت در این مقاله بر اساس ماتریس کوواریانس قیمت پایانی سهام‌ها تعریف شده است که بر اساس مطالعات عددی انجام شده نتایج قابل قبولی حاصل می‌نماید. دوم اینکه ساختار شبکه‌های عمیق پیشنهادی برای یادگیری مقادیر Q و اقدام‌ها نیز به نحوی صورت گرفته است که بیشترین بازده در بررسی‌های عددی حاصل شود.

ساختار مقاله در ادامه به این صورت است. ابتدا در بخش دوم به مروری بر مطالعات انجام شده خواهیم پرداخت. مساله اصلی پژوهش در بخش سوم بیان شده و با استفاده از فرایند تصمیم‌گیری مارکف مدل‌بندی می‌شود. بخش چهارم روش حل پیشنهادی را مبتنی بر یادگیری تقویتی عمیق Q تشریح می‌نماید. عملکرد روش پیشنهادی در مقایسه با استراتژی‌های مرسوم سرمایه‌گذاری و دو الگوریتم یادگیری تقویتی عمیق دیگر در بخش پنجم و با استفاده از یک مجموعه داده واقعی ارزیابی می‌شود. در انتها به جمع‌بندی مطالب پرداخته شده است.

۲) پیشینه پژوهش

یادگیری تقویتی ابزاری قدرتمند برای حل مسائل پیچیده تصمیم‌گیری، از جمله بهینه سازی سبد سهام است. روش‌های سنتی مانند بهینه‌سازی میانگین-واریانس و سایر انواع آن که ریشه در مطالعات مارکوویتز دارند (Markowitz, 1991) دارای محدودیت‌هایی نظیر حساسیت به تخمین پارامتر و فرض نرمال بودن توزیع بازده دارایی هستند. در مقابل یادگیری تقویتی یک رویکرد تطبیقی مبتنی بر داده را برای مدیریت سبد سهام ارائه می دهد که بر چنین مفروضاتی متکی نیست و نوعا به مدل‌سازی صریح پویایی‌های محیط بازار نیاز ندارد. بعلاوه نتایج تجربی حاکی از برتری روش یادگیری تقویتی نسبت به روش‌های سنتی در مدیریت سبد سهام است (Ngo et al., 2023). در ادامه مروری بر کاربرد یادگیری تقویتی در بهینه سازی سبد سهام خواهیم داشت. مطالعات در این بخش بر حسب نوع روش حل به سه دسته یادگیری تقویتی فقط منتقد[3]، یادگیری تقویتی فقط بازیگر[4]، و یادگیری تقویتی بازیگر-منتقد[5] تقسیم بندی می‌شوند.

[3] Critic-only
[4] Actor-only
[5] Actor-Critic

۲-۱) یادگیری تقویتی فقط منتقد

در این رویکرد عامل به دنبال یادگیری تابع ارزشی است که به کمک آن بتواند نتایج مورد انتظار ناشی از اقدام‌های مختلف را مقایسه یا نقد نماید. در هنگام تصمیم‌گیری، عامل حالت فعلی محیط را در نظر گرفته و اقدامی را انتخاب می‌نماید که بهترین نتیجه را بر اساس تابع ارزش به دست دهد (Zhang et al., 2020).

مهمترین الگوریتم در این دسته، شبکه عمیق Q[6] (DQN) است که در اصل برای یادگیری بازی‌های آتاری پیشنهاد شد (Mnih et al., 2013). چن و گائو (۲۰۱۹) سعی کردند تا از ایده مشابهی برای توسعه یک معامله‌گر استفاده کنند که محیط بازار را همچون یک بازی در نظر گرفته و سعی در بیشینه کردن سود خود دارد. در این مطالعه از ترکیب DQN با شبکه‌های عصبی بازگشتی استفاده شد. نویسندگان از داده‌های تاریخی S&P 500 برای آموزش و آزمون الگوریتم پیشنهادی خود استفاده کردند و عملکرد آن را با استراتژی‌های سنتی مقایسه نمودند. نتایج این پژوهش برتری الگوریتم پیشنهادی را با حدود ۲۳ درصد سود سالیانه نشان می‌دهد (Chen & Gao, 2019). با این حال این مطالعه تنها به معامله یک سهام می‌پردازد.

جین و الساوی (۲۰۱۶) از یادگیری عمیق Q به عنوان یک رویکرد فقط منتقد برای بهینه‌سازی سبد سهام استفاده کردند. در این راستا از شبکه عصبی مصنوعی برای تقریب تابع اقدام-ارزش بهره گرفته شد. بعلاوه از داده‌های مربوط به قیمت پایانی روزانه سهام از ۲۰ ژوئیه ۲۰۰۱ تا جولای ۲۰۱۶ استفاده کردند. نتایج این پژوهش نشان داد که شبکه عصبی مصنوعی پیشنهادی در برابر برخی از روش‌های موجود به طور قابل‌توجهی بهتر عمل می‌کند (Jin & El-Saawy, 2016).

۲-۲) یادگیری تقویتی فقط بازیگر

در این رویکرد عامل حالت محیط را دریافت کرده و به صورت مستقیم و بدون محاسبه و مقایسه نتایج مورد انتظار عمل می‌نماید. بنابراین عامل، نگاشت مستقیمی از حالت‌ها به اقدام‌ها را یاد می‌گیرد که به این نگاشت در اصطلاح سیاست گفته می‌شود.

جیانگ و همکاران (۲۰۱۷) چارچوبی مبتنی بر یادگیری تقویتی فقط بازیگر و بدون مدل مالی برای مدیریت سبد سهام ارائه دادند. این چارچوب شامل گروه ارزیابان مستقل یکسان، یک حافظه بردار پرتفو، یک طرح آموزش دسته‌ای تصادفی آنلاین و یک تابع پاداش صریح است. نویسندگان از سه شبکه عصبی پیچشی، شبکه عصبی بازگشتی و شبکه عصبی حافظه طولانی کوتاه مدت در این چارچوب استفاده کردند و از آن‌ها برای معاملات ۳۰ دقیقه‌ای در بازار رمز ارزها بهره بردند. هر سه شبکه در این چارچوب بهترین نتایج را در همه آزمایش‌ها به دست آوردند (Jiang et al., 2017).

یی و همکاران (۲۰۲۰) جهت مدیریت و بهینه‌سازی سبد سهام یک الگوریتم یادگیری تقویتی فقط بازیگر پیشنهاد دادند. این الگوریتم سعی در حل دو چالش اصلی یعنی ناهمگونی داده‌ها و عدم قطعیت در بازار مالی دارد. کارایی

[6] Deep Q Network

الگوریتم پیشنهادی در این پژوهش با استفاده از دو مجموعه داده واقعی مربوط به بازار بیت کوین و بازار سهام نشان داده شد (Ye et al., 2020).

۲–۳) یادگیری تقویتی بازیگر–منتقد

این رویکرد به دنبال ترکیب مزیت‌های دو رویکرد پیشین است. ایده اصلی در این روش استفاده همزمان از بازیگر برای مشخص کردن اقدام عامل در حالت فعلی محیط و منتقد برای قضاوت درباره اقدام انتخاب شده است. به عبارت دیگر بازیگر یاد می‌گیرد که اقدامی را انتخاب کند که از نظر منتقد بهترین است و منتقد یاد می‌گیرد که قضاوت خود را بهبود دهد.

سلیمانی و پاکوت (۲۰۲۱) جهت مدیریت سبد سهام از یک الگوریتم یادگیری تقویتی با استفاده از شبکه عصبی پیچشی گرافی استفاده کردند که هدف آن بهره‌گیری از همبستگی زمانی بین سهام‌های مختلف بود. الگوریتم پیشنهادی شامل یک خودرمزگذار برای استخراج ویژگی‌ها، یک شبکه عصبی پیچشی برای جمع‌آوری اطلاعات مربوط به سهام و روش یادگیری تقویتی بازیگر–منتقد بود. این الگوریتم با استفاده از چندین مجموعه داده واقعی مورد آزمایش قرار گرفت و موفق شد تا از شاخص‌های بازار بهتر عمل نماید (Soleymani & Paquet, 2021).

کابانی و دومان (۲۰۲۲) از فرایند تصمیم‌گیری مارکف قابل مشاهده به صورت جزئی[۷] (POMDP) برای مدل‌بندی مدیریت سبد سهام و از روش گرادیان سیاست قطعی عمیق دوقلو با تاخیر[۸] (TD3) برای حل این مدل استفاده کردند. نتایج این پژوهش نشان می‌دهد که استفاده از فضای اقدام پیوسته در مقایسه با گسسته خروجی بهتری در بر دارد. بعلاوه بکارگیری رویکرد بازیگر–منتقد در داده‌های واقعی عملکرد قابل توجهی از خود نشان داد (Kabbani & Duman, 2022).

۲–۴) مطالعات مقایسه‌ای

گائو و چان (۲۰۰۰) دو روش فقط منتقد و فقط بازیگر را در کنار هم برای مدیریت سبد سهام بهره گرفتند. آن‌ها از سود مطلق و سود نسبی تعدیل شده با ریسک به عنوان عملکرد برای آموزش این سیستم با به‌کارگیری مجموعه‌ای از دو شبکه استفاده کردند. نتایج این پژوهش نشان می‌دهد که ادغام[۹] اقدام‌های ناشی از دو روش فقط منتقد و فقط بازیگر برای اتخاذ تصمیم نهایی می‌تواند در بازارهای ارزی سود قابل‌توجهی ایجاد نماید (Gao & Chan, 2000).

دو و همکاران (۲۰۰۹) روش یادگیری Q را به عنوان یک رویکرد فقط منتقد با روش یادگیری تقویتی بازگشتی مقایسه می‌کنند که رویکردی فقط بازیگر است. هدف عامل در این پژوهش بیشینه‌سازی مجموع بازده با تخصیص سرمایه بین یک دارایی پر خطر و یک دارایی بدون خطر (پول نقد) در هر دوره زمانی است. نتایج این مطالعه برتری رویکرد فقط بازیگر را نسبت به رویکرد فقط منتقد نشان می‌دهد (Du et al., 2009). با این حال با بررسی مطالعات مشابه

[7] Partially Observable Markov Decision Process
[8] Twin Delayed Deep Deterministic Policy Gradient
[9] Ensemble

نمی‌توان به جمع‌بندی شفافی در مقایسه بین این دو رویکرد رسید. به عنوان نمونه دورسون و همکاران (۲۰۰۵) نتایج روش یادگیری Q یعنی رویکرد فقط منتقد را بهتر ارزیابی می‌کنند (Duerson et al., 2005).

لیانگ و همکاران (۲۰۱۸) برای مدیریت سبد سهام از سه الگوریتم یادگیری تقویتی یعنی گرادیان سیاست قطعی عمیق[10] (DDPG)، بهینه‌سازی سیاست پروکسیمال[11] (PPO) و گرادیان سیاست[12] (PG) استفاده کردند که الگوریتم‌های اول و دوم مبتنی بر رویکرد بازیگر-منتقد و الگوریتم سوم از رویکرد فقط بازیگر بهره می‌گیرد. نویسندگان آزمایش‌های مختلفی را با استفاده از این الگوریتم‌ها در بازار سهام چین انجام دادند که نشان داد گرادیان سیاست نسبت به دو روش دیگر در بازارهای مالی بهتر عمل می‌کند (Liang et al., 2018).

فیلوس (۲۰۱۸) برای مدیریت سبد سهام به مقایسه رویکردهای یادگیری تقویتی مبتنی بر مدل و بدون مدل می‌پردازد. نتایج این پژوهش نشان می‌دهد که روش‌های یادگیری تقویتی بدون مدل نظیر روش‌های مبتنی بر یادگیری Q نه تنها قادر به کاهش پیچیدگی محاسباتی و حافظه هستند، بلکه قابلیت تعمیم بالایی به بازارهای مختلف دارند. نویسنده از داده‌های شبیه‌سازی شده و واقعی برای ارزیابی الگوریتم‌ها استفاده کرد و نشان داد که یادگیری تقویتی بدون مدل ۹/۲ درصد بازده تجمعی سالانه بهتری به دست می‌دهد (Filos, 2018). در این مطالعه از روش‌های فقط منتقد و فقط بازیگر استفاده شد.

۲-۵) جمع‌بندی

با نگاهی به پیشینه پژوهش می‌توان دریافت که رویکرد یادگیری تقویتی فقط منتقد در مقایسه با دو رویکرد دیگر، بیشترین مطالعات را در حوزه مالی به خود اختصاص داده است. با این حال مهمترین محدودیت این رویکرد به ویژه روش‌های مبتنی بر DQN آن است که فضای اقدام این دست روش‌ها نوعا گسسته است و نیازمند تغییراتی هستند تا بتوانند برای قیمت‌های پیوسته سهام مورد استفاده قرار بگیرند. این محدودیت هنگامی برجسته‌تر می‌شود که نیاز باشد درباره معامله چندین سهام تصمیم‌گیری شود. در این صورت فضای حالت و اقدام برای تعیین وزن هر یک از سهام‌ها به صورت نمایی رشد می‌کند که این امر کاربردپذیری این دست روش‌ها را در عمل تحت تاثیر قرار می‌دهد (Xiong et al., 2022).

رویکرد فقط بازیگر در جایگاه دوم در بین سایر رویکردها از منظر تعداد مطالعات انجام شده قرار دارد. مزیت کلیدی رویکرد فقط بازیگر، فضای اقدام پیوسته آن برای بدست آوردن وزن سهام‌ها است. با این حال زمان مورد نیاز برای یادگیری سیاست بهینه در این رویکرد نوعا بالاتر است (Zhang et al., 2020). رویکرد بازیگر-منتقد سعی دارد تا با بهره‌گیری همزمان از نقاط قوت دو رویکرد دیگر، بر محدودیت‌های مورد اشاره فائق آید. با این حال و بر خلاف مزیت‌های بالقوه آن، مطالعات نسبتا کمی با استفاده از این رویکرد در بازارهای مالی انجام شده است. برای پوشش این شکاف در این پژوهش روشی مبتنی بر رویکرد بازیگر-منتقد برای بهینه‌سازی سبد سهام ارائه خواهد شد. سپس عملکرد

[10] Deep Deterministic Policy Gradient
[11] Proximal Policy Optimization
[12] Policy Gradient

روش پیشنهادی با دو روش یادگیری تقویتی عمیق دیگر و همچنین با استراتژی‌های مرسوم سرمایه‌گذاری مقایسه خواهد شد.

۳) بیان مساله

مساله اصلی این پژوهش، بهینه‌سازی سبد سهام با استفاده از هوش مصنوعی و به صورت مشخص روش‌های مبتنی بر یادگیری تقویتی است. در این راستا فرض کنید که به دنبال الگوریتمی برای خرید و فروش سهام هستیم که بتواند در طول دوره‌های زمانی مشخص اقدام به خرید و فروش مجموعه‌ای از سهام‌های از پیش تعریف شده کند به نحوی که ارزش پایانی سبد سهام بیشینه شود. در ابتدای هر دوره زمانی، سرمایه‌گذار می‌تواند ترکیب سبد سهام خود را تغییر دهد. در طول دوره قیمت هر سهم می‌تواند تغییر کند. با این حال چهار ویژگی قیمت آغازین، بیشترین قیمت، کمترین قیمت و قیمت پایانی برای هر یک از سهام در انتهای هر دوره زمانی در دسترس خواهد بود. طول دوره‌های زمانی مساوی و به صورت روزانه در نظر گرفته می‌شود. فرض می‌شود که در ابتدای هر دوره سرمایه‌گذار می‌تواند هر سهم را به قیمت آغازین آن بخرد یا بفروشد.

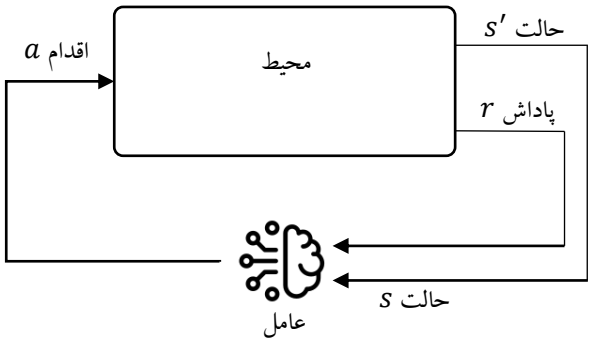
با توجه به ماهیت پویا و تعاملی مساله قصد داریم تا با استفاده از مفاهیم یادگیری تقویتی اقدام به مدل‌بندی و حل مساله نماییم. یادگیری تقویتی یکی از روش‌های یادگیری ماشین است که به واسطه دو مولفه زیر از سایر روش‌های موجود در این حوزه نظیر یادگیری بانظارت و یادگیری بدون نظارت متمایز می‌گردد (Sutton & Barto, 2018).

۱- محیط[۱۳]: سیستمی است که شرایط، قوانین و دینامیک مساله را در بر دارد و می‌تواند یک بازی ویدیویی، یک شبیه‌ساز، یا بازار سهام باشد.

۲- عامل[۱۴]: در واقع هوش مصنوعی است که یاد می‌گیرد چگونه در یک محیط معین تصمیم‌گیری کند و موفق شود.

روشی که عامل یاد می‌گیرد چگونه در محیط کار کند از طریق یک حلقه بازخورد تکراری است. عامل در ابتدا در حالت[۱۵] فعلی s قرار دارد سپس اقدام[۱۶] a را انجام می‌دهد و در نتیجه آن پاداش r را که می‌تواند مثبت یا منفی باشد دریافت می‌کند و به حالت جدید s' می‌رسد (شکل ۱). این تجربه عامل به صورت $< s, a, r, s' >$ نشان داده می‌شود. هدف عامل این است که در تعامل با محیط، دنباله‌ای از اقدام‌ها را انتخاب کند که نوعا مجموع کل پاداش دریافتی را بیشینه نماید. یکی از راه‌های مدل‌سازی این مساله استفاده از فرایند تصمیم‌گیری مارکوف است.

[13] Environment
[14] Agent
[15] State
[16] Action



شکل ۱. نحوه تعامل عامل با محیط

فرایند تصمیم‌گیری مارکف یک سیستم پویای تصادفی است که به صورت $M =< S, \mathcal{A}, \mathcal{P}, R, \gamma >$ نمایش داده می‌شود که در آن S فضای حالت، $\mathcal{A}$ فضای اقدام، $\mathcal{P}$ تابع انتقال، R تابع پاداش و γ نرخ تخفیف است (Graesser & Keng, 2019).

۳-۱) فضای حالت

فضای حالت مجموعه‌ای از حالت‌هاست که از خاصیت مارکف تبعیت می‌کنند. طبق خاصیت مارکف برای هر حالت s_t، حالت قبلی یعنی s_{t-1} حاوی تمامی اطلاعات مورد نیاز برای تعریف تابع احتمال آن است (van Otterlo & Wiering, 2012). یعنی داریم:

$$P[s_t|s_{t-1}, s_{t-2}, \dots, s_1] = P[s_t|s_{t-1}] \quad (۱)$$

تابع انتقال $\mathcal{P}$ احتمال رفتن به یک حالت مشخص s_{t+1} بعد از انجام اقدام a_t در حالت s_t را مشخص می‌نماید. بنابراین داریم:

$$\mathcal{P}_{s,s'} = P[s_{t+1} = s'|s_t = s, a_t = a] \quad (۲)$$

به عبارت دیگر فضای حالت مشاهدات و اطلاعاتی است که عامل از محیط در هر دوره زمانی دریافت می‌کند و با کمک آن‌ها سعی می‌کند تا یادگیری خود را بهبود بخشد. در این مقاله، حالت در انتهای هر دوره زمانی با استفاده از ماتریس کوواریانس محاسبه شده بر مبنای قیمت پایانی سهام‌ها در طول یک سال گذشته تعریف می‌شود.

۳-۲) فضای اقدام

فضای اقدام، اقداماتی را توصیف می‌کند که عامل می‌تواند در تعامل با محیط انجام دهد (Sutton & Barto, 2018). در این مقاله عامل می‌تواند وزن اختصاص یافته به سهام‌ها را در سبد سهام در هر دوره زمانی تغییر دهد. این وزن تخصیص سرمایه به هر سهم را بر مبنای ارزش کل سبد سهام در هر دوره نشان می‌دهد. لازم به ذکر است که مجموع وزن‌های اختصاص یافته باید یک شود. در نهایت این بردار وزن به عنوان اقدام در هر دوره زمانی تعریف می‌شود.

$$\sum_{i=1}^{n} w_{i,t} = 1 \quad (۳)$$

که در آن $w_{i,t}$ وزن سهام i در دوره t از ارزش کل سبد سهام است.

۳-۳) تابع پاداش

تابع پاداش محرکی برای عامل است تا بتواند اقدامات بهتری را یاد بگیرد. در این مساله تابع پاداش به صورت ارزش کل سبد سهام تعریف می‌شود. داریم:

$$R_t = \mathcal{V}_{t-1}\left[\sum_{i=1}^{n} \frac{p_{i,t}^{c}}{p_{i,t-1}^{c}} w_{i,t-1}\right] \quad (۴)$$

که در آن $\mathcal{V}_t$ ارزش کل سبد سهام در دوره t و $p_{i,t}^{c}$ قیمت پایانی سهام i در دوره t است.

عامل در تعامل با محیط باید تصمیم بگیرد که در هر دوره زمانی t چه اقدام a_t را برحسب حالت فعلی s_t انجام دهد به نحوی که در نهایت مجموع پاداش تخفیف یافته دریافتی G_t بیشینه شود (Graesser & Keng, 2019).

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \quad (۵)$$

این تصمیم بر مبنای سیاست عامل تعیین می‌شود. تابع سیاست π تابعی است که رفتار عامل در هر حالت را مشخص می‌سازد.

۴) روش حل

در این بخش روشی برای بهینه‌سازی سبد سهام با استفاده از یادگیری تقویتی عمیق Q ارائه می‌دهیم. در ادامه ابتدا مفاهیم مورد نیاز را تعریف کرده و سپس به بیان الگوریتم حل خواهیم پرداخت.

تابع ارزش[17] $V_\pi(s)$، ارزش حالت s تحت سیاست π را نشان می‌دهد. به عبارت دیگر این تابع، پاداش مورد انتظار با شروع از حالت s و در ادامه پیگیری سیاست π است و به صورت زیر تعریف می‌شود (Sutton & Barto, 2018).

$$V_\pi(s) = E_\pi[G_t | s_t = s] \quad (۶)$$

که در آن $E_\pi[.]$ امید ریاضی با فرض پیروی عامل از سیاست π است.

به صورت مشابه می‌توانیم تابع اقدام-ارزش[18] $Q_\pi(s,a)$ را به صورت پاداش مورد انتظار برای اتخاذ اقدام a در حالت s و در ادامه پیگیری سیاست π و به صورت زیر تعریف نماییم (Sutton & Barto, 2018).

$$Q_\pi(s,a) = E_\pi[G_t | s_t = s, a_t = a] \quad (۷)$$

[17] Value function
[18] Action-value function

تابع ارزش و تابع اقدام–ارزش به صورت غیرمستقیم برای یافتن سیاست بهینه مورد استفاده قرار می‌گیرند. سیاست بهینه π^* یعنی سیاستی که به مجموع پاداش بیشینه برسد. می‌توان نشان داد که برای فرایند تصمیم‌گیری مارکف، سیاست بهینه π^* وجود دارد و تحت این سیاست، تابع ارزش و تابع اقدام–ارزش به مقدار بهینه خود خواهند رسید (Poole & Mackworth, 2023). لذا برای یافتن سیاست بهینه تنها کافی است این توابع را بهینه نماییم. داریم:

$$Q^*(s,a) = Q_{\pi^*}(s,a) = \max_{\pi} Q_{\pi}(s,a) \quad (۸)$$

در روش یادگیری تقویتی Q برای یافتن تابع اقدام–ارزش بهینه بر اساس تجربه $< s,a,r,s' >$ از یک فرمول تقریبی و بازگشتی به صورت زیر استفاده می‌کند که در طول گام‌های مختلف الگوریتم بهبود می‌یابد.

$$Q(s,a) \leftarrow Q(s,a) + \alpha \left[r + \gamma \max_{a' \in \mathcal{A}} Q(s',a') - Q(s,a) \right] \quad (۹)$$

که در آن $\alpha \geq 0$ نرخ یادگیری الگوریتم است و عبارت داخل براکت نیز نوعا خطای تفاوت زمانی[19] نامیده می‌شود (Sutton & Barto, 2018).

۴–۱) شبکه عمیق Q

یادگیری Q زمانی که محیط نسبتاً ساده‌ای برای حل داریم به خوبی کار می‌کند، اما وقتی تعداد حالت‌ها و اقداماتی که می‌توانیم انجام دهیم پیچیده‌تر می‌شود، از شبکه عمیق Q بهره می‌بریم. در این روش، الگوریتم یادگیری Q با دو شبکه عصبی مصنوعی یکی به عنوان شبکه یادگیری با پارامترهای θ و دیگری به عنوان شبکه هدف با پارامترهای θ^- ادغام می‌شود. این پارامترها در حقیقت معرف وزن‌های شبکه عصبی هستند و برای یادگیری آن‌ها نیاز است تا تابع زیان شبکه را با استفاده از یکی از الگوریتم‌های کاهش گرادیان کمینه نماییم. بدین منظور نیاز است تا رابطه (۹) به صورت تابع زیانی مبتنی بر مجموع مربعات خطا در نظر گرفته شود (Wang et al., 2016). داریم:

$$L(\theta) = E_{s,a,r,s'} \left[\left(r + \gamma \max_{a' \in \mathcal{A}} Q(s',a';\theta^-) - Q(s,a;\theta) \right)^2 \right] \quad (۱۰)$$

بنابراین برای به‌روزرسانی θ از رابطه زیر استفاده خواهیم کرد.

$$\theta \leftarrow \theta + \alpha [y(r,s') - Q(s,a;\theta)] \nabla_{\theta} Q(s,a;\theta) \quad (۱۱)$$

که در آن داریم:

$$y(r,s') = r + \gamma \max_{a' \in \mathcal{A}} Q(s',a';\theta^-) \quad (۱۲)$$

لازم به ذکر است که برای بهبود عملکرد الگوریتم، از مفهومی به نام بازپخش تجربه[20] استفاده می‌شود که ذخیره‌سازی مجموعه‌ای از تجربیات مشاهده شده توسط عامل است. سپس برای به‌روزرسانی شبکه از این حافظه به صورت تصادفی یکنواخت نمونه‌گیری خواهد شد (Mnih et al., 2015).

[19] Temporal Difference Error
[20] Experience Replay

۴-۲) فضای اقدام پیوسته

آنچه درباره شبکه عمیق Q بیان شد نوعا برای مسائل با فضای اقدام گسسته کاربرد دارد. از آنجایی که فضای اقدام این مقاله پیوسته است نیاز است تا در نحوه عملکرد شبکه عمیق Q تغییراتی را ایجاد نماییم که در ادامه توضیح داده خواهند شد.

ابتدا باید توجه کرد که هنگامی که فضای اقدام پیوسته است یافتن بیشینه تابع اقدام-ارزش Q_{θ^-} در رابطه (۱۰) بر روی تمامی اقدامات ممکن کار چالش‌برانگیزی است. برای رفع این مشکل از یک شبکه عصبی دیگر با نام شبکه سیاست هدف بهره گرفته می‌شود. این شبکه اقدامی را به دست می‌دهد که به صورت تقریبی Q_{θ^-} را بیشینه می‌نماید. بدین ترتیب تابع زیان به صورت زیر خواهد شد.

$$L(\theta) = E_{s,a,r,s'}\left[\left(r + \gamma Q(s', \mu(s'; \phi^-); \theta^-) - Q(s, a; \theta)\right)^2\right] \quad (۱۳)$$

که در آن $\mu(s'; \phi^-)$ معرف خروجی شبکه سیاست هدف با پارامترهای ϕ^- به ازای حالت s' است.

برای یادگیری سیاستی که اقدامی را بدهد که تابع Q را بیشینه نماید از آنجایی که فضای اقدام پیوسته است فرض می‌کنیم که تابع Q نسبت به متغیر اقدام مشتق‌پذیر است. بنابراین می‌توانیم از روش کاهش گرادیان برای حل عبارت زیر استفاده کنیم (Lillicrap et al., 2019).

$$\max_{\phi} E_s[Q(s, \mu(s; \phi); \theta)] \quad (۱۴)$$

توجه کنید که پارامترهای تابع Q در این عبارت به عنوان ثابت فرض می‌شوند.

لازم به ذکر است که برای افزایش پایداری الگوریتم به جای به‌روزرسانی وزن‌های شبکه هدف در تکرارهای مشخص، با هر به‌روزرسانی شبکه یادگیری، وزن‌های شبکه هدف به صورت زیر به‌روزرسانی می‌شوند.

$$\theta^- \leftarrow \rho\theta^- + (1 - \rho)\theta \quad (۱۵)$$

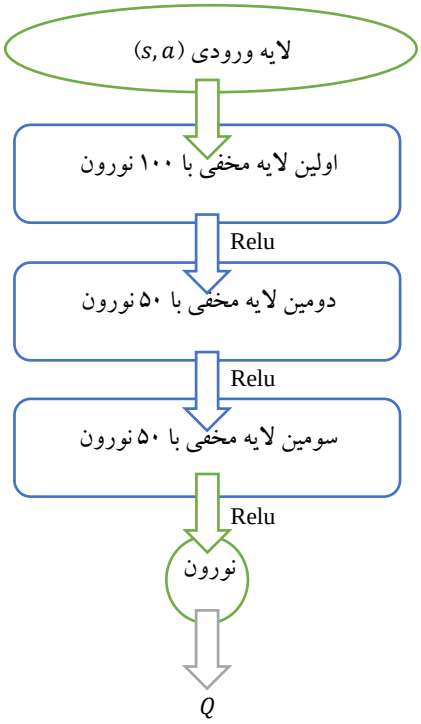
که در آن ρ پارامتری بین صفر و یک و نوعا نزدیک به یک است تا تغییرات وزن شبکه هدف به آرامی صورت گیرد. شبکه سیاست هدف نیز مشابه با شبکه هدف کار می‌کند و وزن‌های آن با استفاده از عبارت زیر به‌روزرسانی می‌شوند.

$$\phi^- \leftarrow \rho\phi^- + (1 - \rho)\phi \quad (۱۶)$$

۴-۳) ساختار شبکه عصبی

در الگوریتم معرفی شده در این بخش از چهار شبکه عصبی مختلف استفاده می‌شود که عبارت‌اند از: شبکه یادگیری، شبکه هدف، شبکه سیاست، و شبکه سیاست هدف. در ادامه ساختار هر یک از این شبکه‌ها معرفی می‌شود. لازم به ذکر است که برای رسیدن به این ساختارهای پیشنهادی، ساختارهای چند لایه عمیق مختلفی مورد بررسی قرار گرفته‌اند و نتایج حاصل از آن‌ها مورد ارزیابی و مقایسه با یکدیگر قرار گرفته است و در نهایت ساختارهای با بهترین عملکرد در این بخش گزارش شده‌اند.

شکل ۲ ساختار پیشنهادی برای شبکه یادگیری و شبکه هدف را نشان می‌دهد. مطابق این شکل، این دو شبکه از سه لایه مخفی تماما متصل تشکیل شده‌اند. تابع فعالیت نورون‌ها در این لایه‌ها از نوع تابع یک‌سوساز خطی[۲۱] (ReLU) است. تعداد نورون‌ها در لایه ورودی برابر با مجموع تعداد مولفه‌های حالت s و اقدام a است و تعداد نورون‌های لایه خروجی برابر با یک و معرف مقدار Q است.

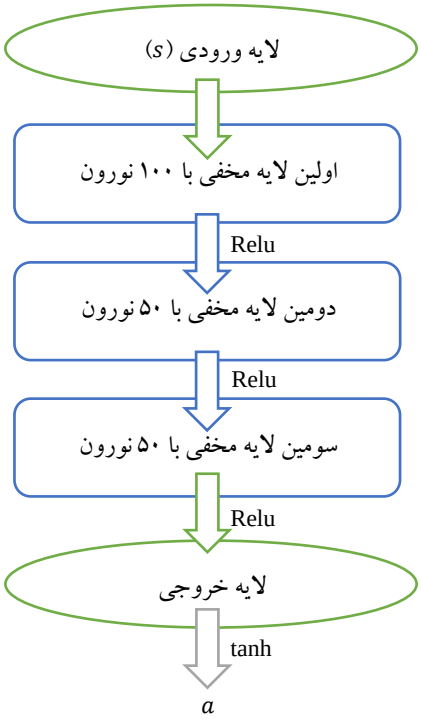


شکل ۲. ساختار شبکه یادگیری و شبکه هدف

توجه کنید که ساختار شبکه یادگیری و شبکه هدف کاملا یکسان بوده و تنها در نحوه به‌روزرسانی وزن با یکدیگر متفاوت می‌باشند. وزن‌های شبکه هدف با استفاده از رابطه (۱۵) و بر اساس وزن‌های شبکه یادگیری محاسبه می‌شوند.

شکل ۳ ساختار پیشنهادی برای شبکه سیاست و شبکه سیاست هدف را نشان می‌دهد. مطابق این شکل، این دو شبکه از سه لایه مخفی تماما متصل تشکیل شده‌اند. تابع فعالیت نورون‌ها در این لایه‌ها از نوع تابع یک‌سوساز خطی است. بعلاوه تابع فعالیت نورون‌ها در لایه خروجی از نوع تابع تانژانت هیپربولیک[۲۲] است. تعداد نورون‌ها در لایه ورودی برابر با تعداد مولفه‌های حالت s است و تعداد نورون‌های لایه خروجی برابر با تعداد مولفه‌های اقدام a است.

[21] Rectified Linear Unit
[22] tanh



شکل ۳. ساختار شبکه سیاست و شبکه سیاست هدف

توجه کنید که ساختار شبکه سیاست و شبکه سیاست هدف کاملا یکسان است و وزن‌های شبکه سیاست هدف با استفاده از رابطه (۱۶) و بر اساس وزن‌های شبکه سیاست به‌روزرسانی می‌شوند.

برای آموزش این شبکه‌ها از الگوریتم آدام[۲۳] به عنوان یک روش بهینه‌سازی مبتنی بر گرادیان کاهشی تصادفی با پارامترهای زیر استفاده می‌شود (Kingma & Ba, 2015).

$$\alpha = 0.001, \beta_1 = 0.9, \beta_2 = 0.999, \epsilon = 1e^{-8}$$

که در آن α نرخ یادگیری، β_1 نرخ کاهش نمایی برای تخمین گشتاور مرتبه اول، β_2 نرخ کاهش نمایی برای تخمین گشتاور مرتبه دوم، و ϵ یک مقدار کوچک برای ثبات عددی الگوریتم است.

۴-۴) الگوریتم حل

در ادامه الگوریتم حل پیشنهادی به صورت گام به گام ارائه می‌شود.

گام ۱. ارزش اولیه سبد سهام یا سرمایه اولیه $(\mathcal{V}_0)$، تعداد سهام‌ها (n)، نرخ تخفیف (γ)، نرخ یادگیری (α)، ضریب به‌روزرسانی وزن‌های شبکه هدف و شبکه سیاست هدف (ρ)، تعداد نمونه‌های موجود در هر دسته آموزشی B $(|B|)$، تعداد تجربه‌های قابل ذخیره‌سازی در بازپخش تجربه D $(|D|)$، تعداد تکرارهای اولیه برای آماده‌سازی الگوریتم $(iters_{warmup})$، و تعداد کل تکرارهای الگوریتم $(iters_{total})$ را مقداردهی اولیه نماید.

[23] Adam

گام ۲. شبکه‌های یادگیری، یادگیری هدف، سیاست و سیاست هدف را مطابق با ساختار چند لایه پیشنهادی در شکل‌های ۲ و ۳ ایجاد نمایید.

گام ۳. وزن‌های شبکه یادگیری (θ) و شبکه سیاست (ϕ) را مقداردهی اولیه نموده و حافظه مربوط به ذخیره‌سازی بازپخش تجربه (D) را خالی کنید.

گام ۴. مقادیر پارامترهای شبکه هدف (θ^-) و شبکه سیاست هدف (ϕ^-) را به ترتیب برابر با پارامترهای متناظر در شبکه یادگیری و شبکه سیاست در نظر بگیرید.

$$\theta^- \leftarrow \theta, \phi^- \leftarrow \phi$$

گام ۵. فضای اقدام را به صورت یک مربع چند بعدی با مقادیر بین صفر و یک تعریف کنید. تعداد ابعاد این مربع برابر با تعداد کل سهام‌ها است.

گام ۶. فضای حالت در انتهای هر دوره زمانی را برابر با ماتریس کوواریانس محاسبه شده بر مبنای قیمت پایانی سهام‌ها در طول یک سال گذشته تعریف نمایید.

گام ۷. تا وقتی تکرارهای الگوریتم به مقدار $iters_{total}$ نرسیده است گام‌های زیر را تکرار نمایید.

گام ۸. تا وقتی به آخرین دوره زمانی نرسیده‌اید گام‌های زیر را تکرار کنید (ایجاد بازپخش تجربه).

گام ۸-۱. تا وقتی تعداد تکرارهای گام هشتم کمتر از $iters_{warmup}$ است گام ۸-۲ را انجام دهید. در غیر اینصورت به گام ۸-۳ بروید.

گام ۸-۲. اقدام $\bar{a}$ (وزن سهام‌ها) را به صورت تصادفی از فضای اقدام تعریف شده در گام پنجم انتخاب کنید.

گام ۸-۳. آخرین حالت مشاهده شده s را به عنوان ورودی به شبکه سیاست داده و اقدام متناظر با آن را به عنوان خروجی شبکه دریافت کنید. سپس خروجی شبکه را به بازه بین صفر و یک نگاشت نمایید و آن را اقدام $\bar{a}$ بنامید.

گام ۸-۴. اقدام $\bar{a}$ را به بازه منفی یک و یک نگاشت نمایید و آن را اقدام مقیاس شده a بنامید.

گام ۸-۵. اقدام انتخاب شده $\bar{a}$ را با استفاده از تابع softmax نرمال‌سازی نمایید تا مجموع وزنی آن برابر با یک شود.

گام ۸-۶. به دوره زمانی یک واحد اضافه نمایید و آن را t بنامید.

گام ۸-۷. حالت جدید s_t (s') را در دوره زمانی جدید t محاسبه نمایید (ماتریس کوواریانس قیمت پایانی سهام‌ها در یک سال گذشته).

گام ۸-۸. پاداش R_t (r) را بر اساس ارزش کل سبد سهام و به صورت زیر محاسبه کنید.

$$R_t = \mathcal{V}_{t-1}\left[\sum_{i=1}^{n} \frac{p_{i,t}^c}{p_{i,t-1}^c} w_{i,t-1}\right]$$

گام ۸-۹. تجربه $< s, a, r, s' >$ را در D ذخیره کنید.

گام ۹. به تعداد تکرارهای گام ۸، گام‌های زیر را تکرار کنید (به‌روزرسانی وزن‌های شبکه‌های عصبی).

گام ۹-۱. به تعداد $|B|$ از تجربه‌های ذخیره شده در D نمونه انتخاب کنید و آن‌ها را در مجموعه B قرار دهید.

گام ۹-۲. حالت‌های s' در مجموعه B را به شبکه سیاست هدف داده و اقدام‌های جدید متناظر را به عنوان خروجی شبکه $(\mu(s';\phi^-))$ به دست آورید.

گام ۹-۳. حالت‌های s' در مجموعه B و اقدام‌های جدید $\mu(s';\phi^-)$ به دست آمده در گام قبل را به شبکه هدف دهید تا مقادیر $Q(s',\mu(s';\phi^-);\theta^-)$ متناظر جدید به عنوان خروجی حاصل شوند.

گام ۹-۴. حالت‌های s و اقدام‌های a در مجموعه B را به شبکه یادگیری داده و مقادیر $Q(s,a;\theta)$ را به دست آورید.

گام ۹-۵. مقدار Q هدف را به صورت زیر محاسبه کنید.

$$y(r,s') = r + \gamma Q(s',\mu(s';\phi^-);\theta^-)$$

گام ۹-۶. وزن‌های شبکه یادگیری را با روش کاهش گرادیان و با استفاده از عبارت زیر به‌روزرسانی کنید.

$$\nabla_\theta \frac{1}{|B|} \sum_{(s,a,r,s')\in B} \big(y(r,s') - Q(s,a;\theta)\big)^2$$

گام ۹-۷. حالت‌های s در مجموعه B را به شبکه سیاست داده و اقدام‌های متناظر را به عنوان خروجی شبکه $(\mu(s;\phi))$ به دست آورید. سپس این اقدام‌ها را به همراه همان حالت‌های s به شبکه یادگیری دهید تا مقادیر $Q(s,\mu(s;\phi);\theta)$ به عنوان خروجی حاصل شوند.

گام ۹-۸. وزن‌های شبکه سیاست را با روش کاهش گرادیان و با استفاده از عبارت زیر به‌روزرسانی کنید.

$$\nabla_\theta \frac{1}{|B|} \sum_{s\in B} Q(s,\mu(s;\phi);\theta)$$

گام ۹-۹. وزن‌های شبکه هدف و شبکه سیاست هدف را با کمک روابط زیر به‌روزرسانی کنید.

$$\theta^- \leftarrow \rho\theta^- + (1-\rho)\theta$$
$$\phi^- \leftarrow \rho\phi^- + (1-\rho)\phi$$

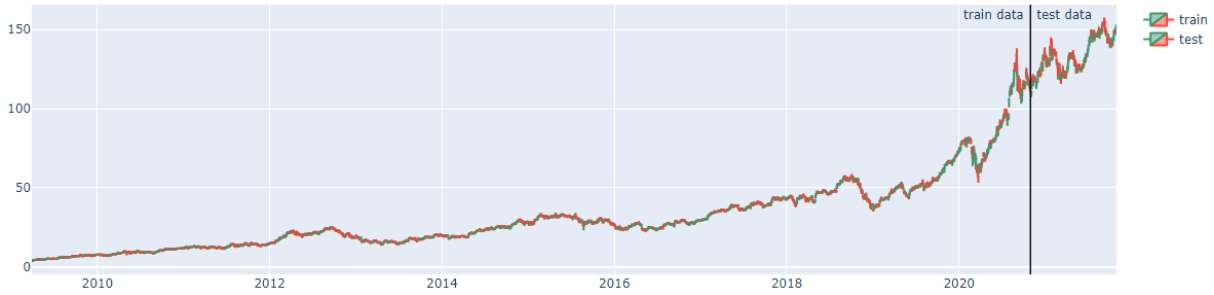
۵) نتایج عددی

در این بخش مدل پیشنهادی یادگیری تقویتی عمیق Q در زبان برنامه‌نویسی پایتون پیاده‌سازی شد و عملکرد آن با استفاده از داده‌های واقعی مورد ارزیابی قرار گرفت. بدین منظور از داده‌های مربوط به شرکت‌های تشکیل‌دهنده شاخص داو جونز (DJIA) در بورس نیویورک (NYSE) و بورس نزدک (NASDAQ) استفاده شد که مشخصات آن‌ها در جدول ۱ آمده است. برای هر یک از این شرکت‌ها اطلاعات روزانه شامل تاریخ، قیمت آغازین، بالاترین قیمت، کمترین قیمت، قیمت پایانی و حجم معامله از ۳۰ مارس ۲۰۰۸ تا ۳۱اکتبر ۲۰۲۱ با استفاده از کتابخانه yfinance در زبان پایتون استخراج شد که شامل ۳۴۲۲ روز معاملاتی می‌شود. این کتابخانه امکان دانلود داده‌های مالی را با بهره‌گیری از Yahoo! Finance API مهیا می‌سازد (Aroussi, 2024). نتایج این بخش با استفاده از یک کامپیوتر با پردازنده Corei7 و حافظه تصادفی ۲۰ گیگابایت حاصل شده است.

جدول ۱. مشخصات شرکت‌های مدنظر در این مطالعه

نماد	نام شرکت	نماد	نام شرکت
AAPL	.Apple Inc	JPM	.JPMorgan Chase & Co
AMGN	.Amgen Inc	KO	The Coca-Cola Company
AXP	American Express Company	MCD	McDonald's Corporation
BA	The Boeing Company	MMM	3M Company
CAT	.Caterpillar Inc	MRK	.Merck & Co., Inc
CRM	.Salesforce, Inc	MSFT	Microsoft Corporation
CSCO	.Cisco Systems, Inc	NKE	.NIKE, Inc
CVX	Chevron Corporation	PG	The Procter & Gamble Company
DIS	The Walt Disney Company	TRV	.The Travelers Companies, Inc
GS	.The Goldman Sachs Group, Inc	UNH	UnitedHealth Group Incorporated
HD	.The Home Depot, Inc	V	.Visa Inc
HON	.Honeywell International Inc	VZ	.Verizon Communications Inc
IBM	International Business Machines Corporation	WBA	.Walgreens Boots Alliance, Inc
INTC	Intel Corporation	WMT	Walmart Inc.
JNJ	Johnson & Johnson		

تاریخ ۳۰ اکتبر ۲۰۲۰ برای جداسازی داده‌های آموزش از آزمون انتخاب شد. با کمک داده‌های جمع‌آوری شده در این پژوهش می‌توان نمودار شمعی سهام شرکت اپل را ترسیم نمود (شکل ۴). در این شکل بخش‌بندی داده‌ها به داده آموزش و آزمون با استفاده از یک خط عمودی در سمت راست نمودار نشان داده شده است.



شکل ٤. نمودار شمعی سهام شرکت اپل در طول دوره آزمون و آزمایش

بعد از آموزش الگوریتم پیشنهادی بر روی داده‌های آموزش، نتایج آن بر روی داده‌های آزمون بررسی شد. مقادیر پارامترهای مورد استفاده برای آموزش این الگوریتم در جدول ۲ ارائه شده است.

جدول ۲. مقادیر پارامترهای مورد استفاده در الگوریتم پیشنهادی

پارامتر	مقدار	پارامتر	مقدار
$\mathcal{V}_0$	1,000,000	$\|B\|$	128
n	29	$\|D\|$	50,000
γ	0.99	$iters_{warmup}$	100

پارامتر	مقدار	پارامتر	مقدار
α	0.001	$iters_{total}$	50,000
ρ	0.995		

برای ارزیابی عملکرد این الگوریتم، از مقایسه نتایج آن با دو استراتژی سرمایه‌گذاری مرسوم (استراتژی سرمایه‌گذاری غیرفعال و استراتژی کمینه واریانس) و دو الگوریتم یادگیری تقویتی عمیق (بهینه‌سازی سیاست پروکسیمال (PPO) و بازیگر-منتقد نرم[۲۴] (SAC)) استفاده شد که در ادامه به شرح مختصری از هر یک خواهیم پرداخت.

استراتژی سرمایه‌گذاری غیرفعال یکی از روش‌های شناخته شده سرمایه‌گذاری است و در آن معامله‌گر به جای خرید و فروش مستقیم سهام شرکت‌های مختلف، اقدام به خرید و فروش شاخص‌ها یا صندوق‌های قابل معامله می‌نماید (Malkiel, 2003). با توجه به سهام شرکت‌های مورد استفاده در این پژوهش، نتایج الگوریتم پیشنهادی را با بازده حاصل از خرید و فروش شاخص داو جونز (DJIA) مقایسه خواهیم کرد. شاخص داو جونز یکی از مهم‌ترین شاخص‌های مالی در بورس آمریکا و به نوعی نشانگر عملکرد بازار است. در این راستا ابتدا اطلاعات مربوط به شاخص DJIA در دوره زمانی متناظر با داده‌های آزمون با استفاده از کتابخانه yfinance از داده‌های تاریخی موجود استخراج می‌شود. سپس بازده روزانه این شاخص محاسبه شده و با بهره‌گیری از کتابخانه pyfolio درصد بازده نهایی شاخص DJIA محاسبه می‌گردد (Quantopian Inc, 2019).

استراتژی کمینه واریانس (Min-Variance) یکی دیگر از استراتژی‌های شناخته شده است که به دنبال ایجاد تعادلی بین ریسک و سود است (Ang, 2012). بدین منظور این استراتژی با ایجاد تنوع در سبد سهام سعی دارد تا ضمن رسیدن به سود بیشتر میزان ریسک (واریانس) را کاهش دهد. برای یافتن سبد سهام بهینه مبتنی بر استراتژی کمینه واریانس از کتابخانه PyPortfolioOpt کمک گرفته شده است (Martin, 2021). این کتابخانه با محاسبه ماتریس کوواریانس برای داده‌های آزمون بر مبنای تاریخچه بازده سهام‌ها سعی می‌نماید که ترکیب وزنی در سبد سهام را در هر دوره به نحوی ارائه دهد که میزان واریانس کمینه شود.

الگوریتم بهینه‌سازی سیاست پروکسیمال (PPO) خانواده جدیدی از روش‌های مبتنی بر گرادیان سیاست (PG) است که سعی دارد با محدود کردن تغییرات در به‌روزرسانی سیاست در هر گام، پایداری یادگیری را بهبود بخشد. بدین منظور این الگوریتم از یک روش محافظه‌کارانه برای به‌روزرسانی سیاست استفاده می‌کند به نحوی که نسبت تغییر سیاست در هر گام نسبت به گام قبل در یک بازه مشخص قرار گیرد. برای این محدودسازی از تابع هدف جدیدی با نام تابع هدف جایگزین بریده شده[۲۵] بهره گرفته می‌شود تا از به‌روزرسانی‌های بزرگ و نوعا مخرب در وزن‌های شبکه سیاست جلوگیری به عمل آید (Schulman et al., 2017).

الگوریتم بازیگر-منتقد نرم (SAC) یک روش بدون مدل و مبتنی بر بیشینه آنتروپی است. در این روش بازیگر سعی دارد تا علاوه بر پاداش مورد انتظار، آنتروپی را نیز بیشینه نماید. به عبارت دیگر این روش به دنبال آن است که بازیگر در رسیدن به هدف موفق شود و تا جای ممکن نیز به صورت تصادفی عمل نماید. این الگوریتم به دنبال یادگیری

[24] Soft Actor-Critic
[25] Clipped Surrogate Objective Function

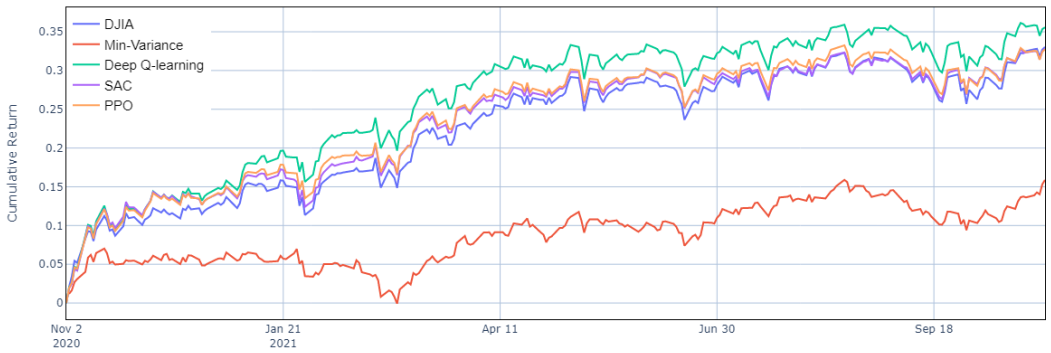
سیاست‌های تصادفی با استفاده از بیشینه کردن آنتروپی در تابع ارزش و سیاست است که هم از همگرایی زودرس سیاست جلوگیری می‌کند و هم موجب کاوش بیشتر فضای حالت می‌شود (Haarnoja et al., 2018).

جدول ۳ عملکرد الگوریتم پیشنهادی را بر روی داده‌های آزمون در مقایسه با نتایج حاصل از پیاده‌سازی و استفاده از شاخص داو جونز، استراتژی کمینه واریانس، الگوریتم PPO و الگوریتم SAC نشان می‌دهد. مطابق این جدول الگوریتم پیشنهادی با مجموع بازده ۳۵.۶ درصد بهترین عملکرد را در بین سایر استراتژی‌ها و الگوریتم‌های مورد بررسی داشته است. بعلاوه نسبت شارپ (Sharpe, 1994) الگوریتم پیشنهادی بیشترین مقدار است که نشانگر آن است که این استراتژی در متعادل‌سازی بین سود و ریسک عملکرد بهتری دارد.

جدول ۳. مقایسه الگوریتم پیشنهادی با استراتژی‌های مرسوم و الگوریتم‌های یادگیری تقویتی عمیق

استراتژی شاخص	Q-learning Deep	SAC	PPO	DJIA	Min-Varianve
سرمایه اولیه	1,000,000	1,000,000	1,000,000	1,000,000	1,000,000
سرمایه نهایی	1,355,700	1,328,128	1,327,503	1,330,343	1,158,860
درصد بازده	35.6%	32.8%	32.7%	33.0%	15.9%
نسبت شارپ	2.40	2.30	2.33	2.36	1.52

شکل ۵ مجموع بازده هر یک از استراتژی‌های مورد بررسی را به صورت روزانه و در طول بازه آزمون نشان می‌دهد. مطابق این شکل مشاهده می‌شود که الگوریتم پیشنهادی به صورت روزانه نیز عملکرد بهتری را در کل دوره زمانی آزمون نسبت به سایر استراتژی‌های مرسوم و الگوریتم‌های یادگیری تقویتی عمیق مورد بررسی دارد.

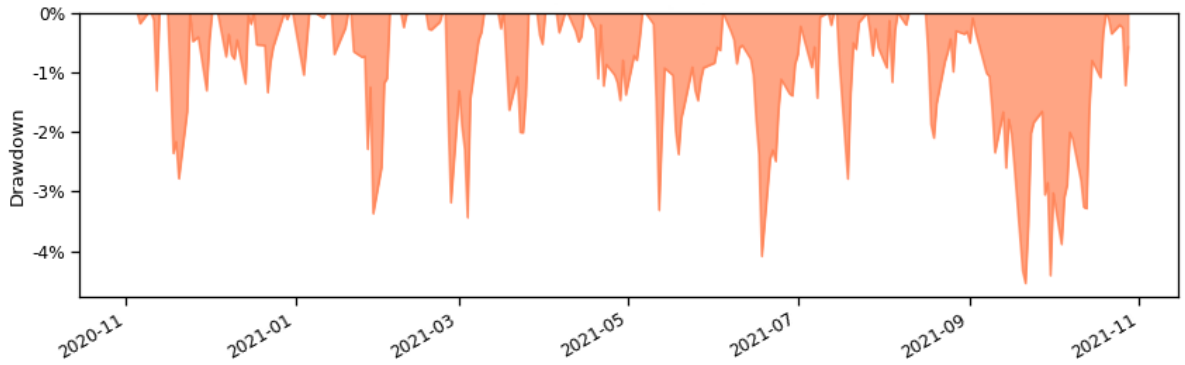


شکل ۵. عملکرد الگوریتم پیشنهادی در مقایسه با استراتژی‌های مرسوم و الگوریتم‌های یادگیری تقویتی عمیق

برای بررسی بیشتر کاربرد الگوریتم پیشنهادی، میزان افت سرمایه[۲۶] در طول دوره آزمون در شکل ۶ ترسیم شده است. مطابق آنچه در این شکل دیده می‌شود استراتژی حاصل از الگوریتم پیشنهادی بر روی داده‌های واقعی در بیشترین حالت، با افت سرمایه‌ای کمتر از ۵ درصد روبرو می‌شود که مقدار قابل قبولی در سرمایه‌گذاری است. بعلاوه این افت سرمایه نیز مطابق شکل در انتهای دوره سرمایه‌گذاری جبران شده است. افت و خیز سرمایه یکی از ویژگی‌های ذاتی در بازارهای مالی به ویژه مدیریت سبد سهام است و عامل تعیین‌کننده در یک سرمایه‌گذاری موفق استفاده از استراتژی است

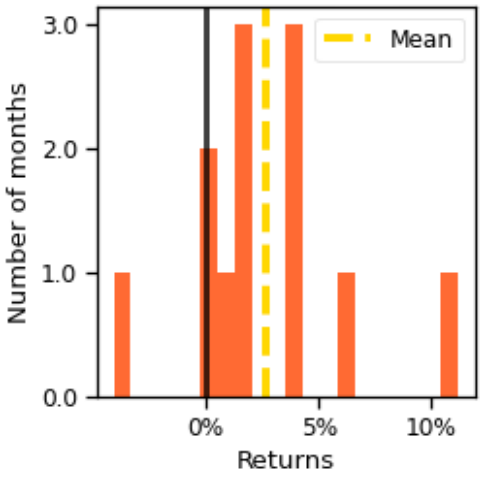
[26] Drawdown

که بتواند این نوسانات را در بازه معقولی حفظ نماید و عین حال به سود قابل قبولی برسد که با توجه به نتایج به دست آمده الگوریتم پیشنهادی به خوبی از عهده این کار برآمده است.



شکل ٦. افت سرمایه الگوریتم پیشنهادی در طول دوره آزمون

شکل ۷ توزیع ماهانه بازده سرمایه را با بکارگیری الگوریتم پیشنهادی نشان می‌دهد. مطابق آنچه در این شکل نمایش داده شده است، میانگین بازده سرمایه ماهانه بیش از ۲/۵ درصد است که نرخ بازده ماهانه خوبی در بازار سرمایه به حساب می‌آید. از سوی دیگر بازده سالانه سرمایه نیز به بالای ۳۵ درصد رسیده است (شکل ۵) که کاربردپذیری الگوریتم پیشنهادی را نشان می‌دهد.



شکل ۷. توزیع ماهانه بازده الگوریتم پیشنهادی در طول دوره آزمون

۶) نتیجه‌گیری

هدف از این پژوهش ارائه راهکاری برای بهینه‌سازی سبد سهام مبتنی بر یادگیری تقویتی بود. در این راستا به دنبال الگوریتمی برای خرید و فروش سهام بودیم که بتواند عامل هوشمندی را برای تصمیم‌گیری در بازار آموزش دهد. این عامل در تعامل با محیط یاد خواهد گرفت که چگونه در طول دوره‌های زمانی مشخص اقدام به خرید و فروش مجموعه‌ای از سهام‌های از پیش تعریف شده کند به نحوی که ارزش پایانی سبد سهام بیشینه شود. در ادامه به منظور

مدل‌سازی مساله از فرایند تصمیم‌گیری مارکف استفاده کردیم و در نهایت الگوریتمی بر مبنای شبکه عمیق Q برای خرید و فروش الگوریتمی سهام پیشنهاد شد.

با استفاده از یک مجموعه داده واقعی از سهام‌های معامله شده در بورس نیویورک و بورس نزدک به بررسی کارایی الگوریتم مورد اشاره پرداختیم. نتایج این بررسی‌ها نشان داد که ساختار شبکه پیشنهادی برای یادگیری تقویتی عمیق Q می‌تواند ضمن بیشینه‌سازی پاداش به کمترین میزان ریسک دست یابد و در مقایسه با استراتژی‌های مرسوم سرمایه‌گذاری نظیر شاخص داو جونز و روش کمینه واریانس عملکرد بهتری داشته باشد و بازدهی بیش از ۳۵ درصد به دست دهد. بنابراین الگوریتم یادگیری عمیق Q به عنوان روش پیشنهادی این پژوهش می‌تواند به تصمیم‌گیری سرمایه‌گذاران در بازارهای مالی کمک نماید.

برای پژوهش‌های آتی می‌توان با استفاده از شبکه‌های از پیش آموزش داده شده به دنبال بهبود ساختار و عملکرد شبکه‌های پیشنهادی در این پژوهش بود. استفاده از چنین شبکه‌هایی به یکی از روندهای جذاب در حوزه هوش مصنوعی تبدیل شده است و بسته به نوع مساله مورد مطالعه می‌تواند به بهبودهای چمشگیری در خروجی شبکه‌های پیشنهادی منجر شود. با این حال باید توجه کرد که نوعا این مدل‌های از پیش آموزش داده شده در مقایسه با شبکه‌های عمیق سنتی به منابع سخت‌افزاری بیشتری نیاز دارند که همین امر کاربردپذیری آن‌ها را در موارد واقعی کاهش می‌دهد. بعلاوه معمولا برای آموزش چنین مدل‌هایی به داده‌های زیادی نیاز است در حالی که داده‌های بازارهای مالی محدود است و برای حل این مشکل باید راهکاری اندیشید. در نهایت می‌توان در پژوهش آتی، عملکرد الگوریتم پیشنهادی را با سایر الگوریتم‌های یادگیری تقویتی عمیق موجود در پیشینه پژوهش مقایسه کرد و با بینش‌های حاصل از آن به دنبال بهبود روش پیشنهادی و یا ترکیب روش‌های مختلف برای رسیدن به استراتژی بهینه برای سرمایه‌گذاری در بازار سهام بود.

منابع

Alfonso-Sánchez, S., Solano, J., Correa-Bahnsen, A., Sendova, K. P., & Bravo, C. (2024). Optimizing credit limit adjustments under adversarial goals using reinforcement learning. European Journal of Operational Research, 315(2), 802-817. https://doi.org/10.1016/j.ejor.2023.12.025

Ang, A. (2012). Mean-variance investing. Columbia Business School Research Paper, no. 12/49. https://dx.doi.org/10.2139/ssrn.2131932

Aroussi, R. (2024). yfinance. Download market data from Yahoo! Finance's API. https://github.com/ranaroussi/yfinance

Chen, L., & Gao, Q. (2019). Application of deep reinforcement learning on automated stock trading. In Proceedings of the 10th International Conference on Software Engineering and Service Science (ICSESS), 29-33. https://doi.org/10.1109/ICSESS47205.2019.9040728

Du, X., Zhai, J., & Lv, K. (2009). Algorithm trading using q-learning and recurrent reinforcement learning. Stanford University, 1-7

Duerson, S., Khan, F., Kovalev, V., & Malik, A. H. (2005). Reinforcement learning in online stock trading systems. Georgia Institute of Technology.

Filos, A. (2018). Reinforcement learning for portfolio management. MEng Thesis. Imperial College London. https://doi.org/10.48550/arXiv.1909.09571

Gao, X., & Chan, L. (2000). An Algorithm for Trading and Portfolio Management Using Q-learning and Sharpe Ratio Maximization. In Proceedings of the 7th International Conference On Neural Information Processing (ICONIP 2000), 832-837.

Graesser, L., & Keng, W. L. (2019). Foundations of Deep Reinforcement Learning: Theory and Practice in Python. Addison-Wesley Professional.

Haarnoja, T., Zhou, A., Abbeel, P., & Levine, S. (2018). Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. arXiv, 1-14. https://doi.org/10.48550/arXiv.1801.01290

Jiang, Z., Xu, D., & Liang, J. (2017). A deep reinforcement learning framework for the financial portfolio management problem. arXiv, 1-31. https://doi.org/10.48550/arXiv.1706.10059

Jin, O., & El-Saawy, H. (2016). Portfolio management using reinforcement learning. Stanford University, 1-6.

Kabbani, T., & Duman, E. (2022). Deep Reinforcement Learning Approach for Trading Automation in the Stock Market. IEEE Access, 10, 93564-93574. https://doi.org/10.1109/ACCESS.2022.3203697

Kingma, D., & Ba, J. (2015). Adam: A Method for Stochastic Optimization. In Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015), 1-15. https://doi.org/10.48550/arXiv.1412.6980

Li, S. E. (2023). Reinforcement Learning for Sequential Decision and Optimal Control.Singapore: Springer Verlag. https://doi.org/10.1007/978-981-19-7784-8

Liang, Z., Chen, H., Zhu, J., Jiang, K., & Li, Y. (2018). Adversarial deep reinforcement learning in portfolio management. arXiv, 1-11. https://doi.org/10.48550/arXiv.1808.09940

Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., & Wierstra, D. (2019). Continuous control with deep reinforcement learning. arXiv, 1-14. https://doi.org/10.48550/arXiv.1509.02971

Malkiel, B. G. (2003). Passive investment strategies and efficient markets. European Financial Management, 9(1), 1-10. https://doi.org/10.1111/1468-036X.00205

Markowitz, H. M. (1991). Portfolio Selection: Efficient Diversification of Investments, 2nd Edition. New York: Wily.

Martin, R. A. (2021). PyPortfolioOpt: portfolio optimization in Python. Journal of Open Source Software, 6(61), 3066. https://doi.org/10.21105/joss.03066

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., & Riedmiller, M. (2013). Playing Atari with Deep Reinforcement Learning. arXiv, 1-9. https://doi.org/10.48550/arXiv.1312.5602

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., & Hassabis, D. (2015). Human-level control through deep reinforcement learning. Nature, 518, 529–533. https://doi.org/10.1038/nature14236

Ngo, V. M., Nguyen, H. H., & Van Nguyen, P. (2023). Does reinforcement learning outperform deep learning and traditional portfolio optimization models in frontier and developed financial markets?. Research in International Business and Finance, 65, 101936. https://doi.org/10.1016/j.ribaf.2023.101936

Poole, D. L., & Mackworth, A. K. (2023). Artificial Intelligence: Foundations of Computational Agents, 3rd edition. Cambridge: Cambridge University Press. https://doi.org/10.1017/9781009258227

Quantopian Inc. (2019). pyfolio. Portfolio and risk analytics in Python. https://github.com/quantopian/pyfolio

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal Policy Optimization Algorithms. arXiv, 1-12. https://doi.org/10.48550/arXiv.1707.06347

Sharpe, W. F. (1994). The Sharpe Ratio. The Journal of Portfolio Management, 21, 49-58. https://doi.org/10.3905/jpm.1994.409501

Soleymani, F., & Paquet, E. (2021). Deep graph convolutional reinforcement learning for financial portfolio management – DeepPocket. Expert Systems with Applications, 182, 115-127. https://doi.org/10.1016/j.eswa.2021.115127

Sutton, R. S., & Barto, A. G. (2018). Reinforcement Learning: An Introduction, Second edition. Cambridge: MIT press.

Tolouie Eshlaghy, A., & Haghdoust, S. (2007). Modelling of Prediction Stock Price by Using Neural Networks and Compare it with Mathematical Prediction Methods. Economics Research, 7(25), 237-251. (In Persian)

van Otterlo, M., & Wiering, M. (2012). Reinforcement Learning and Markov Decision Processes. In: Wiering, M., van Otterlo, M. (eds) Reinforcement Learning. Adaptation, Learning, and Optimization, vol 12. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-27645-3_1

Varga, B., Kulcsár, B., & Chehreghani, M. H. (2023). Deep Q-learning: A robust control approach. International Journal of Robust and Nonlinear Control, 33(1), 526-544. https://doi.org/10.1002/rnc.6457

Vergara, G., & Kristjanpoller, W. (2024). Deep reinforcement learning applied to statistical arbitrage investment strategy on cryptomarket. Applied Soft Computing, 153, 111255. https://doi.org/10.1016/j.asoc.2024.111255

Wang, Y., Wang, D., Zhang, S., Feng, Y., Li, S., & Zhou, Q. (2017). Deep Q-trading. Technical Report-20160036. Center for Speech and Language Technologies (CSLT), Tsinghua University, 1-9.

Wang, Z., Schaul, T., Hessel, M., van Hasselt, H., Lanctot, M., & de Freitas, N. (2016). Dueling Network Architectures for Deep Reinforcement Learning. arXiv, 1-15. https://doi.org/10.48550/arXiv.1511.06581

Xiong, Z., Liu, X.-Y., Zhong, S., Yang, H., & Walid, A. (2022). Practical deep reinforcement learning approach for stock trading. arXiv, 1-7. https://doi.org/10.48550/arXiv.1811.07522

Ye, Y., Pei, H., Wang, B., Chen, P.-Y., Zhu, Y., Xiao, J., & Li, B. (2020). Reinforcement-Learning Based Portfolio Management with Augmented Asset Movement Prediction States. In Proceedings of the 34th AAAI Conference on Artificial Intelligence, 34(01), 1112-1119. https://doi.org/10.1609/aaai.v34i01.5462

Zhang, Z., Zohren, S., & Roberts, S. (2020). Deep reinforcement learning for trading. The Journal of Financial Data Science, 2(2), 25-40. https://doi.org/10.3905/jfds.2020.1.030