Benchmarking Reinforcement Learning's Ability to Reach Markowitz's Efficient Frontier

May 12, 2023

Department of Economics University of California, Berkeley Undergraduate Honors Thesis

Avantika Ghosh Advised by Professor Stephen Bianchi

Abstract

This paper addresses the research gap in how reinforcement learning can be used for the 70% of everyday amateur traders in the United States, in an attempt to reach and even surpass an optimal portfolio formulated by the Markowitz Efficient Frontier. To do so, I created a Reinforcement Learning algorithm that outputs the reward and loss functions of each S&P 100 stock over the span of two years. In combination with the technical indicators that were implemented in the training function, my Reinforcement Learning algorithm outputs the total rewards (expected returns) and losses (decrease in profit from investment) that were experienced per stock, with which I sort into a list to determine the stocks with the highest weight to create an optimal portfolio based on highest reward and lowest loss. Overall, Reinforcement Learning as a feasible trading strategy that reaches and potentially even surpasses the returns of a portfolio produced by the Markowitz Efficient Frontier Curve shows promise as a useful trading strategy to amateur traders who cannot afford or do not use investment firms.

Introduction

Many past papers create benchmarks between automated algorithms, yet none are able to support a large training and testing data set while taking into account hyperparameters that are specific to everyday non-professional traders who do not use brokerage fees or other external variables. This paper will cater toward non-professional traders who have access to historical stock data, whether it is through open-source information or trading applications.

The most significant contribution this paper will provide is in the methodology. These include training models with a large training data set, using a model that puts an adequate weight on reward functions, using a well-rounded reward function, and comparing the supposed optimal portfolio that the RL algorithm outputs to what the Efficient Frontier suggests.

In order to avoid using a weak reward function in my reinforcement learning model, I plan to use the average profit or area under the trade, which should prioritize capitalizing on profit. Essentially, I plan to use the Accumulated Asset Value as a reward function, which incorporates the entire performance of an evolved trader since it produces a significantly better performing trader on volatile stocks (Nicholls et al 2008).

In order to ascertain the expected reward and loss behind a series of actions, reinforcement learning algorithms use a function approximator to estimate the previous action value function or expected return for taking an action in the current state of the portfolio, namely a Q-network. A function approximator is a method that is used to estimate the value of a state or an action, without computing the real “q value”, with the use of historical data, with the added benefit of saving computation time and memory space. Similar to the goals of a reward maximizing reinforcement learning algorithm, a Q-network is trained to minimize sequences of loss functions that change at each iteration (Van Hasselt 2015). While a loss function is typically

defined to measure how well an algorithm models the dataset, in this study, the loss function also minimizes the error of optimizing the weights of the portfolio. This indicates that the stocks with the highest weight in the optimal portfolio should also have the lowest loss output from the model's loss function.

In order to calculate the reward function, the model calculates the profit at each iteration based on whether the proposed optimal portfolio places a higher weight (buy an additional share), lower weight (sell an additional share), or same weight (do nothing) on the stock. The profit is calculated using the Accumulated Asset Value equation:

$$AAV = \frac{\sum_{i=1}^{N} [(Price_{t_s} - c_s) - (Price_{t_b} + c_b)]}{N}$$

In the equation above, "*i"* represents the action being taken on the stock in question in terms of adjusting the weight or number of shares acquired (buy, sell, do nothing). "$Price_{ts}$" represents the price that the stock in question was sold at. *"*$Price_{tb}$" represents the price that the stock in question was bought at. "c_s" represents the selling cost of the stock and "c_b" represents the purchase price of the stock.

Unlike other deep reinforcement learning in stock trading papers, my reward function only takes into account information that is available to the average amateur trader such as last price, selling and purchase cost, and historical averages. Other factors such as short selling and brokerage fees were taken out of the model since only investment firms deal with these factors (S. V. Stoyanov, et al 2007). Furthermore, while not described in this model, the statistical measure of various technical indicators -- including simple moving average, average true range, average directional index, stochastic oscillators, relative strength index (Panigrahi, 2022),

moving average convergence divergence, bollinger bands (Fang et al., 2017), and rate of change -- were used in the training model. For further discussion about the technical indicators, please visit the References section.

My reinforcement learning model minimizes the error estimated using this loss function by optimizing the weights of the portfolio, θ, using the difference between the predicted profit and actual profit based on historical data.

$$L(\theta) = \frac{(TargetQValue)}{((r + \gamma * max_{a_{t+1}} Q(s_{t+1}, a_{t+1}, \theta^{target})) - Q(s, a, \theta^{pred}))^2}$$

In the equation above, "*TargetQValue"* represents the predicted profit when the model takes a certain action on the stock in question (buying an additional share, selling an additional share, doing nothing to the quantity of shares in the portfolio). "*r"* represents the reward or expected return of the specified action on the stock in question. "γ" represents the discount factor, namely the amount that the expected return declines due to delay in action over time. "a" represents the action that is being taken on the particular stock (buying an additional share, selling a share, or doing nothing on the stock). "*s"* represents the current state of the portfolio given by the current weights of each stock in the portfolio. "θ^{target}" represents the actual weight of the stock in question and "θ^{pred}" represents the predicted weight of the stock in question. "*t"* represents the time in days over which the action is being taken, and "*Q"* represents the profit function of the stock based on the action.

By iterating through each stock in the portfolio and adjusting the optimal weighting using the reward and loss function defined above, my reinforcement learning algorithm moves toward an improved portfolio in an effort to increase the expected returns while minimizing losses in

profits. Using this technique, my results demonstrated a near one-to-one mapping between the highest weighted stocks in both the Reinforcement Learning algorithm and Efficient Frontier, which implies that Reinforcement Learning is a feasible trading strategy to reach and even surpass the Efficient Frontier.

For the remainder of this paper, I will be identifying how my methodology in this study addresses key concerns and flaws of previous research, providing background information necessary to understand the relevance of my study which will lead to the hypothesis of this paper. Finally, I will discuss in greater detail the design of the Reinforcement Learning algorithm, with specific attention to the reward function, followed by a summary of the results from the study and the implications that the results provide.

Review of Literature

Within the literature on trading algorithms in comparison to the Efficient Frontier curve (Lwin et al. 2017; Youmbi 2017), I contribute, to my knowledge, the first benchmark of using Reinforcement Learning as a feasible trading algorithm that produces an optimal portfolio that reaches the Efficient Frontier curve. In the paper detailing the Value-at-Risk model, the author's replace the variance in the Mean-Variance model with "an industry standard risk measure, conditional Value-at-Risk (VaR)" to better account for market risk exposure as a result of asset price fluctuation (Lwin et al. 2017). This paper contributed ideas to my methodology, in that as a further benchmark to test whether Reinforcement Learning was a feasible strategy to reach Markowitz's curve, I had planned to identify if my outputted optimal portfolio would conceive a higher mean and lower variance, but using conditional VaR proves to be a better measure.

This paper is also related to the literature on Reinforcement Learning algorithms (Li 2021) in that while alternative algorithms take into account measures of optimal return,

Reinforcement Learning algorithms are devised on the basis of hyperparameters (parameters initialized before the training process). This paper focuses on controlling for every possible market condition that active traders working on Wall Street are interested in. Compared to the goals of my study, the paper places too high of a weight on the certain parameters such as the brokerage cost ($\delta = 0.001$ compared to $\varrho = 0.1$, $\lambda = 0.01$), and a trading window of 100 days (which is too large considering the volatility of stocks). More than 70% of American traders are independent passive trader's who bimonthly push money into stocks on free apps like Fidelity, with most stock price fluctuations occurring every month. Placing a considerable weight on brokerage cost as a hyperparameter that is used for training the model is irrelevant and biases the model for investors that this study caters to. However, besides this potential flaw, the Reinforcement Learning model the paper presents creates a well-rounded foundation for my own algorithm.

While Deep Reinforcement Learning has been proposed for passive trading strategies, most present day experiments have demonstrated slightly flawed demonstrations in weighing hyperparameters, risk, and creating testing and training data sets (London et al., 2022). Unlike the other papers, this study discusses a combination of studies related to using deep RL for stock trading. Some of the most pressing concerns that were highlighted in these papers that I have accounted for in my methodology include testing only one stock and having a weak reward function which does not adequately measure risk, and fixing this using Sharpe's Ratio (Chen, Gao 2019). Furthermore, the paper points out how using "raw price data contains a lot of noise - technical analysis features should be used to mitigate that" and having too small of a training data set would be a problem for teaching the algorithm, which is why I planned to use two years worth of the most recent data available. However, it is important to note that even though the

S&P 100 consists of a diverse variety of stocks, due to current market conditions of hiring freezes, the results may show slightly anomalous results compared to other time intervals.

Background

Reinforcement Learning (RL) has a long history of serving as a tool for optimal decision making, from autonomous Game Theory simulations in Pacman to sports betting. In fact, there is extensive research conducted in using RL in creating stock trading algorithms for investment firms, such as Bloomberg and Goldman Sachs. However, there is a significant research gap in how reinforcement learning can be used for the everyday amateur trader using applications such as Fidelity or Robinhood, where brokerage fees and short selling are not applicable, in an attempt to reach and even surpass an optimal portfolio formulated by the Markowitz Efficient Frontier.

Reinforcement Learning's wide range of usage is mostly due to its ability to "use intelligent agents that are trained to take actions in an environment in order to maximize the cumulative reward or net benefit of taking those sets of actions"(Hammoudeh 2018). For example, if an agent had the option to flip a fair coin and earn a positive profit no matter the result, versus earning no profit from not flipping the fair coin at all, the rational agent, as well as the Reinforcement Learning algorithm, would always choose to flip the coin because they would be maximizing their reward from such an action. However, if the agent was presented with the idea that flipping a heads outcome would cause a positive profit, while flipping a tails outcome would cause a loss, the rational agent would be more hesitant to choose to flip the coin and would have to take other factors into account, such as expected return.

In order to better advocate for amateur traders who have limited open-source data available, this paper will focus on how Reinforcement Learning models that take into account technical indicators, can be used to reach the Markowitz Efficient Frontier. The Markowitz

Efficient Frontier plays a crucial role as a success metric as it is “the set of optimal portfolios that offer the highest expected return for a defined level of risk or the lowest risk for a given level of expected return”(Youmbi 2017). In this study, this model will serve as the theoretical success metric in determining the feasibility of Reinforcement Learning as a trading algorithm due to its ability to identify an optimal portfolio. However, it is important to note that the Efficient Frontier only serves as a theoretical point due to the unrealistic assumptions it makes, such as “assets follow a normal distribution, investors are rational and avoid risk when possible, that there is not enough investors to influence market prices, and that investors have unlimited access to borrowing and lending money at the risk-free interest rate”(Youmbi 2017). Therefore, I believe that the Reinforcement Learning model will surpass the expected returns of the Efficient Frontier Curve.

Hypothesis

The Markowitz Efficient Frontier theory predicts the form of the optimal portfolio of an investor, namely the highest expected return for a defined risk. Therefore, I will be studying the feasibility of a better portfolio in terms of mean return and volatility through the use of Reinforcement Learning (RL) techniques. RL algorithms that take into account technical indicators – including simple moving average, average true range, average directional index, stochastic oscillators, relative strength index, moving average convergence divergence, bollinger bands, and rate of change – can serve as a feasible passive trading strategy that reaches the Markowitz Efficient Frontier. In my study, the Efficient Frontier will serve as the theoretical success metric because past research into various other algorithms, including the Mean-Variance algorithm, prove to create sub-optimal portfolios to the Efficient Frontier.

Methodology

To test my hypothesis, I built a Reinforcement Learning forecasting tool to predict stock price movement using the technical indicators mentioned in my hypothesis. Specifically, I utilized web scraping to obtain the various company stock data during the past two years from October 29, 2020 to November 18, 2022 and inputted the Last Price, Highest Price, Lowest Price, and Volatility data columns into my model. My model outputs the reward and loss functions over time of each of the 101 stocks. With these predictions available to me, I selected the stock with the highest probability of return. With my stocks selected, I utilized the investment platform Think or Swim to input my stocks and output plots that show predicted movement over the span of a week. In order to test my hypothesis, I derived a script that simulates the Efficient Frontier theory using a Kaggle Notebook. Since this script is less than a year old and allows for inputting as large a portfolio as desired, I inputted the original company stock dataset into the Efficient Frontier notebook to see if the stocks that lie on the curve are the same stocks that the Reinforcement Learning algorithm identifies. The most optimized portfolio lies on the frontier curve, when mapping volatility over rate of return. This curve serves as the metric for success because any rates of return above the curve are theoretically impossible so if the stocks chosen from the Reinforcement Learning algorithm lie on this curve, we know that the portfolio is the best theoretical portfolio possible. However, it is important to note that the Efficient Frontier does make a few unrealistic assumptions as described above. Since these assumptions do not always translate to the real world, I used the Efficient Frontier curve as a theoretical success metric, but ultimately further investigated if Reinforcement Learning can surpass the returns of this Efficient Frontier curve due to the theoretical possibility.

Dataset

This study uses S&P 100 company data ranging over the span of two years (10/29/2020 - 11/18/2022) from Barchart, an open source website and leading provider of real time or delayed intraday stock and commodities charts and quotes. Since this data updates daily, I use two years worth of stock data and normalize the data such that every stock starts at a relatively equivalent starting price (100 dollars), which will make it easier to check to ensure that the basic predicting functionality of the Reinforcement Learning algorithm is functional. The downloaded data is a CSV file that includes 101 different observations, namely each company in this index. The fact that we have one observation per company daily makes it necessary to collect data everyday for normalization and to ensure a large enough dataset for this study. Each observation (company) includes the last trade price (Last), the difference between the current price and the previous day's settlement price (Change), the associated price change percentage (%Chg), the highest trade price for the day (High), the lowest trade price for the day (Low), and the total number of shares or contracts traded that day for the stock (Volume), and date (Time).

Past research has found issues with having a large enough training data set, especially during time periods that are not representative to general market trends. In the paper about Deep Learning, the study uses the time period from 2006 to 2018, which inherently contains data produced during the 2008 financial crisis. This was largely an anomaly for the algorithm to learn from. This paper will be using S&P 100 data from the website Barchart. The S&P 100 Index is a "stock market index of United States stocks maintained by Standard & Poor's. [It is] a subset of the S&P 500 [which] is designed to measure the performance of large-cap companies and comprises 100 major blue chip companies across multiple industry groups"(S&P 100 Dow Jones Indices). Since this data is open-source and contains the fields "volatility" and "rate of return", it

will be a much cleaner process to isolate these fields every week and input the cleaned data into my Reinforcement Learning model and Efficient Frontier for comparison. In order to reach as well-rounded results as possible, I use historical data from the past two years (2020-2022) as a comparison. While the current market is experiencing a hiring freeze in the technology industry, the S&P 100 includes a variety of company stocks in many different industries, making the market conditions more normalized.

According to the Markowitz Efficient Frontier Theory, the theory maps the optimal portfolio curve with Volatility vs Expected Return. In order to use this dataset as an input into the Efficient Frontier algorithm, I only use the "Last" (last trade price) column since I need to calculate the rate of return and volatility in my algorithm. For rate of return, this would be the dot product of the last closing price and hypothetical weight for the stock. For volatility, this would be the square root of the dot product of the weight and dot product of weight with covariance of the return. An additional calculation that the Theory does is a Sharpe's ratio calculation which takes the ratio of the rate of return and volatility. Since it is the purpose of the Efficient Frontier, the weight for each stock in the optimal portfolio will be determined by these technical indicators, with the constraints that the summation of the weights need to be equal to one so each weight for a stock in the portfolio is between [0, 1]. Therefore, the only column needed from the dataset to create this study's success metric is the last closing price.

In order to use my dataset in my Reinforcement Learning model to predict the optimal portfolio, I used slightly more columns than what is needed in the Markowitz Efficient Frontier. The three main components of a Reinforcement Learning model are the current state, the action, and the reward function. In regards to the dataset, this would equate to the last closing price of the stock (current state), the weight of each stock in our optimal portfolio (action), and the

learning parameters and technical indicators (reward function). This dataset has the last closing price as a column given and provides vital volatility information such as the percent change in list price from the previous day to the current day for each stock as well as projected returns. Furthermore, the dataset includes the volume of trades and contracts made with each stock on a daily basis, which provides insight on how prices will change as mass selling or buying of stocks is a key indicator of where the price will be in the future. Using these fields was vital for designing a reward function that takes into account volatility/risk and expected return.

Design Overview

I. Efficient Frontier Algorithm Design

As mentioned above, I collected data from the last two years from Barchart and extracted the "Last" (last trade price) column for each company and input this data into the algorithm for the Efficient Frontier Theory. In order to map out the Efficient Frontier Theory, we must note that the price of the stocks will have varying movement, so I normalized the price points to create the same starting point for each stock. Furthermore, the key to portfolio diversification is combining stocks with low covariance in order to mitigate risk in the portfolio, so covariance was calculated in this algorithm.

As discussed above, the two input measurements for finding the Efficient Frontier is rate of return and volatility (Sharpe's Ratio is an additional technical indicator):

1. $Annualized\ Rate\ of\ Return\ =\ Daily\ Avg\ Return\ *\ 252\ (number\ of\ trading\ days)$
2. $Volatility\ =\ \sqrt{dot\ product(w,\ dot\ product(w,\ Cov(Daily\ Avg\ Return)))}$
3. $Sharpe's\ Ratio\ =\ Return/Volatility$ (Evaluates the return of an investment compared to its risk)

Once the rate of return, sharpe's ratio, and volatility were found for each stock, the next step was creating the optimization that calculates the highest return for the lowest level risk. To accomplish this, I used the Trust-Region Constrained Algorithm, which assigns initial weights for each stock assigned "θ_0", which is the first guess at the values of each stock's weight. In order to continue moving up the return axis by a fixed point amount, I calculated the volatility as an update function, which updated the weights that each stock has on the portfolio and outputs the optimized portfolio.

II. Reinforcement Learning Reward Function Design

The important part of what distinguishes a successful Reinforcement Learning algorithm is the reward function, the component that many past papers have found flaws within their own design. As I previously mentioned, Barchart provides all of the current necessary variables needed to design the reward function for my model, namely the "Last" closing prices, volatility, "%Chg", and expected returns. These parameters were used as a functionality test to make sure that my Reinforcement Learning model was working correctly. Specifically, ensuring that higher return and lower volatility (risk) stocks are given more weight in general.

The main idea behind a functioning reward function is to give a high reward for when the algorithm outputs a weight greater than 0.5 to stocks that have a higher expected return and lower volatility, while outputting a weight greater than 0.5 for the opposite characteristics in a stock would cause a penalty. Since I have the data to compare and evaluate these characteristics, the basic premise of my reward function was achievable. However, a potential issue was when creating more complex rules for the reward function to assign reward or penalty, namely additional technical indicators. As calculated above, I considered using the Sharpe's Ratio as a rule in the reward function to measure risk efficiency, since it is an industry standard. However,

the Mean-Variance technique that Sharpe's Ratio relies on proves that the ratio is heavily influenced by investments that don't have normal distributions of returns, which we cannot necessarily guarantee. Furthermore, I considered using the "Volumes" column from the data as well as historical data to create a predictive measure of how mass selling and buying of stocks may influence closing prices. However, to create such a measure would require honing in on a specific time period where certain stocks experienced drastic market changes, which skewed the overall results, proving to not be beneficial in this study.

Once the optimal portfolio is outputted from my Reinforcement Learning model, I extracted the expected return and volatility for each stock in the portfolio to create the same benchmark metric as what the Efficient Frontier curve will show.

Since both my Reinforcement Learning model and Efficient Frontier curve algorithm will output plots measuring expected return over volatility, both of which are calculated above, I observed how close the different plots are to each other. More importantly, the success metric will be defined by comparing which stocks are reported to have the highest weight in both models and their associated expected returns and volatility. If there is a significant discrepancy between which stocks have a higher weight in the portfolio, I can infer that Reinforcement Learning may not be a feasible trading strategy compared to the Markowitz Efficient Frontier. However, as mentioned above, Mean-Variance can be proven to be a sub-optimal method .

The main focus of this study is to prove that Reinforcement Learning can serve as a feasible trading strategy to reach the Markowitz Efficient Frontier. Therefore, besides the interpretation metrics discussed in the above paragraph, I focused on creating a reward function in my Reinforcement Learning that takes into account only relevant technical indicators for everyday amateur traders.

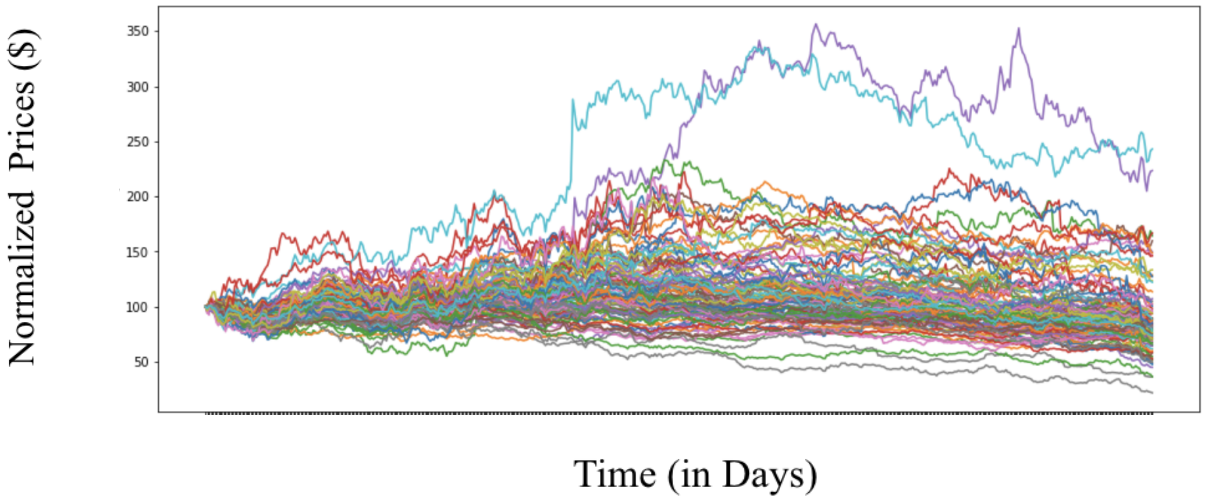
Results

Once the plots of each stock's total rewards and total losses over time are outputted, I sorted the stocks by highest returns and lowest loss to determine which stocks have the most weight in the optimal portfolio. In order to compare my results to the success metric, I ran the same two years worth of S&P 100 stock data through an Efficient Frontier algorithm to not only make sure that an optimal portfolio exists, but to identify which stocks are recommended to have the most weight in an optimal portfolio that has (i) the lowest risk associated and (ii) the lowest Sharpe's Ratio risk.

I. Figures and Plots of the Study

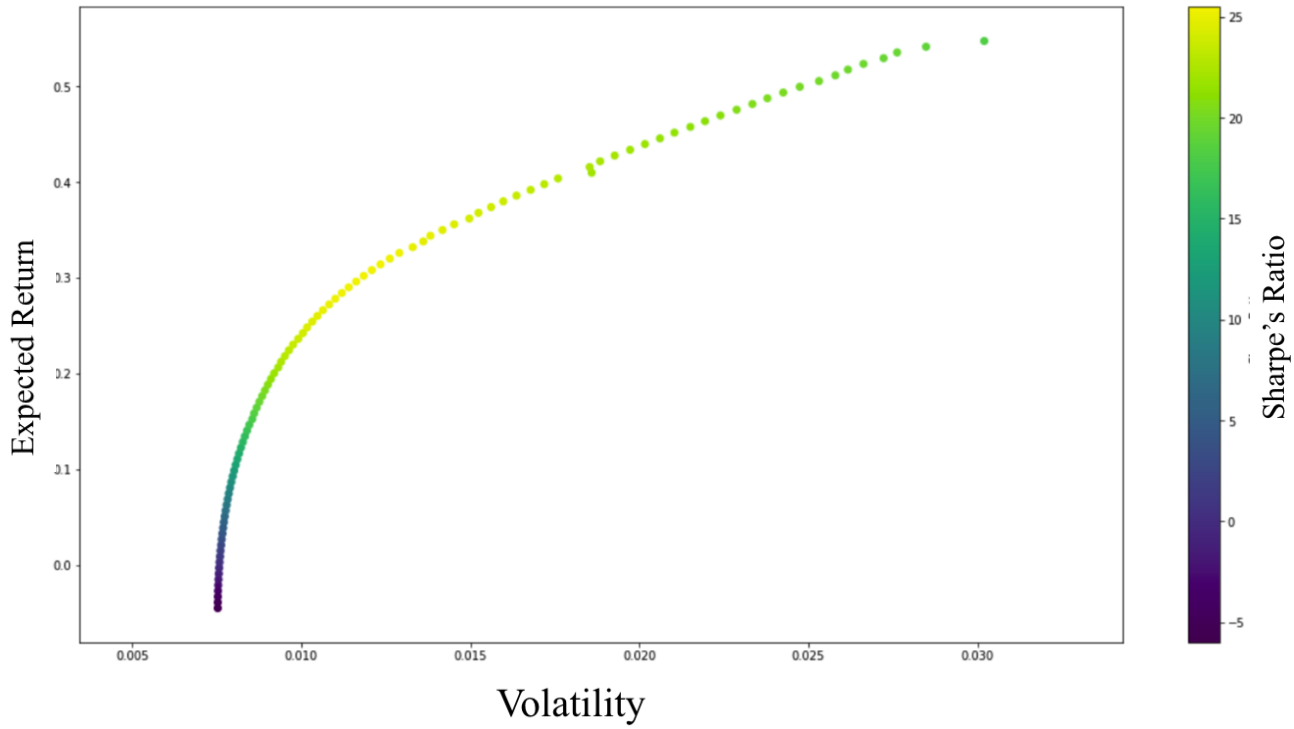
In the plots below, Figure 1 identifies the normalized prices of each stock in the S&P 100 dataset which serves as a cleaned starting point for inputting into the Reinforcement Learning Algorithm and the Efficient Frontier Curve theory. Figure 2 proves that the S&P 100 stocks does offer an optimal portfolio given the right weight on stock investment, and also provides a basis for comparison to identify which stocks should be given a higher weight in the optimal investment portfolio. Figure 3 includes the plots of the stocks that have the highest reward and lowest losses, which also correspond to stocks with the highest suggested weights from the Efficient Frontier Curve. Essentially, when inputting the S&P 100 dataset into the Efficient Frontier Curve, these three stocks were among the suggested highest weights and were identified by the Reinforcement Learning algorithm as having the highest rewards with lowest losses.

Figure 1: Normalized Prices of All S&P 100 Stocks



Notes: Since the prices of all 101 stocks are very different, it is difficult to compare without a normalized starting point for all the stocks, namely at 100 dollars. With normalization, it is easier to see that two of the stocks, represented by the blue and purple lines, increase much more compared to the others.

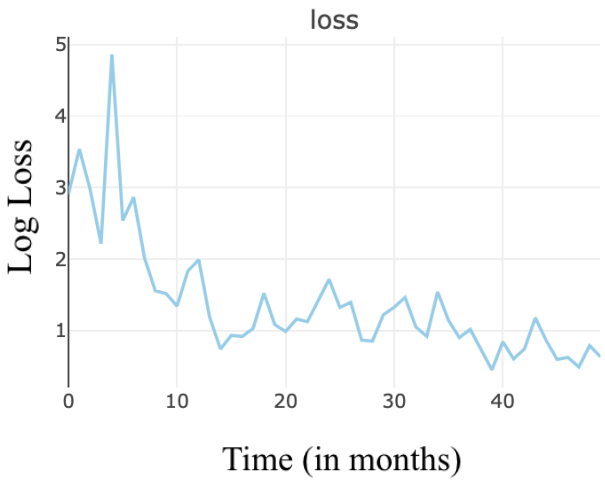
Figure 2: Efficient Frontier Curve of Optimal Stock Portfolio in S&P 100



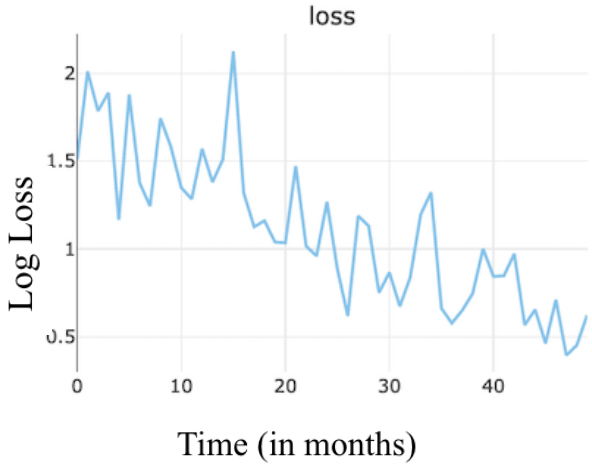
Notes: The figure above demonstrates that given the same dataset inputted into my Reinforcement Learning model, there exists an optimal portfolio that lies on the Efficient Frontier curve. Later we will see that the highest weighted stocks in this Efficient optimal portfolio have a close to one-to-one mapping with the highest weighted stocks in the Reinforcement Learning optimal portfolio.

Figure 3: Largest Assigned Weights from RL Algorithm that Correspond to the Stocks with Weights On The Higher End of the Efficient Frontier

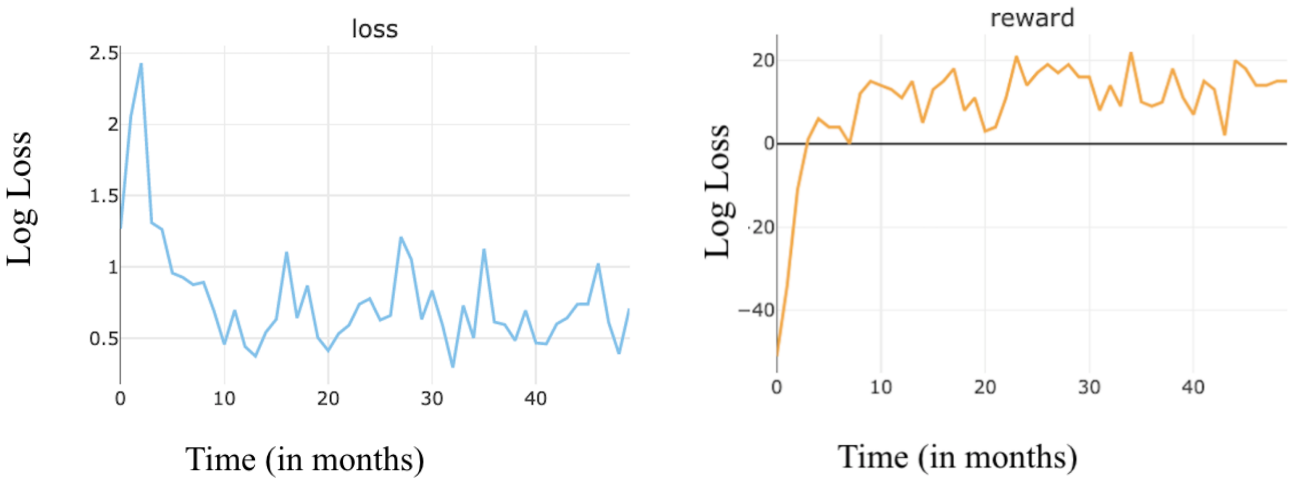
Plot A: Loss and Reward Function of Highest Weighted Stock in RL Algorithm and Markowitz Curve (Johnson & Johnson (JNJ))



Plot B: Loss and Reward Function of Second Highest Weighted Stock in RL Algorithm and Markowitz Curve (Comcast Corporation (CMCSA))



Plot C: Loss and Reward Function of Second Highest Weighted Stock in RL Algorithm and Markowitz Curve (Verizon Communications Inc (VZ))



Notes: The above graphs show the reward and loss functions of the highest weighted stocks in the optimal Portfolio outputted by the Reinforcement Learning algorithm, namely stocks with the highest expected returns and lowest risk. The reinforcement learning model shows that the highest expected return portfolio should have Johnson & Johnson, Comcast Corporation, and Verizon Communications Inc as the stocks with the highest weight in an optimal portfolio.

II. Interpretation

The perfect Reinforcement Learning algorithm would have shown that the S&P 100 stocks with the highest total reward and lowest total loss also correspond to the suggested highest weights in the Efficient Frontier Curve optimal portfolio. While there was not a direct correspondence in results between the algorithm and the success metric, I did find that the stocks that were suggested to have the highest weight in the Efficient Frontier Curve were among the "top stocks" in my sorted list for my Reinforcement Learning algorithm. Out of 101 stocks, the "top stocks" list consisted of seven stocks as a cutoff measurement due to the significant gap in highest reward to lowest loss between the seventh and eighth stock in the sorted list.

In terms of accuracy and success of this study in proving that Reinforcement Learning can be a feasible trading strategy that reaches the Markowitz Efficient Frontier Curve, the

Efficient Frontier Curve outputted the suggested weights of all 101 stocks in an optimal portfolio, 96 of which were weighted very low at about 0.01. The stocks that had the highest weight in the Reinforcement Learning optimal portfolio were weighted at 0.11 (JNJ), 0.15 (CMCSA), 0.18 (VZ), and 0.2, three of which correspond to the "top stocks" in the Reinforcement Learning algorithm. Therefore, due to this high precision result, we can conclude that Reinforcement Learning can be a feasible trading strategy for this problem. However, we can also conclude that the Reinforcement Learning algorithm that was used in this study can be improved for even higher accuracy. As mentioned above, an ideal Reinforcement Learning algorithm would have the same stocks listed as "top stocks" and have the highest weights in the Efficient Frontier curve optimal portfolio.

Contextually speaking, the "top stocks", otherwise known as the stocks that had the highest expected return and lowest volatility for both the Reinforcement Learning algorithm and Efficient Frontier do provide connection to real world circumstances. Due to the COVID-19 pandemic, the medical device company Johnson & Johnson was one of the companies tasked with finding a vaccination to COVID-19. This caused outside investment to the company, as well as the company's valuation to increase, which may contribute to why the Reinforcement Learning algorithm predicted higher returns with this stock. Similar circumstances may have caused both Verizon and Comcast to increase in valuation, namely that the pandemic induced many families to become more involved in the internet as a form of communication and entertainment. However, without an extremely controlled experiment, it is impossible to conclude these contextual assumptions as to why these three stocks were predicted to provide high returns.

Conclusion

Seventy percent of traders in America are everyday passive traders who use applications like Think or Swim in order to make educated decisions on what stocks they would like to invest in to increase the return of their portfolios. Most investment platforms use historical data in order to map trends to give valuable insights to investors. However, as seen in this study, when compared to the Markowitz Efficient Frontier, Reinforcement Learning provides another layer of rational decision making that takes into account multiple technical indicators that can further help investors make educated investment decisions, beyond historical averages. Using Reinforcement Learning to predict an optimal portfolio outputted stocks (Johnson & Johnson, Verizon Communications, and Comcast) that current investment platforms such as Think or Swim and Robinhood do not show as the "best stocks to invest in", despite the expected returns surpassing the Efficient Frontier returns. Yet, when checking the validity of these stocks using technical indicators defined by Asset Pricing Theory -- simple moving average, average true range, average directional index, stochastic oscillators, relative strength index, moving average convergence divergence, bollinger bands, and rate of change -- these stocks show empirical promise in providing higher expected returns. Therefore, Reinforcement Learning as a feasible trading strategy that reaches and potentially even surpasses the returns of a portfolio produced by the Efficient Markowitz Curve shows promise as a useful trading strategy to the 70% of Americans who cannot afford or do not use investment firms.

I. Possible Improvements for Continuation

Since my preliminary results did not yield a one-to-one matching between the Reinforcement Learning algorithm and the Efficient Frontier theory, I plan to map out the expected return over volatility of all of the assets to identify if this form of measurement is more

accurate. I did not use this methodology beforehand because while this would have yielded a more direct comparison between the Reinforcement Learning algorithm and the success metric, plotting the total rewards and total losses based on a learning function that already rewards higher expected returns or profits essentially equates to the same thing. However, since this current study focuses more on profit over time than the actual expected return, there may be a third party factor that I am not taking into account which could contribute to the slight discrepancy in the results.

Furthermore, as another continuation of this study, I will be grabbing data from 2011 and running the methodology on this new dataset. While it is always best to use the most recent data available to account for current market conditions, the last two years have proven to be an anomaly in terms of the financial markets. Therefore, to ensure a more robust conclusion for this study, I will be taking a more normalized time interval where the financial market was not as heavily impacted from externalities.

Finally, while this current dataset did not include the relevant information for me to calculate the Value-at-Risk measure, which has proven to be more accurate for volatility (Visaltanachoti 2014), I will continue this study by using this measurement in the future in order to better evaluate the risk of investing in certain stocks in a portfolio.

Continuation Study

In subsequent iterations of this study, I decided to investigate the benchmarking results of the Reinforcement Learning model compared to the Efficient Frontier by making three main changes to the preliminary methodology. I created an Out of Sample iteration, revised the risk measure in both models to use Conditional Value at Risk instead of Volatility, and reran an

iteration of the study with a Convex Optimizer modification of the Reinforcement Learning model.

I. Out of Sample Study

To validate the effectiveness of the Reinforcement Learning model, I collected data from outside of the preliminary study's date range, namely January, February, and March of 2011, to conduct an Out of Sample Study. Since my initial dataset had included the pandemic era where there were abnormal economic booms in the technology sector, while the service industry suffered, it was important to expand the time horizon of the study with a more representative dataset of normal economic conditions. The Out of Sample Study demonstrated similar effectiveness as the preliminary study in that the stocks that were suggested to have a higher weight in an optimal portfolio between both Reinforcement Learning model and Efficient Frontier showed a high correlation. Therefore, I can infer that the Reinforcement Learning model is effective, regardless of time interval.

II. Conditional Value at Risk as a Risk Measurement

In the preliminary iterations of this study, I used S&P 100 company stock data over October 29, 2020 to November 18, 2022 as a dataset for benchmarking Reinforcement Learning's ability to forecast a portfolio that would surpass the expected return of the Efficient Frontier, given the same dataset. In the previous studies, the main measure for the risk of a stock was the volatility of the stock. However, risk measurements such as volatility and Value-at-Risk (VaR) are flawed in that VaR dismisses the lowest percentile of loss (or worst-case loss) associated with a probability and a time horizon, which does not represent risk accurately. As a result, I reran the study by revising the Reinforcement Learning model's risk measurement to use conditional value at risk. Conditional Value at Risk quantifies the average loss over a specified

time period of unlikely scenarios beyond the confidence level. In order to calculate the Conditional Value at Risk, I computed the daily returns of each stock using the Adjusted Close percent change, and subsequently sorted the returns. Using the "quantile" Python method, I was able to find the Value at Risks at 90%, 95%, and 99% respectively with the daily returns. In order to take into account the lowest percentiles of loss, as is the case in the Conditional Value at Risk, I indexed into where the closing price of each stock was less than the respective Value at Risk for each percentile and calculated the mean at each quantile.

Implementing the Conditional Value at Risk measure into the Reinforcement Learning Model required deviating from the typical implementation techniques that past research has shown due to the less strict constraints that come with amateur versus investment firm trading. Since the current loss system of the Reinforcement Learning model takes penalties when the average return of a portfolio goes down due to either selling or buying a share of a company, I calculated the average CVaR of the whole dataset and assigned a proportional loss penalty depending on where each company's individual CVaR stood relative to the average CVaR. This proved to be a much cleaner solution to seamlessly implementing Conditional Value at Risk without creating a completely different measurement metric that translated to the model context.

The results from this revision demonstrated that between October 29, 2020 to November 18, 2022, the stocks that the Reinforcement Learning model suggested to have the highest weight in an optimal portfolio were Altria Group Inc, NVIDIA, Disney, Oracle, and Intel, all of which were suggested to hold 0.2 weight out of 1. When compared to the Efficient Frontier Model, the "top stocks" of both models were the same, suggesting that during the specific time interval, Reinforcement Learning is a competitive model for suggesting an optimal portfolio to Efficient Frontier, with the risk measurement modification. In order to infer a more robust conclusion, the

same study was performed on the Out of Sample Dataset (January, February, March 2011) with the same competitive results.

III. Convex Optimizer

Convex optimization is a recursive game between a learner and opponent where at each iteration t, the learner first presents a solution that is defined in a set K, otherwise known as the solution space. The learner subsequently receives a convex function and suffers a loss based on their solution that they had presented earlier. The goal of the learner is to engineer a sequence of solutions that minimize the loss.

Since this study focuses on how Reinforcement Learning is a feasible strategy to surpass the Efficient Frontier, convex optimization was only used to find the ideal model parameters that minimize the loss function as defined in the Introduction. Acknowledging standard practice, the loss function was modified to serve as a convex optimization problem with linear constraints, along with a convex objective function, specifically gradient descent. Gradient descent is a convex optimization technique used to update the parameters in the direction of the negative gradient of the objective function. The update is determined by the size of each iteration's step, namely the learning rate (Sarykalin, Sergey, et al, 2008).

The convex optimization loss function used in this iteration of the study is defined below.

$$L_i(\theta_i) = E_{(s,\, a,\, r,\, s')}[(r + \gamma * max_{a'} Q(s', a', \theta_i^-) - Q(s, a; \theta_i))^2]$$

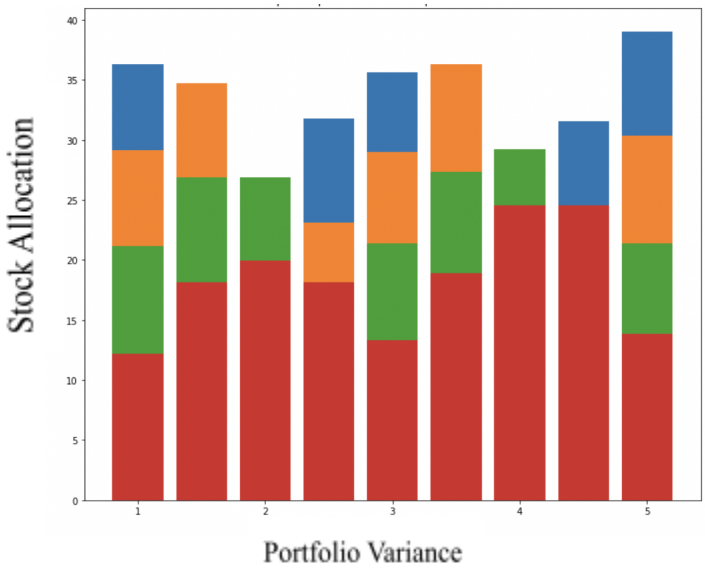
The gradient descent is defined below.

$$\nabla_{\theta i} L_i(\theta_i) = E_{s,\, a,\, r,\, s'}[(r + \gamma * max_{a'} Q(s', a', \theta_i^-) - Q(s, a; \theta_i)) \nabla_{\theta i} Q(s, a; \theta_i)]$$

Using this loss function that updates itself based on the direction of the negative gradient in order to reduce loss as efficiently as possible, I reran the same methodology from the

preliminary study on the 2020 to 2022 dataset, with this revision, and found that the optimal portfolio only recommended two top stocks since the other 98 companies were given trivial weights. When running the same iteration with the other time interval datasets presented in this paper, namely month to month S&P 100 companies in 2011, the revised loss function performed similarly to the iteration that used Expected Shortfall Risk. However, there was a slight improvement of 0.68% observed in the convex optimization loss function when comparing previous expected return to the expected return in this iteration. Therefore, we can infer that the differences in results can likely be attributed to events that occurred during the time intervals of the datasets that caused fluctuation in the stock price. While this revised study did incentivize maximizing expected portfolio return for a given portfolio variance based on historical data, a more optimal portfolio would have additional diversification beyond two stocks, which implies that expected return may not be the best objective function to use in a convex optimizer.

Figure 4: Expected portfolio return vs. portfolio variance (2D evaluation of 3D Graph)



Notes: The above graph shows the Expected Portfolio Return vs Portfolio Variance of the Highest Weight Stocks From the Convex Optimizer, including Stock Allocation.

IV. Conclusion

When benchmarking which revision to the preliminary Reinforcement Learning study outputs the most robust, optimal portfolio with the highest expected returns, using Expected Shortfall Risk instead of a volatility as a risk measure is presented as the best solution. Not only did the iterations of the study that used this risk measure outperform the Efficient Frontier, but also the iterations of the study where the loss function was a convex optimizer. Initial research about the expected shortfall risk and convex optimization hypothesized that the most robust optimal portfolio finder would be via convex optimization due to the minimized number of steps required to reach the optimal solution. This is likely due to the time intervals of the datasets used for this study, as there were significant economic events that occurred during the pandemic compared to 2011. Furthermore, due to the low diversity portfolio the convex optimizer outputs, it is possible that the objective function can be improved to attain a stricter goal, due to its previously relaxed state. However, Expected Shortfall Risk is a better measure for risk when using Reinforcement Learning to surpass the Efficient Frontier.

References

"Automation | Bloomberg Professional Services." 2022. *Bloomberg.com*. Bloomberg. Accessed December 11, 2022. https://www.bloomberg.com/professional/automation/.

Chen, Lin, and Q Gao. 2019. "Application of Deep Reinforcement Learning on Automated Stock Trading." *IEEE*. IEEE. October 1, 2019. https://ieeexplore.ieee.org/document/9040728.

Fang, Jiali, Ben Jacobsen, and Yafeng Qin. 2017. "Popularity versus Profitability: Evidence from Bollinger Bands." *The Journal of Portfolio Management*. Institutional Investor Journals Umbrella. July 31, 2017. https://jpm.pm-research.com/content/43/4/152.

Hammoudeh, Ahmad. 2018. "A Concise Introduction to Reinforcement Learning - Researchgate." Research Gate. February, 2018. https://www.researchgate.net/publication/323178749_A_Concise_Introduction_to_Reinforcement_Learning.

Lasdon, Leon S., Richard L. Fox, and Margery W. Ratner. "Nonlinear Optimization Using the Generalized Reduced Gradient Method." French Review of Automatic Control, Computer Science, Operational Research, January 1, 1974. http://www.numdam.org/item/?id=RO_1974__8_3_73_0.

Li, Lin. 2021. "An Automated Portfolio Trading System with Feature Preprocessing and Recurrent Reinforcement Learning." October 30, 2021. https://arxiv.org/pdf/2110.05299v2.pdf.

London, Chunli Liu King's College, Chunli Liu, King's College London, Carmine Ventre King's College London, Carmine Ventre, Maria Polukarov King's College London, Maria Polukarov, et al. 2022. “Synthetic Data Augmentation for Deep Reinforcement Learning in Financial Trading: Proceedings of the Third ACM International Conference on AI in Finance.” *ACM Other Conferences*. ACM Digital Libraries. November 1, 2022. https://dl.acm.org/doi/abs/10.1145/3533271.3561704.

Lwin, Khin T, Rong Qu, and Bart L MacCarthy. 2017. “Mean-VAR Portfolio Optimization: A Nonparametric Approach - Nottingham.” University of Nottingham. February 3, 2017. http://www.cs.nott.ac.uk/~pszrq/files/EJOR17.pdf.

Nicholls, J. F., A. P. Engelbrecht, and K. M. Malan. 2008. “Evaluation of Fitness Functions for Evolved Stock Market.” University of Pretoria, Pretoria, 0002, South Africa. October 30, 2008. http://aiecon.org/conference/2008/CIEF/Evaluation%20of%20Fitness%20Functions%20for%20Evolved%20Stock%20Market%20Forecasting/s98028571.pdf.

Panigrahi, CMA(Dr.) Ashok. 2022. “Trend Identification with the Relative Strength Index (RSI) Technical Indicator – a Conceptual Study.” *SSRN*. Social Science Research Network. February 11, 2022. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3986000.

S. V. Stoyanov & S. T. Rachev & F. J. Fabozzi, 2007. "Optimal Financial Portfolios," Applied Mathematical Finance, Taylor & Francis Journals, vol. 14(5), pages 401-436.

Sarykalin, Sergey, et al. *Value-at-Riskvs.conditionalvalue-at-Riskin*

Riskmanagementandoptimization. Tutorials Operational Research, 2008, https://www.ise.ufl.edu/uryasev/files/2011/11/VaR_vs_CVaR_INFORMS.pdf.

"S&P 100." 2022. *S&P Dow Jones Indices*. Accessed December 11. https://www.spglobal.com/spdji/en/indices/equity/sp-100/#overview.

Van Hasselt, Hado, et al, 2015. "Deep Reinforcement Learning with Double Q-Learning." *NASA/ADS*, September 22, 2015. https://ui.adsabs.harvard.edu/abs/2015arXiv150906461V/.

Visaltanachoti, Nuttawat. 2014. "Trading Strategy Performance When Using Value at Risk or Expected Shortfall as a Risk Constraint." *SSRN Electronic Journal*. Social Science Research Network. April 22, 2014. https://www.academia.edu/626211/Trading_Strategy_Performance_When_Using_Value_at_Risk_or_Expected_Shortfall_As_a_Risk_Constraint.

Youmbi, Didier. 2017. "Building the Markowitz Efficient Frontier." *SSRN*. Social Science Research Network. August 9, 2017. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3016043.