

Algo-Trading using Statistical Learning and Optimizing Sharpe Ratio and Drawdown



Penumatcha Bharath varma, Neeraj Kasheety, Hanumanula Sravya, Chinthapalli Amarnath Reddy, Jaypal Medida

Abstract: Modernization in computers and Machine Learning have created new opportunities for improving the methods involved in trading, Changes have been noticed parallelly at the level of investment decisions, and at the faster executions of trades via algorithms. Nowadays 90% of the trades are placed by algorithms, to execute a transaction, algorithms that follow a trend and construct a set of instructions are used in algorithmic trading. It executes the trades more precisely by precluding the effect of human feelings on trading. It all started way back in the 20th century and nowadays it's becoming more and more competitive, with more big players entering the market every day. Our research aims to advance the market revolution by developing an Algorithmic Trading approach that will automatically trade user strategies alongside its own algorithms for intraday trading based on different market conditions and user approach, and throughout the day invest and trade with continuous modifications to ensure the best returns for day traders and investors.

Keywords: Algorithmic Trading, high-frequency trading, Machine learning, Statistical Learning.

I. INTRODUCTION

Black-box trading is also known as “Algorithmic trading”, is a mechanism of performing traditional trades without any human interposition, it is basically a digital version of trading strategy that could be based on the event, price, patterns, news, quantity, mathematical model, etc. Although it is becoming very competitive day-by-day, earlier only actuary’s and institutional traders have access to the data and they took advantage and increased the Sharpe ratio and annual returns, Studies shown a new algo-trading trend emerged out of the blue, Fin-Tech enabled mobile apps have been deeply penetrated into the Trading market with the help of cheaper data plans provided by big Indian telecom giants

like jio , airtel, etc, hence resulted in the rise of retail trades. However, these fin-tech-enabled trading applications got approved way back in 2010 by the Securities and Securities and Exchange Board of India (SEBI) but it wasn’t as prevalent as it is today. Few Discount Brokerage platforms use artificial intelligence to provide an investor with curated portfolios. These platforms also have educational content which allows people to easily learn about investing methods and trading strategies/methods. Example: (Zerodha’s Kite Connect) is the platform where we can create, test, and execute trades using Kite Connect API. AI rule-based investment engines can analyze more than a billion data points instantly before extending a recommendation to its users, to give you intuition if a human wants to study a stock he/she might take one data point per second, it takes more than 31 years to reach their level of precision as systems. It helps users to make trades based on pre-set rules and gain maximum returns without any human emotion bias, these algos offer new opportunities for creating wealth and pave the way for systematic trading. In near future, 100% (percent) of all trades executed in the stock exchange will be done by algo’s. Machine learning helped programmers, financial experts, and developers to create many new strategies to make money from the market. Algo-trading practices are considered profitable, but it has their own drawbacks. Anyone who doesn’t know how to write a code is not encouraged as it might lead to poor results/returns, so to solve this issue some of the platforms created no code algo-trading services these platforms allow you to trade without any coding knowledge (platform like Streak by Zerodha) helps the user to backtesting and automate the trades based on Rule-based strategy. The study shown by the author algo-trading and market volumes raised exponentially from 2003 to 2012 as shown in the figure-1 The world algo trading market is set to break new highs in 2030.

Manuscript received on October 04, 2021.

Revised Manuscript received on October 18, 2021.

Manuscript published on November 30, 2021.

*Correspondence Author

Penumatcha Bharath Varma*, Department of Computer Science Mallareddy College of Engineering and Technology, MRCET Campus Hyderabad, India. Email: cogs132@gmail.com

Dr. Jaypal Medida, Professor, Department of Computer Science Mallareddy College of Engineering and Technology, MRCET Campus Hyderabad, India, Email: jaypalmedida@gmail.com

Neeraj Kasheety, Department of Computer Science Mallareddy College of Engineering and Technology, MRCET Campus Hyderabad, India

Hanumanula Sravya, Department of Computer Science Mallareddy College of Engineering and Technology, MRCET Campus Hyderabad, India

Chinthapalli Amarnath Reddy, Department of Computer Science Mallareddy College of Engineering and Technology, MRCET Campus Hyderabad, India

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

Algorithmic Trading. Percentage of Market Volume

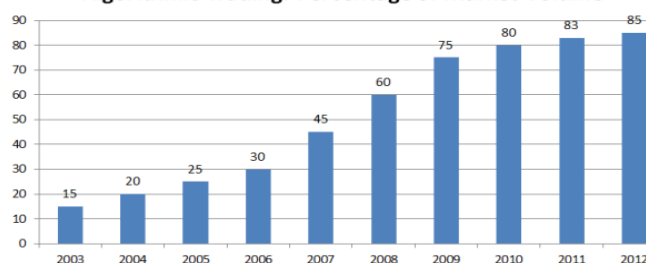


Figure 1. Algorithmic Trading Percentage of Market Volume

A. Few advantages of algo trading:

- a) Historical Assessment (Backtesting)
- b) Efficiency
- c) Rule-Based implementation
- d) Comparison
- e) Higher Frequency

II. RELATED WORK

This section provides an overview of the many strategies for algorithmic trading with machine learning that are already in use. Machine Learning is deployed in the software for algorithmic trading. Only Random Forest, Probit regression, Genetic Algorithms including such Deep MLP Neural Network, Support Vector Machine Regression, Random forests, and Gradient boosted decision trees are now available for trading using Machine Learning approaches.

A. Existing Softwares –

- a) **Zerodha Streak:** It's a great tool for non-developers and non-programmers, Helping every individual to use algos without programming knowledge.
- b) **Algonomics:** NSEIT created this trading platform, which is recognised as one of the best algo trading platforms. It offers ultra-low latency execution, which is beneficial for high-volume transactions by investment banks, fund managers, and retail algo traders.
- c) **Omnesys Nest:** It offers all of the advantages of a cutting-edge Fin-Tech trading platform, such as low latency rates and high levels of performance, and is actively utilised by HNI (high net worth people), institutional, and retail traders.

B. Automated trading system:

Automated Trading System (ATS) uses computer software to execute buy and sell orders automatically on market exchanges with the help of alpha (Trading Strategies) to achieve significant returns when compared to traditional trading, some computer software automatically generates orders based on predefined rules using events and news these systems also called as event-based systems

To build an efficient ATS we need a few parameters to be checked

- Alpha (Trading Strategy)
- Backtesting

B.1. Alpha

Trading on the Exchange always gives you the potential to lose money at an alarming pace, thus it is always necessary to "know oneself" more than what alpha you select. The most significant attributes one needs have to establish a successful ATS are discipline, emotional detachment, and patience.. As algos place the trades on behalf of us, we will face difficulties in the extended drawdown phase (The drawdown is the measure of the decline from a historical peak) . There will always be a shelf life for Alpha, so there should be ongoing investigation into trading strategies to maintain a profitable portfolio. Programming expertise is important in creating robust trading systems, programming languages such as python,r,SQL, help in creating backtesting engines, order execution, and report generation. It also allows consumers to

develop innovative techniques and trading strategies and you'll have complete control over the "Tech Stack."

Finally, do not expect to get wealthy in a short period of time. Algo trading is not a get-rich-quick scheme; in fact, it might be a get-poor-quick scheme. To be effective in algorithmic trading, much study, care and patience, as well as greater knowledge experience, are required. Consistent earnings might take months, if not years, to create (Returns). Nowadays, all trading methods are available on the internet with a few mouse clicks, but the question is who to trust? The backtesting module will cover the approaches and procedures, we need to have proven and backtested alphas in a **pipeline** so that we can use it based on the market conditions, having a pipeline of proven alphas or with higher Sharpe ratios always has the advantage of getting higher returns.

B.2. Backtesting

Backtesting is the general method for seeing how well a strategy works. Backtesting is the basic element in the creation of trading strategies. It gives answers to complex questions like how profitable a trading strategy was in historic times. It helps to optimize it by running simulations that generate dynamic results showing Sharpe ratio and returns before risking any capital loss. If the backtesting returns good results (high profits with reasonable risk or minimal risk), it will encourage getting this strategy to go alive. If the results are not satisfactory, we can change the parameters and optimize the alpha then follow backtesting again.

There are four major biases in backtesting,

- **Cognitive Bias:** This phenomenon covers instances in which a person is confronted with a systemic error in thinking, affecting one's ability to make judgments and decisions based on objective reality rather than subjective impressions.
- **Look-Ahead Bias:** It occurs by using data in a study that would not have been available during the period being analysed. This can lead to false results in the study and leads to capital-loss
- **Survivorship Bias:** It is a sort of sample selection bias that happens when a data collection only examines extant observations while ignoring observations that have previously ceased to exist.
- **Optimization Bias:** It is also known as unrealistic optimism, a cognitive bias in which someone believes that they are less likely to suffer a terrible occurrence than others..

C. Backtesting v/s Reality

There are several elements that may be considered in order to create decent realistic returns through backtesting. Overfitting, inadequate data (unclean data), inappropriate management of transaction charges and other costs, market news change, and backtest performance often differs from real strategy deployment owing to trading constraints are all risks. As a result, one must be aware that future backtesting findings are highly unlikely to directly mirror historical future backtesting outcomes.



III. DATASETS AND FEATURES

Free resources of data, Stock data can be fetched from google (Gupta et al. 2019), yahoo, *yfinance* (Gupta et al. 2019) [2] is the python module for downloading the historical stock prices from yahoo finance, but the drawback is that it only provides historical data but not real-time data.

For real-time data, we can use paid API (application program interface) such as Zerodh's kite Connect, Market stack, EOD Historical Data, Alpha Vantage, IEX Cloud App, Tiingo, Intrinio, Quandl, Polygon.

The data is passed through the gateway to the respective systems as shown in Figure (2) below usually this process will be deployed in the cloud

A. Features:

- The order book (Book Builder)
- Order Manager
- Strategy
- Command and control

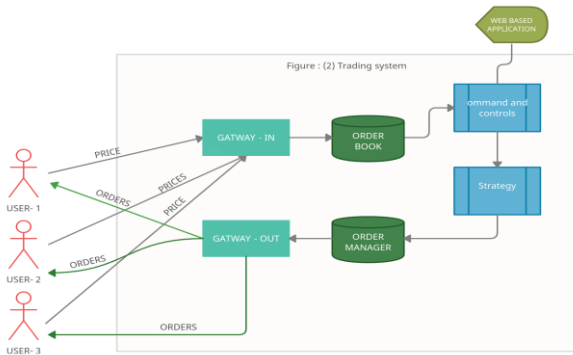


Figure 2. Trading System Design

IV. METHODS

Many principles from statistical machine learning are employed in the subject of quantitative finance, which is a huge multidisciplinary topic. Regression, classification, and time series models are examples of popular applications for such models.

A. Regression It is a supervised machine learning approach that provides both predictive and inferential statistics. Regression approaches are commonly used in the field of algorithmic trading. Regression helps to build a model between the response and predictor variables. The regression approach is linear regression way, which builds upon a linear connection in between the response and the predicted variable. This model includes parameter estimates for the predictor variables, and the parameters are estimated using **ordinary least squares**. (OLS)(Hartog et al. n.d.) [3] Let's say "A" is the response variable and "B" is the predictor variable. Consider a linear regression model is applied on predicting NIFTY 50 data (*The NIFTY 50 is a benchmark Indian stock market index*) in the following date, from price data over last week's trading days i.e. 5 days, By using the OLS approach the model can be utilized across 10000 past data points, then when a new market data arrives from the NIFTY 50 it can be inputted into the model ("B") to generate a predicted price for tomorrow's trading session. This is an example of a simple predictive trading system using linear regression. In the case of the INFERENCE approach, we determine the strength of the relationship between ("A") and

("B"). This is more significant if we have to know the intuition behind the relationship, But the inference is often less used in algo-trading, as the prediction is more important than the reason behind it. Hence one should not totally rely on algorithmic trading due to the prioritized of over-fitting to noise in the data.

B. Classification: In general it is the process of converting or dividing input features like images, text and any other formats into outputs that separate into two categories or classes one with the positive values and the other with the negative values after that it uses its learnings to classify new observations It has two categories binary and multi-class Binary Class only works on two possibilities where as in multi-class classifiers work with multiple classes or categories here in this scenario we are using only binary classifier , e.g. "low", "medium", "high". In the other scenarios Classification algorithms classifiers are widely used in Algorithmic Trading usually in the market direction prediction. we are able to use continuous predictors that is "B", such as market prices as observations. We will consider both linear and Logistic Regression, Linear Discriminant Analysis, Support Vector Machines (SVM)(Yang et al. n.d.) [4] and Artificial Neural Networks (ANN) (Bring et.al.2012) [5]

C. Time Series models: Prediction of financial time series data is critical in algorithmic trading. The idea is to predict the market's next/future outcome. As a result, time series modelling may be viewed as a hybrid of regression and classification. The predictors are frequently based on prior values and, on rare occasions, on current values, whereas the responses are frequently based on projected future values. We are interested in two types of time series models in algorithmic trading. The first group comprises linear autoregressive integrated moving average (ARIMA) models, which are used to characterise changes in a time series' real value. AR is an abbreviation for autoregressive conditional Another sort of time series is heteroskedasticity (ARCH). (Bollerslev 1986) [6] models are used to model the variation (or volatility) of time series across time. ARCH models anticipate future values by analysing a time series' historical values (volatilities) (volatilities). Stochastic volatility models, on the other hand, employ several stochastic time series to characterise volatility. All of the historical price time series are discrete, with only a few possible values. In the topic of algorithmic trading, continuous time series models are frequently investigated. The well-known Geometric Brownian Motion model, the Heston Stochastic Volatility model, and the Ornstein-Uhlenbeck [7] model all represent covariance. The ADF test takes use of the fact that if a price series spans mean reversion, the next price level is exactly proportionate to the present price level. The ADF is statistically founded on the concept of testing for parameters in a sample of autoregressive time series data. Consider the following scenario: known as a linear lag model of order k which is present in time series. This model proposes, the change in the time series value is proportional to a constant, the time itself, and the previous k values of the time series, as well as an error term:

$\Delta y_k = \alpha + \beta k + \gamma y_{k-1} + \delta_1 \Delta y_{k-1} + \dots + \delta_{k-1} \Delta y_{t-k+1} + \epsilon$ (1).
 Where α is a constant, β represents the coefficient of a temporal trend and $\Delta y_k = y(k) - y(k - 1)$. If $\gamma = 0$, ($\alpha = \beta = 0$) imply non mean reverting. If hypothesis = 0 is rejected, the next movement of the price series data is proportional to the mean price data, deeming the random walk unsustainable. When we say "statistical test," we mean this.

The ADF test was conducted by computing the test statistic, $DF\tau$, that are used in the decision to reject the null hypothesis, as well as evaluating the distribution of the test statistic in accordance with the critical values to determine whether to reject the null hypothesis.. Let us compute the test statistic. ($DF\tau$) . This is calculated by dividing the sample proportionality constant by the standard error of the sample proportionality constant:

$$DF\tau = \hat{\gamma} / SE(\hat{\gamma}) \quad (2)$$

Now that we have the test statistic, one important practical concern for traders is that any consistent long-term drift in a price is far lower in size than any short-term surprise change, therefore the drift is frequently considered to be zero ($= 0$) for the linear lag model given above. Because we are contemplating a lag model of order k , we must actually specify p to a specific number. For trading study, setting $k = 1$ is generally adequate to allow us to reject the null hypothesis. We can use the pandas package to compute the Augmented Dickey-Fuller test. For example, we will take AMAZON stock from January 1st, 2000 to January 1st, 2015.

The code below demonstrates how to run an ADF test in Python. Algorithm for Implementation of *Augmented Dickey-Fuller test* in python.

- Step 1:** Start
- Step 2:** Fetching the data from the trusted and valid source
 If data is not available use paid api's
- Step 3:** Clean data if any
- Step 4:** Here in this model we assign $B=0$ and used linear leg, model
- Step 5:** If $p = 1$, rejects null Hypothesis for trading research and fetching a optimized trading strategy
- Step 6:** If $p = 0$, accepts null Hypothesis and stops the process
- Step 7:** After Step 5 it performs ADF test
- Step 8:** Data Visualization of the results
- Step 9 :** Stop

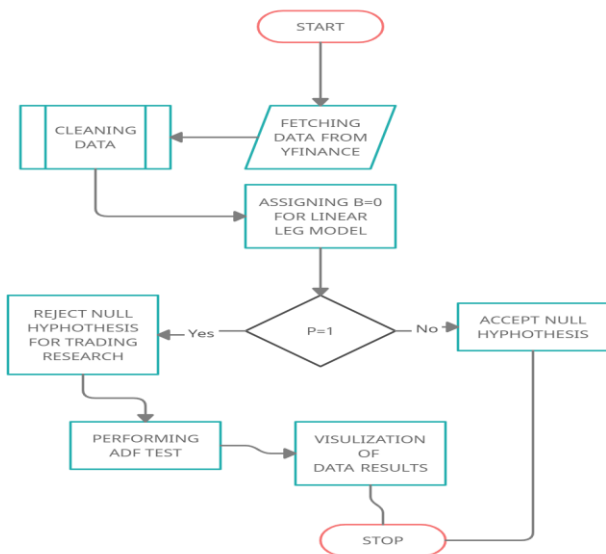


Figure 3. Implementation of Augmented Dickey-Fuller Test

Below are the results for Amazon stock from 2000-2014
 Test-Statics = 0.049177575166452235 , p-value = 0.96241494632563063 ,Number of data points in a sample = 3771 , Dictionary contains 1,5,10 percentages respectively
 “{‘1%’: -3.4320852842548395, ‘10%’: -2.5671781529820348, ‘5%’: -2.8623067530084247},”.

V. IMPLEMENTAION

To achieve greater Sharpe ratios in our trading, we must examine higher-frequency intraday trading tactics for our trading system. 2017 [8] Stübinger et al. High-quality intraday data is costly.

The level equivalent most of the components required for interaction with the other system components and establishes the template for their performance. It is in charge of collecting market data, engaging with the AlgorithmManager, generating log events, and sanitising the care order attributes, changing them to the needed type while checking them (for example, altering the order quantity from string to int). Its abstract methods serve as a blueprint for the classes that inherit it. These methods are concerned with the algorithm execution characteristics (load, reload, finish, stop, and execute), but also the validation associated with specific parameters and its semantics.

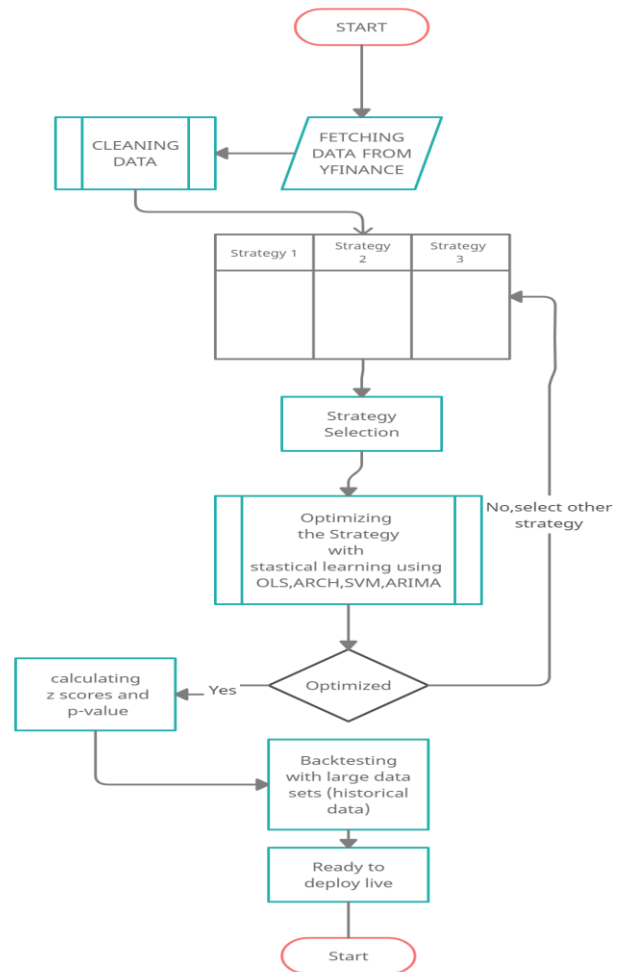


Figure 4. Algorithm of the procedure



VI. RESULTS

The backtesting output the above alpha model using backtesting simulation ,the outcome of the strategy performance is well grounded to the reality during this period Below are the results. The backtest is performed on the AMZN on the historic data from 2000 to 2015 and creating summary stats, equity bend, drawdown,signals,order,fill Parameters of the strategy back testing :

- Starting funds
- Ending funds
- Duration
- Exposure Time
- Equity Final in terms of USD
- Return percentage
- Buy and Hold returns useful in longterm quoted as long term capital gains
- Volatility percentage
- Sharpe Ratio
- Sorting Ratio
- Calmar ratio
- Maximum Drawdown
- Average Drawdown
- Maximum Drawdown duration
- Average Drawdown duration
- Number of trades
- Win Ratio
- Best Trade
- Average Trade in percentage
- Maximum Trade duration
- Average Trade Duration
- Profit Factor

- Expectancy in percentage
- SQN
- Name of the Strategy

Start	0
End	6155
Duration	6155
Exposure Time [%]	90.6433
Equity Final [\$]	1745.87
Equity Peak [\$]	24482.8
Return [%]	-82.5413
Buy & Hold Return [%]	175477
Return (Ann.) [%]	0
Volatility (Ann.) [%]	NaN
Sharpe Ratio	2
Sortino Ratio	NaN
Calmar Ratio	1
Max. Drawdown [%]	-95.3948
Avg. Drawdown [%]	-37.8297
Max. Drawdown Duration	5869
Avg. Drawdown Duration	1021.5
# Trades	287
Win Rate [%]	40.4181
Best Trade [%]	223.522
Worst Trade [%]	-40.2805
Avg. Trade [%]	-0.591771
Max. Trade Duration	74
Avg. Trade Duration	19.4321
Profit Factor	1.16876
Expectancy [%]	0.803253
SQN	-0.42368
_strategy	SmaCross

Table 1: Backtesting Data of Amazon

The output is saved into the outputdata.csv file

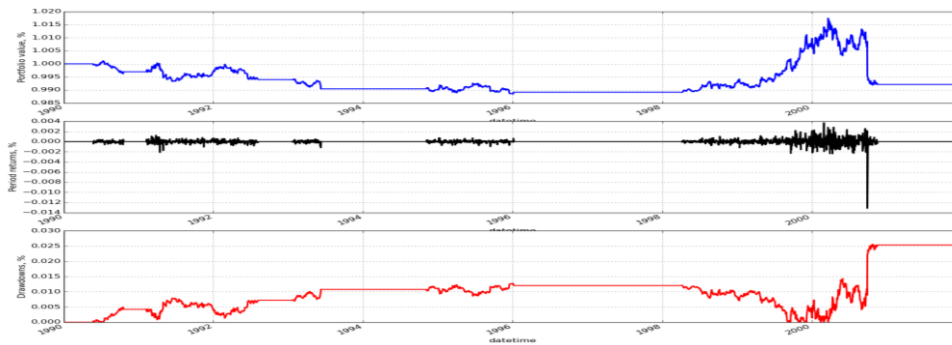


Figure 5. performance of the Strategy Before optimizing

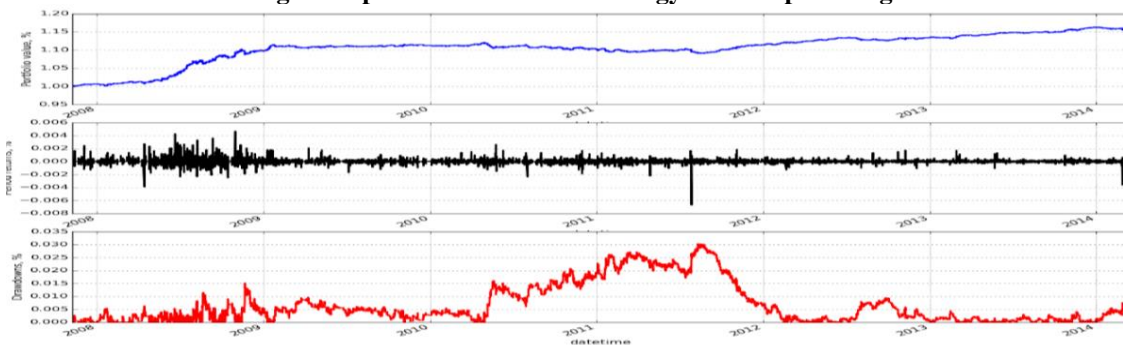


Figure 6. Performance of the Strategy After optimizing



A predictive model and a viable and successful algorithmic approach are not the same thing. Focusing on improving parameters with a direct impact on Sharpe ratio and Drawdown. To do this, we will employ mean reversion trading. We'll look at a specific strategy with three parameters that use a grid search mechanism to traverse the space produced by the cartesian product of the parameters. Sharpe ratio and Drawdown performance can now be improved. Matplotlib will be used to do this. We will read the outputdata.CSV file and restructure the data so that the findings may be seen.

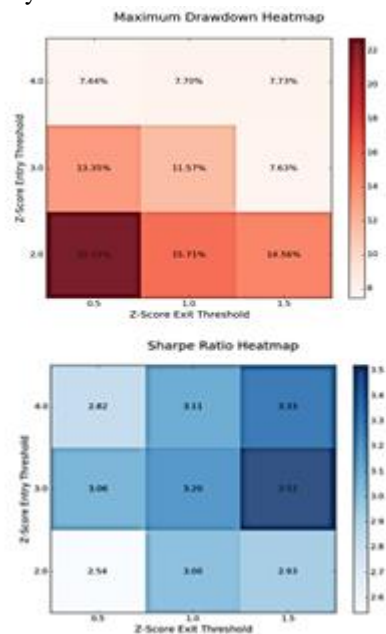


Figure 7. Heatmap of MAX drawdown and Sharpe Ratio

VII. CONCLUSION

The Sharpe ratio and drawdown results from the previous model are not promising for live trading, but the main intuition of choosing the right model and optimizing parameters like drawdown and Sharpe ratio resulted in a good significance of relative high entry and exit points based on back testing results, allowing us to deploy into live trading.

REFERENCES

1. H. Valenciano Farré, "Machine-Learning-powered Algorithmic Trading Bachelor Degree in Informatics Engineering School of Informatics of Barcelona-UPC BarcelonaTECH Machine-Learning-powered Algorithmic Trading," 2019.
2. A. Gupta and P. Chaturvedi, "Investing Data with Machine Learning using Python IoT use in Social Media Data Analysis View project," 2019. [Online]. Available: <https://www.researchgate.net/publication/334603195>.
3. J. Hartog, P. T. Pereira, and J. A. C. Vieira, "Changing Returns to Education in Portugal During the 1980s and Early 1990s: OLS and quantile regression estimators."
4. H. Yang, L. Chan, and I. King, "Support Vector Machine Regression for Volatile Stock Market Prediction."B. Smith, "An approach to graphs of linear forms
5. Y. Bing, J. K. Hao, and S. C. Zhang, "Stock Market Prediction Using Artificial Neural Networks," *Advanced Engineering Forum*, vol. 6–7, pp. 1055–1060, Sep. 2012, doi: 10.4028/www.scientific.net/aef.6-7.1055.
6. T. Bollerslev, "GENERALIZED AUTOREGRESSIVE CONDITIONAL HETEROSKEDASTICITY," 1986.Y. Yoroza, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interfaces(Translation

7. D. T. Gillespie, "Exact numerical simulation of the Ornstein-Uhlenbeck process and its integral."
8. J. Stübinger and S. Endres, "Pairs trading with a mean-reverting jump-diffusion model on high-frequency data," 2017, [Online]. Available: <https://www.iwf.rw.fau.de/research/iwf-discussion-paper-series/J>.
9. A. Nunes Correia Gouveia, "André Nunes Correia Gouveia Machine Learning Applications on Algorithmic Trading in the Foreign Exchange Market," 2020. [Online]. Available: <https://github.com/joaomloureiro/novathesis>
10. Morton. Glantz and Robert. Kissell, *Multi-Asset Risk Modeling: Techniques for a Global Economy in an Electronic and Algorithmic Trading Era*. Elsevier Science, 2013.
11. G. Chen, Y. Chen, and T. Fushimi, "Application of Deep Learning to Algorithmic Trading."
12. A. Indd, "The Financial Times Guide to Investing." [Online]. Available: www.pearsoned.co.uk
13. Mathur, M., Mhadalekar, S., Mhatre, S., & Mane, V. (2021). *Algorithmic Trading Bot*. ITM Web of Conferences, 40, 03041. <https://doi.org/10.1051/itmconf/20214003041>

AUTHORS PROFILE



Penumatcha Bharath Varma, Computer Science and Engineering Graduate From Malla Reddy College Of Engineering And Technology (2017-2021), Hyderabad, Telangana, Member In Computer Society Of India.



Neeraj Kasheety, Computer Science and Engineering Graduate from Malla Reddy College of Engineering and Technology (2017-2021), Hyderabad, Telangana.



Hanumanula Sravya, Computer Science and Engineering Graduate From Malla Reddy College Of Engineering And Technology (2017-2021), Hyderabad, Telangana



Chinthapalli Amarnath Reddy, Computer Science and Engineering Graduate from Malla Reddy College Of Engineering And Technology (2017-2021), Hyderabad, Telangana



Dr. Jaypal Medida, Professor Department of Computer Science Mallareddy College of Engineering and Technology, MRCET Campus Hyderabad, India, Email: jaypalmedida@gmail.com