Crossover Trading: Estimating Investments using Machine Learning

Submitted in partial fulfilment of the requirements for the degree of

Bachelor of Technology
in
Information Technology

By
AAKANKSHA SATISH BAGAL
20BIT0190

Under the guidance of
Prof. Harshita Patel
SCORE
VIT, Vellore

Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

May, 2024

DECLARATION

I hereby declare that the thesis entitled "Crossover Trading: Estimating Investments using Machine Learning" submitted by me, for the award of the degree of *Bachelor of Technology in Information Technology* to VIT is a record of bonafide work carried out by me under the supervision of Prof. Harshita Patel

I further declare that the work reported in this thesis has not been submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university

Place: Vellore

Date: 8th May, 2024

Signature of the Candidate

<u>CERTIFICATE</u>

This is to certify that the thesis entitled "Crossover Trading: Estimating Investments using Machine Learning" submitted by Aakanksha Satish Bagal, 20BIT0190, SCORE, VIT, for the award of the degree of *Bachelor of Technology in Information Technology,* is a record of bonafide work carried out by him/her under my supervision during the period, 01.12.2023 to 30.04.2024, as per the VIT code of academic and research ethics

The contents of the report have not been submitted and will not be submitted either part or in full, for the award of any other degree or diploma in this institute or any other institute or university. The thesis fulfils the requirements and regulations of the University and in my opinion meets the necessary standards for submission

Place: Vellore

Date: 8th May, 2024

Signature of the Guide

Internal Examiner **External Examiner**

Dr. PRABHAVATHY P

SCORE

ACKNOWLEDGMENTS

We would like to express our sincere gratitude to Prof. Harshita Patel for her invaluable guidance and support throughout the development of this project. Her expertise and deep understanding of the machine learning concepts have been instrumental in shaping our research and implementing our machine learning-based system for stock market prediction.

Prof. Patel's commitment to advancing machine learning technology and her passion for enhancing artificial intelligence have been truly inspiring. Her insights and feedback have greatly influenced the direction of our project, ensuring its relevance and practicality in the field.

We are deeply grateful to Prof. Harshita Patel for her mentorship, patience, and unwavering support. Her expertise and encouragement have been instrumental in the success of this project, and we sincerely appreciate the opportunity to work under her guidance.

Aakanksha Satish Bagal

Executive Summary

Crossovers are crucial for predicting the performance of financial instruments, helping to anticipate market trend changes, such as reversals or breakouts. This method examines two or more moving averages over different time frames to predict market shifts. This study uses the monthly and yearly moving averages instead of the commonly used 50-day and 200-day moving averages.

This research focuses on analysing current trends in the stock market and predicting next-day growth by employing various prediction models, including LSTM (Long Short-Term Memory), K-Nearest Neighbours(KNN), and regression analysis. We utilize stock market data imported from the yfinance API, applying these models and crossover techniques to determine future actions. A bullish crossover is identified when the short-term (monthly) moving average crosses above the long-term (yearly) moving average, indicating a buying opportunity. Conversely, a sell signal is generated when the short-term moving average dips below the long-term average. The goal is to pinpoint the most effective prediction model through comparative analysis, thereby improving the precision and reliability of market trend forecasts and investment decisions.

CONTENTS **Page no.**

List of Figures

List of Tables

List of Abbreviations

LSTM	Long-short Term memory
ARIMA	Autoregressive Integrated Moving Average
KNN	K-Nearest Neighbors
S&P500	Standard and Poor's 500
FX	Foreign Exchange market
SPDR	Spider
DLR	Deep reinforcement learning
MA	Moving Average
yfinance	Yahoo finance
RMSE	Root mean square error
MAE	Mean absolute error
R2	R-squared score

1. INTRODUCTION

1.1. OBJECTIVE

The objective of this project is to enhance stock market trading strategies by predicting stock prices using a Long Short-Term Memory (LSTM) algorithm and identifying optimal buy and sell points through an ARIMA model-based crossover trading strategy. Specifically, we aim to leverage the LSTM's capability to analyze historical data and recognize long-term dependencies to accurately forecast stock prices. Concurrently, the ARIMA model will be utilized to generate trading signals, pinpointing the most advantageous times to buy or sell based on these predictions. By integrating the predictive power of LSTM with the signal generation of ARIMA, our approach seeks to refine trading strategies, enhancing both the accuracy of market entry and exit points and the overall adaptability of the models to market changes. Ultimately, this project aims to provide traders with actionable insights, thereby improving decision-making and potentially increasing profitability through advanced predictive analytics.

1.2. MOTIVATION

The motivation for this project stems from the inherent complexities and volatilities of the stock market, which present substantial challenges to both novice and experienced traders. Traditional trading strategies often fall short in predicting rapid market changes and recognizing long-term trends. This shortfall provides a unique opportunity to leverage advanced machine learning techniques to improve prediction accuracy and decision-making processes.

Specifically, the use of Long Short-Term Memory (LSTM) networks and ARIMA models harnesses the strengths of deep learning and statistical analysis. LSTMs are particularly well-suited for financial time series data due to their ability to capture temporal dependencies and remember information over extended periods, which is critical in the context of volatile markets. On the other hand, ARIMA models excel in forecasting short-term deviations, making them ideal for generating precise trading signals.

By integrating LSTM with ARIMA, this project aims to create a robust framework that not only predicts future stock prices with high accuracy but also assists in making informed trading decisions by pinpointing the optimal times to buy and sell. This approach promises to significantly enhance trading strategies, ultimately leading to increased profitability and risk management for market participants. This project, therefore, seeks to push the boundaries of traditional market analysis and provide a more dynamic, data-driven method for stock market engagement.

1.3. BACKGROUND

The stock market is characterized by its dynamic and often unpredictable nature, driven by countless variables that include economic indicators, political events, and trader sentiment, among others. This complexity makes it a challenging arena for investors who seek to maximize returns while minimizing risks. Traditional financial theories and methods, while foundational, often fail to capture the nuanced behaviours of modern financial markets, leading to gaps in effective prediction and trading strategies.

Over the past few decades, the advent of computational finance and the availability of high-frequency trading data have paved the way for more sophisticated analytical techniques. Machine learning, in particular, has emerged as a powerful tool for financial modelling, capable of analysing large volumes of data to identify patterns that are imperceptible to human analysts. Among these technologies, Long Short-Term Memory (LSTM) networks and Autoregressive Integrated Moving Average (ARIMA) models stand out.

LSTMs are a type of recurrent neural network suitable for sequence prediction problems and have proven effective in capturing long-term dependencies in time-series data, which is critical in the volatile environments of financial markets. Conversely, ARIMA models are more traditional but highly effective in their capacity to model a wide range of sequences with trends and seasonality, which makes them invaluable for short-term forecasting in stock trading.

This project is grounded in the need to integrate these two methodologies to leverage their respective strengths. By combining the memory capabilities of LSTM for understanding long-term trends and the precise, short-term predictions afforded by ARIMA, this project aims to develop a more holistic approach to stock market forecasting and trading strategy development. This background sets the stage for exploring how such an integrated approach can be systematically applied to real-world trading scenarios, driving forward the fields of algorithmic trading and financial analysis.

2. LITERATURE SURVEY

2.1. The Moving Average Crossover strategy: Does it work for the S&P500 Market Index?

The paper investigates the use of an optimized moving average crossover strategy for the S&P500, by using the SPDR S&P500 Exchange Traded Fund as a proxy for the US market index. The optimized strategy is evaluated against a buy and hold strategy over the five distinct waves which were witnessed during the 1993- 2014 period. The annualized returns, annualized risk and the Sharpe performance measure are used as indicators to compare between the two strategies.

Findings tend to support higher absolute returns and risk for the buy-and-hold strategy, particularly during correction waves. When compared to the buy-and-hold strategy over the post financial crisis period, the optimized double cross over strategy resulted in a relatively lower risk and returns. The market timing strategy still outperformed the naïve buy- and-hold strategy, with a relatively higher Sharpe performance measure.

2.2. Recommending Cryptocurrency Trading Points with Deep Reinforcement Learning Approach.

The research used Deep reinforcement learning (DRL) on the stock market, and hence developed an application that observes historical price movements and takes action on real-time prices. The proposed algorithm was tested with three—Bitcoin (BTC), Litecoin (LTC), and Ethereum (ETH)—crypto coins' historical data. The experiment on Bitcoin via DRL application shows that the investor got 14.4% net profits within one month. Similarly, tests on Litecoin and Ethereum also finished with 74% and 41% profit, respectively.

2.3. Machine learning classification and regression models for predicting directional changes trend reversal in FX markets

Most forecasting algorithms in financial markets use physical time for studying price movements, making the flow of time discontinuous. The use of physical time scale can make traders oblivious to significant activities in the market, which poses a risk. Directional changes (DC) is an alternative approach that uses event-based time to sample data. The proposed DC-based framework, uses machine learning algorithms to predict when a trend will reverse. This allows traders to be in a position to take an action before this happens and thus increase their profitability. Combining the approach with a novel DC-based trading strategy and perform an in-depth investigation, by applying it to 10-min data from 20 foreign exchange markets over a 10-month period. The total number of tested datasets is 1,000, which allows to argue that the results can be generalised and are widely applicable. Comparing the results to ten benchmarks (both DC and non-DC based, such as technical analysis and buy-and-hold). The findings show that the proposed approach is able to return a significantly higher profit, as well as reduced risk, and statistically outperform the other trading strategies in a number of different performance metrics.

2.4. Trading Performance Analysis: A Comparisons Between the Original MA Crossover and Modified MA Crossover Strategy

This paper empirically analyses the Trading Performance by using technical analysis approach. The original moving average (MA) crossover strategy as compare with the modified moving-average crossover strategy. The modified trading rules are the rules that been established to trading rules such as entry rule, exit rule, holding rule, and stoploss rule. The results show The MA short of 10-period for modified strategy underperform the original strategy, except for MA (10,100). The modified MA (20,200), (50,100), (50,200), and (100,200) underperform the original strategy. Only modified MA (20,50) and (20,100) outperform the original strategy. The outperformance and underperformance due to the stricter additional trading rule that reduces trading signals, and thus lower number of trades.

2.5. CREST: Cross-Reference to Exchange-based Stock Trend Prediction using Long Short-Term Memory

A cross-reference to exchange-based stock trend (CREST) prediction method is proposed using long short-term memory (LSTM). The daily stock prices of Wipro Limited (WIPRO) company, which is listed on NSE as well as BSE, have been collected and the stock price movement of WIPRO in one exchange has been analysed for predicting the trend in the other exchange. To identify the applicability of our approach, CREST has also experimented with Infosys Limited and Larsen & Toubro Infotech Limited companies. The performance is evaluated using root-mean-square error and directional accuracy along with precision, recall, and F-measure for the results of all three companies. Results have shown that the proposed CREST prediction generates higher accuracy than individual trend prediction using LSTM. It can be applied to study the inherent complex stock trading patterns of correlated companies.

2.6. Information fusion-based genetic algorithm with long short-term memory for stock price and trend prediction

This research proposes an information fusion-based GA approach with inter-intra crossover and adaptive mutation (ICAN) for stock price and trend prediction. Inspired by the genetic diversity and survival capability of various organisms, this proposed approach aims to optimize parameters of a long short-term memory prediction model, and selects a set of features; to address these problems of interest, we integrate inter-chromosome as well as conditional intra-chromosome crossover operations along with adaptive mutation to diversify the potential chromosome solutions.

2.7. Optimizing LSTM for time series prediction in Indian stock market

Long Short Term Memory (LSTM) is among the most popular deep learning models used today. It is also being applied to time series prediction which is a particularly hard problem to solve due to the presence of long term trend, seasonal and cyclical fluctuations and random noise. The performance of LSTM is highly dependent on choice of several hyper-parameters which need to be chosen very carefully, in order to get good results. Being a relatively new model, there are no established guidelines for configuring LSTM. In this paper this research gap was addressed. A dataset was created from the Indian stock market and an LSTM model was developed for it. It was then optimized by comparing stateless and stateful models and by tuning for the number of hidden layers.

2.8. Stock Closing Price Prediction using Machine Learning Techniques

Artificial Neural Network and Random Forest techniques have been utilized for predicting the next day closing price for five companies belonging to different sectors of operation. The financial data: Open, High, Low and Close prices of stock are used for creating new variables which are used as inputs to the model. The models are evaluated using standard strategic indicators: RMSE and MAPE. The low values of these two indicators show that the models are efficient in predicting stock closing price.

2.9. Deep learning-based feature engineering for stock price movement prediction

This paper proposes a novel end-to-end model named multi-filters neural network (MFNN) specifically for feature extraction on financial time series samples and price movement prediction task. Both convolutional and recurrent neurons are integrated to build the multi-filters structure, so that the information from different feature spaces and market views can be obtained. Applying MFNN for extreme market prediction and signal-based trading simulation tasks on Chinese stock market index CSI 300. Experimental results show that our network outperforms traditional machine learning models, statistical models, and single-structure(convolutional, recurrent, and LSTM) networks in terms of the accuracy, profitability, and stability.

2.10. Machine Learning Methods in Algorithmic Trading Strategy Optimization – Design and Time Efficiency

The main aim of this paper was to formulate and analyse the machine learning methods, fitted to the strategy parameters optimization specificity. The most important problems are the sensitivity of a strategy performance to little parameter changes and numerous local extrema distributed over the solution space in an irregular way. The methods were designed for the purpose of significant shortening of the computation time, without a substantial loss of strategy quality. The efficiency of methods was compared for three different pairs of assets in case of moving averages crossover system. The problem was presented for three sets of two assets' portfolios. In the first case, a strategy was trading on the SPX and DAX index futures; in the second, on the AAPL and MSFT stocks; and finally, in the third case, on the HGF and CBF commodities futures. The methods operated on the in-sample data, containing 16 years of daily prices between 1998 and 2013 and was validated on the out-of-sample period between 2014 and 2017. The major hypothesis verified in this paper is that machine learning methods select strategies with evaluation criterion near the highest one, but in significantly lower execution time than the brute force method (Exhaustive Search).

2.11. Evaluating the Effectiveness of Common Trading Models

Created original versions of popular models, like linear regression, K-Nearest Neighbour, and moving average crossovers, and tested how each model performs on the most popular stocks and largest indexes. With the results for each, compared the models, and understand which model reliably increases performance. The trials showed that while all three models reduced losses on stocks with strong overall downward trends, the two machine learning models did not work as well to increase profits. Moving averages crossovers outperformed a continuous investment every time, although did result in a more volatile investment as well.

2.12. A Machine Learning Framework for Stock Selection

This paper demonstrates how to apply machine learning algorithms to distinguish “good” stocks from the “bad” stocks. To this end, constructed 244 technical and fundamental features to characterize each stock, and label stocks according to their ranking with respect to the return-to-volatility ratio. Algorithms ranging from traditional statistical learning methods to recently popular deep learning method, e.g. Logistic Regression (LR), Random Forest (RF), Deep Neural Network (DNN), and the Stacking, are trained to solve the classification task. Genetic Algorithm (GA) is also used to implement feature selection. The effectiveness of the stock selection strategy is validated in Chinese stock market in both statistical and practical aspects, showing that: 1) Stacking outperforms other models reaching an AUC score of 0.972; 2) Genetic Algorithm picks a subset of 114 features and the prediction performances of all models remain almost unchanged after the selection procedure, which suggests some features are indeed redundant; 3) LR and DNN are radical models; RF is risk-neutral model; Stacking is somewhere between DNN and RF. 4) The portfolios constructed by our models outperform market average in back tests.

2.13. A Decision Support System for Technical Analysis of Financial Markets Based on the Moving Average Crossover

A decision support system was developed based on the MA crossover technique that is capable of providing descriptive statistics for the time series data of market indices and securities, evaluating the statistical significance of returns generated by any MA rule, searching for the optimal MA rule that generates the highest significant returns and scan among a group of securities for the latest signals and highest returns. The system has the added of advantage of searching for the optimal

MA rule among a large universe of MA rules. The DSS system was used to investigate the predictive capabilities of the MA crossover with respect to the Egyptian Exchange Stock market. The lengths of 1-20, 1-25 and 1-30 along with the closing price emerged as the most profitable for the rules. The exponential MA model dominated as the most profitable model for many securities, while the simple MA model was effective for a few securities. The results obtained provide strong evidence that the MA crossover technique can predict the Egyptian stock market index and its securities and reject the null hypothesis that the returns earned by the technique are equal to the unconditional buy-and-hold strategy.

2.14. A novel stock trading utilizing long short term memory prediction and evolutionary operating-weights strategy

This paper proposes a novel approach to enhance investment returns by integrating Long Short Term Memory (LSTM) predictions with the Evolutionary Operating-weights (EOW) algorithmic strategy. The proposed method employs a multi-layer LSTM to forecast future stock prices, incorporating the predictions with real market data, subsequently deriving an operational strategy using the EOW algorithm. The results of this research indicate that this proposed methodology outperforms existing approaches.

2.15. The cross-over effect of irrational sentiments in housing, commercial property, and stock markets

This paper examines the dependence in irrational sentiments across housing, commercial property, and stock markets. Our empirical results document an important and lasting impact that commercial real estate sentiment and returns have on broader financial markets. We also show that the cross-over effects of market sentiments are not consistent with cross-over effects in market returns. Sentiments and returns in housing and stock markets exhibit strong dependence on other markets, whereas they evolve independently in commercial real estate. While housing and stock market returns respond to irrational sentiment in commercial real estate markets, the opposite is not true.

3. OVERVIEW OF THE PROPOSED SYSTEM

3.1. INTRODUCTION

In today's rapidly evolving financial landscape, accurate prediction of stock prices is paramount for traders aiming to navigate market volatility and capitalize on opportunities. This report introduces a novel approach that combines two potent methodologies: Long Short-Term Memory (LSTM) networks and Autoregressive Integrated Moving Average (ARIMA) models. LSTM networks excel in capturing long-term dependencies within sequential data, while ARIMA models provide precise short-term forecasts by modelling trends and seasonality. By integrating these techniques, this approach aims to enhance predictive accuracy and refine trading strategies. Through empirical analysis and case studies, we illustrate the effectiveness of the LSTM-ARIMA integration in improving market prediction and trading outcomes. This report contributes to the advancement of algorithmic trading methodologies by showcasing the potential of LSTM-ARIMA fusion to provide traders with actionable insights, ultimately enabling them to make more informed decisions and achieve better results in today's dynamic financial markets. By leveraging the strengths of both LSTM and ARIMA, this approach offers a comprehensive solution for addressing the challenges of stock market prediction and trading strategy development in the digital age.

3.2. ARCHITECTURE OF THE PROPOSED SYSTEM

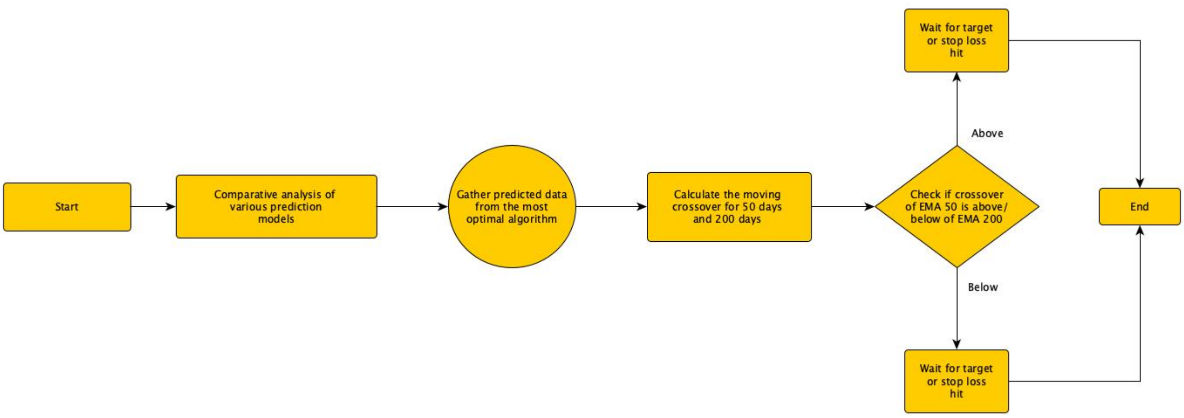

Fig 3.1. Architecture of the proposed system

- **Comparative analysis of various prediction models:** Comparative analysis of various prediction models, including Long Short-Term Memory (LSTM), Linear Regression and K-nearest neighbours. This step aims to assess the performance and accuracy of each model in predicting stock market prices.
- **Data gathering and Prediction:** Following the comparative analysis, the next step involves gathering the predicted data using the selected models. This data is derived from historical stock market data and is used to forecast future stock prices.
- **Calculation of Monthly and Yearly Moving Averages:** Once the predicted data is obtained, we calculate the monthly and yearly moving averages over this data. These moving averages provide insights into the overall trends and patterns in the stock market over different time horizons.
- **Identification of Buy and Sell points:** Based on the calculated moving averages, we identify the buy and sell points using crossover techniques. Specifically, the model looks for instances where the short-term moving average crosses above the long-term moving average (indicating a buy signal) or falls below it (indicating a sell signal). These buy and sell points serve as actionable insights for traders, informing their trading decisions and strategies in the stock market.

3.3. DATASET DESCRIPTION

`yfinance` is a Python library that provides a convenient interface for accessing historical market data, including stock prices, dividends, and corporate actions, from Yahoo Finance. It allows users to easily retrieve financial data for a wide range of publicly traded companies and indices listed on various stock exchanges worldwide.

With `yfinance`, users can retrieve historical market data for specific stocks or indices by specifying the ticker symbol and the desired time period. The library supports fetching data at different intervals, such as daily, weekly, or monthly, and provides functionalities for adjusting the data for stock splits and dividends.

`yfinance` simplifies the process of accessing and analysing historical financial data, making it a popular choice among developers and data analysts for building trading algorithms, conducting market research, and performing quantitative analysis in the field of finance and investment.

Date	Open	High	Low	Close	Adj Close	Volume
2012-03-12	759.429993	760.619995	756.840027	759.130005	759.130005	0
2012-03-13	761.450012	773.030029	761.450012	772.979980	772.979980	0
2012-03-14	773.190002	774.679993	769.280029	771.630005	771.630005	0
2012-03-15	772.380005	776.469971	770.739990	776.419983	776.419983	0
2012-03-16	776.559998	778.099976	775.640015	777.130005	777.130005	0
2012-03-19	777.119995	782.549988	776.070007	780.090027	780.090027	0
2012-03-20	778.030029	778.299988	773.119995	777.369995	777.369995	0
2012-03-21	777.559998	778.650024	774.760010	776.150024	776.150024	0
2012-03-22	773.640015	773.640015	768.000000	770.270020	770.270020	0
2012-03-23	770.650024	774.010010	766.869995	772.929993	772.929993	0

Fig 3.2 Overview of the dataset

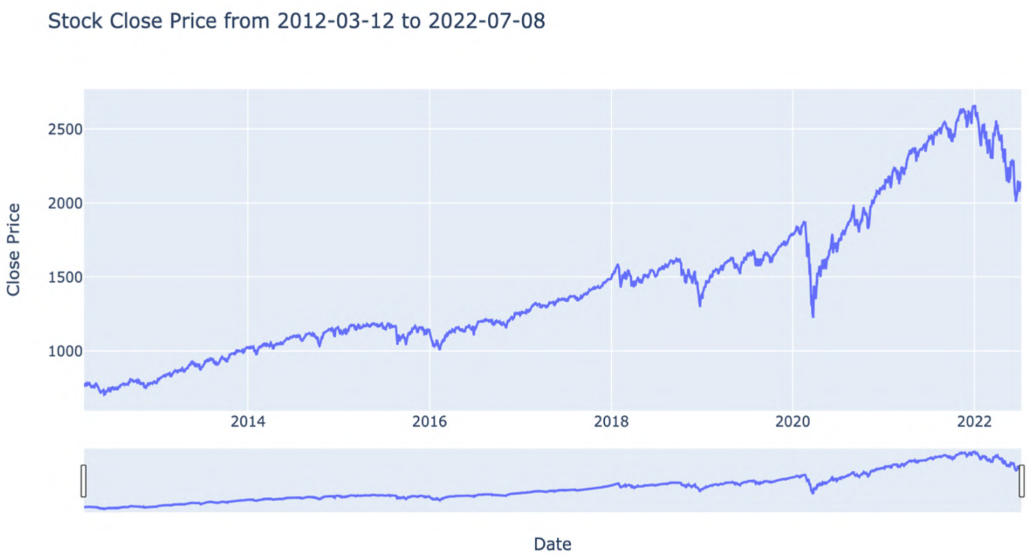

Fig 3.3 The Dataset visualised using Python

4. PROPOSED SYSTEM ANALYSIS AND DESIGN

4.1. COMPARATIVE ANALYSIS

To find the best suitable prediction model for stock price prediction which is a time series model, we use algorithms like LSTM (Long-short term memory), KNN (K-Nearest Neighbour), and Linear Regression.

4.1.1. K-Nearest Neighbour (KNN)

Using KNN for stock market prediction involves collecting historical stock data, selecting relevant features, pre-processing data, training the model, evaluating its performance, tuning parameters, and making predictions. Continuous monitoring and updating are essential due to the dynamic nature of stock markets.

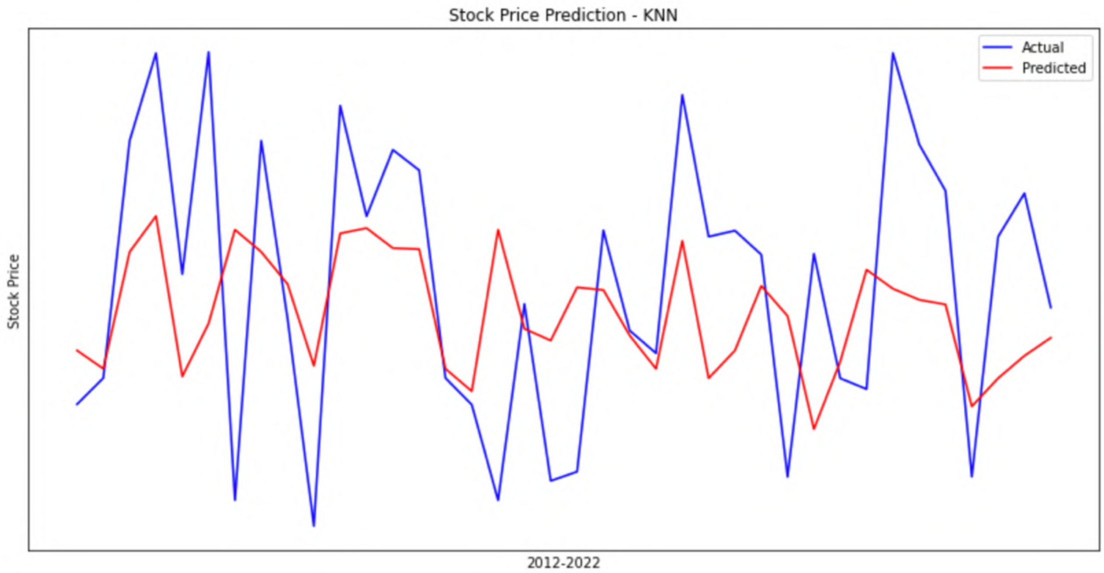

Fig 4.1. Predicted vs Actual values for KNN

4.1.2 Linear Regression

Using linear regression for stock prediction entails fitting a linear model to historical stock market data to forecast future prices. Steps include data collection, feature selection, model training, evaluation, and prediction. Continuous monitoring and adjustment are vital due to market volatility and changing conditions.

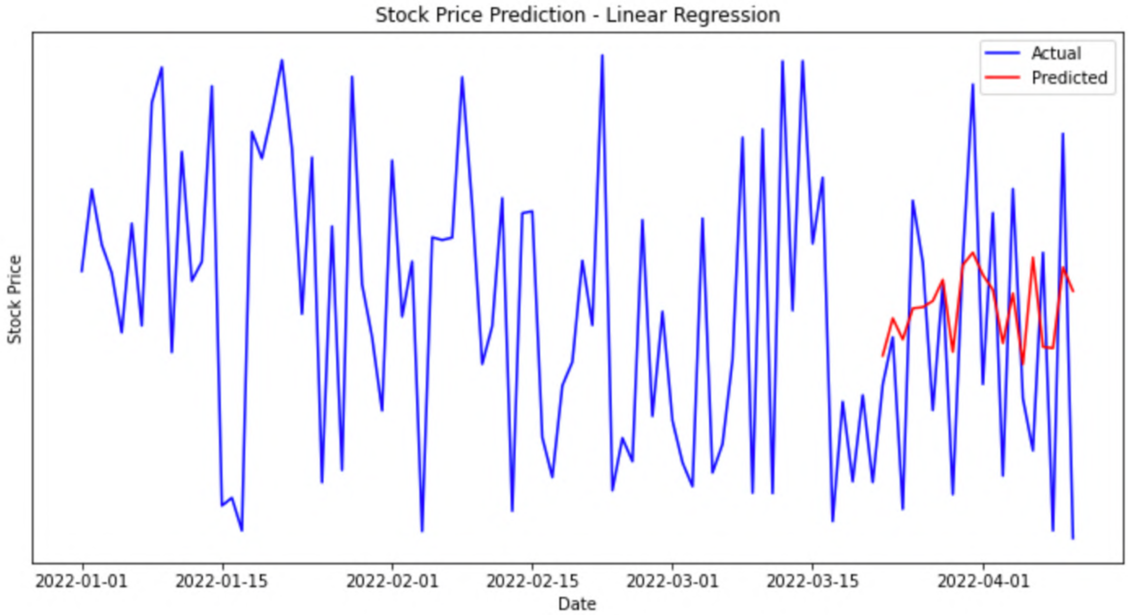

Fig 4.2. Predicted vs Actual Values for Linear Regression

4.1.3. Long-Short Term Memory (LSTM)

LSTM (Long Short-Term Memory) is a recurrent neural network (RNN) architecture widely used in Deep Learning. It excels at capturing long-term dependencies, making it ideal for sequence prediction tasks.
Unlike traditional neural networks, LSTM incorporates feedback connections, allowing it to process entire sequences of data, not just individual data points. This makes it highly effective in understanding and predicting patterns in sequential data like time series, text, and speech.

Characteristics that make LSTM suitable for modelling and predicting stock market movements:

1. Capturing long-term dependencies in sequential data.
2. Handling sequential data inherent in stock price movements.
3. Utilizing memory cells to store and filter relevant historical information.
4. Automatically learning relevant features from the input data.

5. Offering flexibility in model architecture to incorporate additional features.
6. Capturing nonlinear relationships in stock price movements.
7. Predicting multiple steps ahead in a time series.
8. Leveraging large amounts of historical data for training.

Although predicting stock prices remains inherently challenging due to the stochastic nature of financial markets and the presence of various external factors influencing price movements :

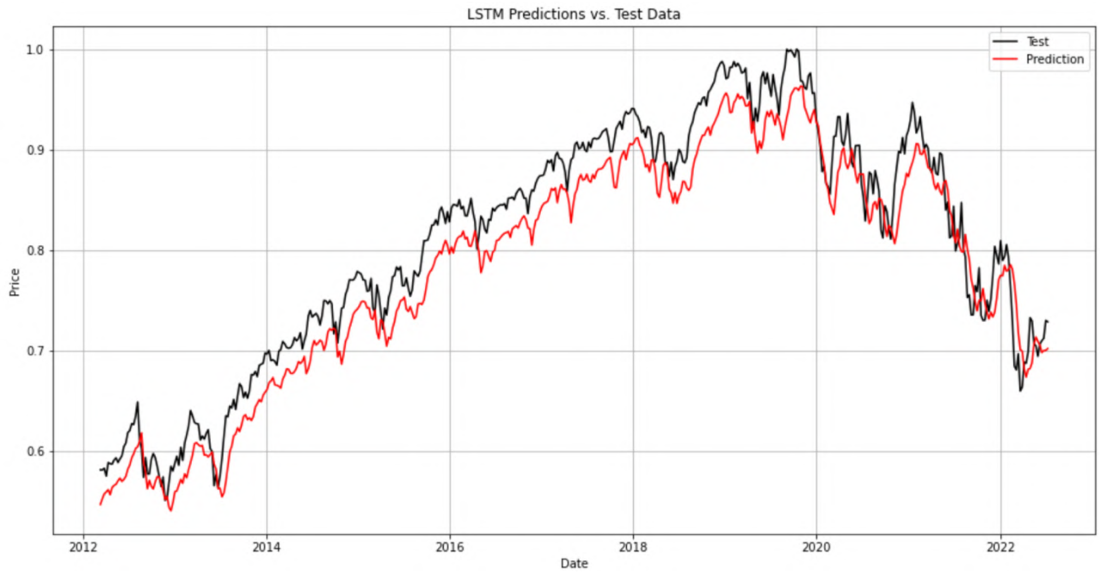

Fig 4.3 Predicted vs Actual Values for LSTM

4.2. MOVING AVERAGE

In finance, a moving average (MA) is a stock indicator commonly used in technical analysis. The reason for calculating the moving average of a stock is to help smooth out the price data by creating a constantly updated average price. By calculating the moving average, the impacts of random, short-term fluctuations on the price of a stock over a specified time frame are mitigated. Simple moving averages (SMAs) use a simple arithmetic average of prices over some timespan, while exponential moving averages (EMAs) place greater weight on more recent prices than older ones over the time period.

The longer the period for the moving average, the greater the lag. A 200-day moving average will have a much greater degree of lag than a 20-day MA because it contains prices for the past 200 days. 50-day and 200-day moving average figures are widely followed by investors and traders and are considered to be important trading signals. Investors may choose different periods of varying lengths to calculate moving averages based on their trading objectives. Shorter moving averages are typically used for short-term trading, while longer-term moving averages are more suited for long-term investors.

Types of crossovers:

- **Bullish Crossover.** This is when the price moves above its moving average. It often suggests that the stock’s momentum is starting to move upward, indicating a potential buying opportunity. When a short-term moving average crosses over a long-term moving average, a golden crossing, a positive technical signal, takes place. The 200-day moving average is often used for the long-term moving average, while the 50-day moving average is typically used for the short-term moving average. The stock price is anticipated to increase in the near future, according to the golden crossover. Let's imagine, for instance, that a stock's 50-day moving average surpasses its 200-day moving average. A bullish indication is said to have been given by this golden crossing. This may be interpreted by traders as a signal that the stock's price will probably climb in the near future

- **Bearish Crossover.** When the stock's price falls below its moving average, it’s generally considered bearish. It tells you that the stock has lost bullish momentum and is either pulling back to take a breather or entering a downtrend. If the latter, it suggests a selling or short-selling opportunity. A death crossing, a bad technical indication, happens when a short-term moving average crosses below a long-term moving average. The short-term moving average is often calculated using the 50-day moving average, while the long-term moving average is frequently calculated using the 200-day moving average. The stock price is anticipated to decline even lower in the near future, according to the death crossing. Let's say a stock's 50-day moving average falls below its 200-

day moving average for the sake of demonstration. This intersection is considered to be a bearish death sign. Trading professionals can perceive this as a signal that the stock's price will likely fall soon.

Applying this theory to our findings, using 2 averages – 365-day average (yearly) and 30-day average (monthly).

ARIMA (Autoregressive Integrated Moving Average)

Using ARIMA algorithm to find the moving averages and finding the points of intersection.
An autoregressive integrated moving average model is a form of regression analysis that gauges the strength of one dependent variable relative to other changing variables. The model's goal is to predict future securities or financial market moves by examining the differences between values in the series instead of through actual values.

An ARIMA model can be understood by outlining each of its components as follows:

- Autoregression (AR): refers to a model that shows a changing variable that regresses on its own lagged, or prior, values.
- Integrated (I): represents the differencing of raw observations to allow the time series to become stationary (i.e., data values are replaced by the difference between the data values and the previous values).
- Moving average (MA): incorporates the dependency between an observation and a residual error from a moving average model applied to lagged observations.

5. CODE EXTRACTS

LSTM model:

```
np.random.seed(10)

lstm_input = Input(shape=(backcandles, 8), name='lstm_input')
inputs = LSTM(150, name='first_layer')(lstm_input)
inputs = Dense(1, name='dense_layer')(inputs)
output = Activation('linear', name='output')(inputs)
model = Model(inputs=lstm_input, outputs=output)
adam = optimizers.Adam()
model.compile(optimizer=adam, loss='mse')
model.fit(x=X_train, y=y_train, batch_size=15, epochs=30, shuffle=True,
validation_split = 0.1)
```

ARIMA:

```
# Create traces for LSTM predictions, ARIMA monthly moving average, buy points, and
sell points
trace_lstm = go.Scatter(x=y_pred_lstm_series_subset.index,
y=y_pred_lstm_series_subset.values, mode='lines', name='LSTM Predictions',
line=dict(color='red'))
trace_arima_monthly = go.Scatter(x=arima_forecast_monthly_subset.index,
y=arima_forecast_monthly_subset.values, mode='lines', name='ARIMA Monthly Moving
Average', line=dict(color='blue'))
trace_arima_yearly = go.Scatter(x=arima_forecast_yearly_subset.index,
y=arima_forecast_yearly_subset.values, mode='lines', name='ARIMA Yearly Moving
Average', line=dict(color='green'))
trace_buy_points = go.Scatter(x=buy_points,
y=arima_forecast_monthly_subset.loc[buy_points], mode='markers', name='Buy Points',
marker=dict(color='green', symbol='triangle-up', size=10))
trace_sell_points = go.Scatter(x=sell_points,
y=arima_forecast_monthly_subset.loc[sell_points], mode='markers', name='Sell
Points', marker=dict(color='red', symbol='triangle-down', size=10))

# Create layout
layout = go.Layout(
    title='LSTM Predictions with ARIMA Moving Averages (Monthly and Yearly) with
Buy/Sell Points from 2014 to 2022',
    xaxis=dict(title='Date'),
    yaxis=dict(title='Price')
)

# Find intersection points
intersection_points = []
for idx in range(1, len(arima_forecast_monthly_subset)):
    if idx < len(arima_forecast_yearly_subset):
```

```
        if arima_forecast_monthly_subset[idx-1] < arima_forecast_yearly_subset[idx-
1] and arima_forecast_monthly_subset[idx] > arima_forecast_yearly_subset[idx]:
            intersection_points.append(idx)
```

6. RESULT AND DISCUSSION

Conducting a comparative analysis between various prediction models to find the most optimal algorithm for a time series calculation and prediction like stock market analysis. From Table 6.1. analysing the values, a lower MAE and RMSE values indicate better accuracy, while a higher R-squared value (closer to 1) indicates better fit of the model to the data. Hence it can be seen that LSTM proves to be the most optimal solution for a prediction model for stock market.

Table 6.1. Comparative Analysis between various prediction models

	KNN	Linear Regression	LSTM
Root Mean Squared Error (RMSE)	0.23079	0.23079	0.02312
Mean Absolute Error (MAE)	0.19129	0.18903	0.01876
R-squared (R2)	0.00826	0.00826	0.96269

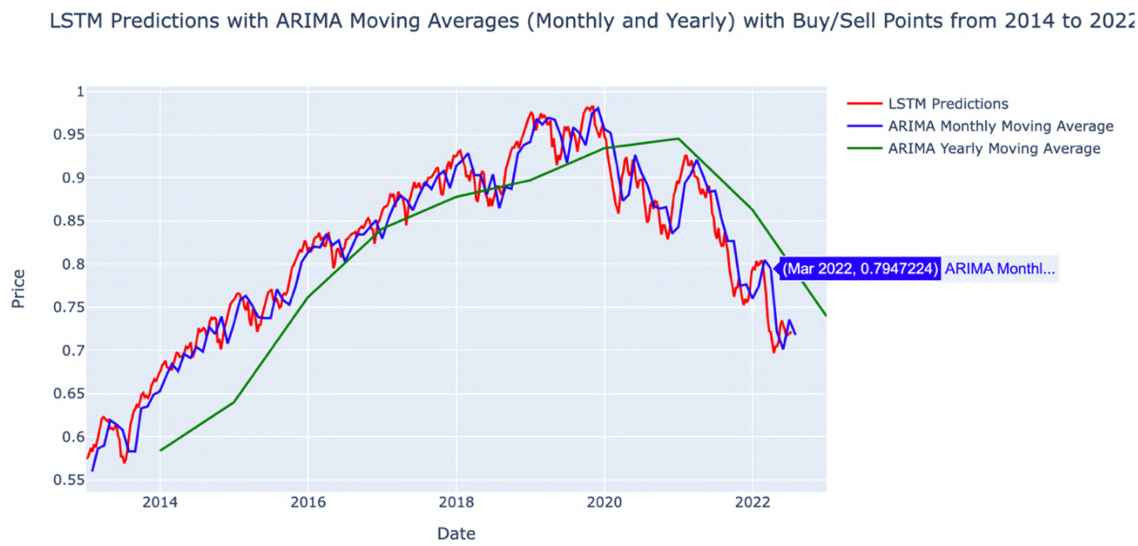

Fig 6.1. Moving Average – Monthly and Yearly time periods

Identifying the buy and sell points:

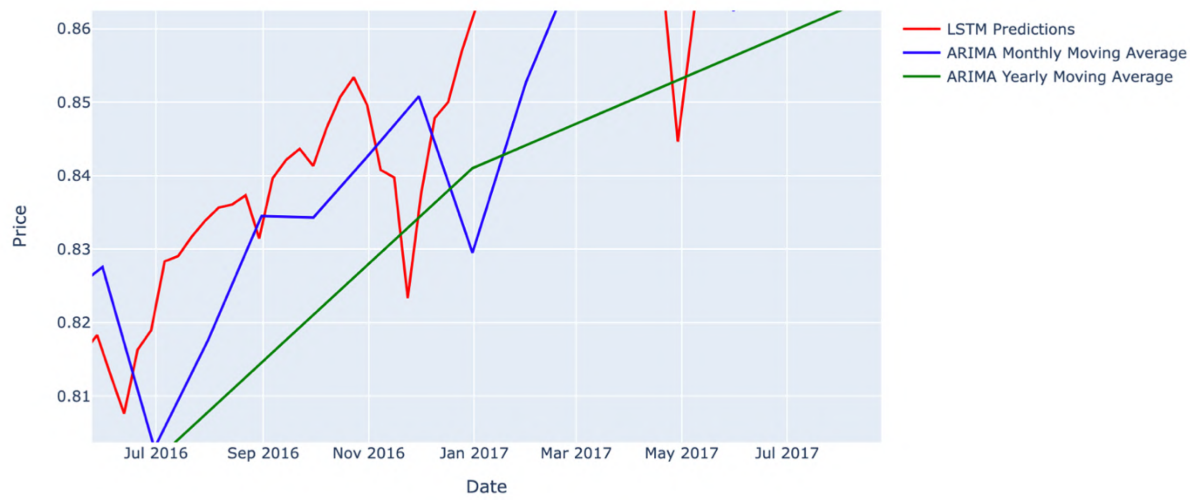

Fig 6.2. Bearish Crossover

The first intersection appears at December, 2016, where the monthly moving average moves below the yearly average indicating a death cross (bearish crossover).
This gives a ‘sell’ signal.

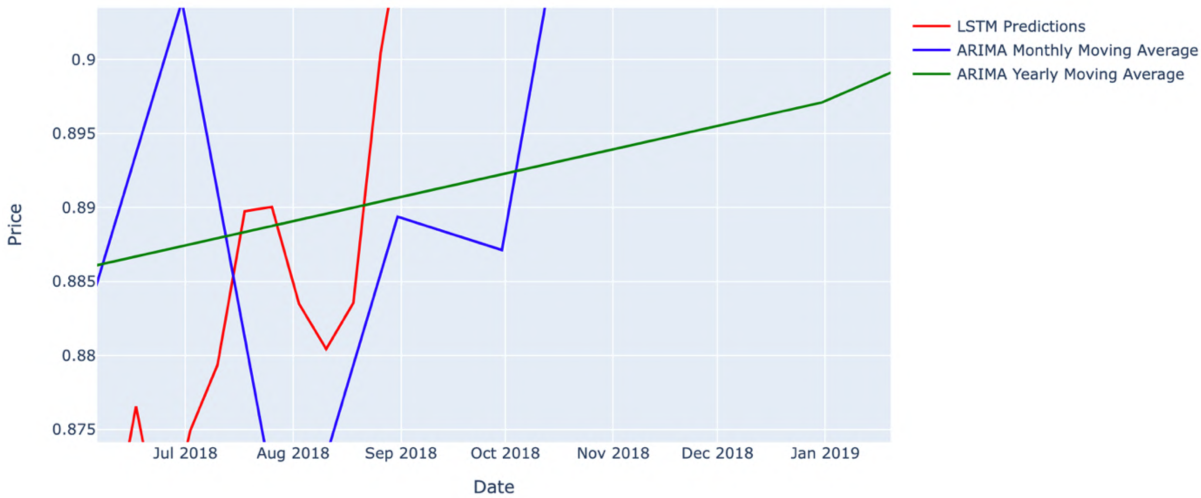

Fig 6.3. Bullish Crossover

Another intersection occurs at October 2018, where the monthly moving average moves above the yearly average indicating a golden cross (bullish crossover).
This gives a 'buy' signal

7. CONCLUSION

In conclusion, the application of LSTM and ARIMA models for stock market prediction offers valuable insights into market dynamics and trends. The LSTM model excels in capturing long-term dependencies and intricate patterns, while the ARIMA model effectively models short-term fluctuations and seasonality. By leveraging the strengths of both models, traders can gain a more comprehensive understanding of market behaviour and make more informed trading decisions.

The results obtained from LSTM and ARIMA models underscore the importance of using diverse modelling techniques to navigate the complexities of financial markets. While each model has its strengths and limitations, their integration can provide a more robust framework for predicting stock prices and identifying optimal trading strategies.

Moving forward, further research and development are needed to refine and enhance the predictive accuracy of LSTM and ARIMA models. Additionally, exploring the integration of other advanced machine learning techniques and incorporating domain knowledge could further improve the performance of stock market prediction models.

Overall, the findings from this study contribute to advancing our understanding of stock market prediction and provide valuable insights for traders looking to navigate volatile market conditions and maximize returns in the stock market. By leveraging the predictive power of LSTM and ARIMA models, traders can enhance their decision-making processes and capitalize on opportunities in the ever-changing landscape of financial markets.

8. REFERENCES

1. Gurrib, I. (2016). The Moving Average Crossover Strategy: Does It Work for the S&P500 Market Index?. *Gurrib, I.(2016), Optimization of the Double Crossover Strategy for the S&P500 Market Index, Global Review of Accounting and Finance*, *7*(1), 92-107.

2. Sattarov, O., Muminov, A., Lee, C. W., Kang, H. K., Oh, R., Ahn, J., ... & Jeon, H. S. (2020). Recommending cryptocurrency trading points with deep reinforcement learning approach. *Applied Sciences*, *10*(4), 1506.

3. Adegboye, A., & Kampouridis, M. (2021). Machine learning classification and regression models for predicting directional changes trend reversal in FX markets. *Expert Systems with Applications*, *173*, 114645.

4. Tapa, A., Yaacob, M. H., Hamzah, A. H., & Yean, S. C. (2018). Trading performance analysis: a comparisons between the original MA crossover and modified MA crossover strategy. *The Journal of Social Sciences Research*, (SPI6), 933-941.

5. Thakkar, A., & Chaudhari, K. (2020). CREST: cross-reference to exchange-based stock trend prediction using long short-term memory. *Procedia Computer Science*, *167*, 616-625.

6. Thakkar, A., & Chaudhari, K. (2022). Information fusion-based genetic algorithm with long short-term memory for stock price and trend prediction. *Applied Soft Computing*, *128*, 109428.

7. Yadav, A., Jha, C. K., & Sharan, A. (2020). Optimizing LSTM for time series prediction in Indian stock market. *Procedia Computer Science*, *167*, 2091-2100.

8. Vijh, M., Chandola, D., Tikkiwal, V. A., & Kumar, A. (2020). Stock closing price prediction using machine learning techniques. *Procedia computer science*, *167*, 599-606.

9. Long, W., Lu, Z., & Cui, L. (2019). Deep learning-based feature engineering for stock price movement prediction. *Knowledge-Based Systems*, *164*, 163-173.

10. Ryś, P., & Ślepaczuk, R. (2018). Machine Learning Methods in Algorithmic Trading Strategy Optimization–Design and Time Efficiency. *Central European Economic Journal*, *5*(52), 206-229.

11. Attia, J. (2019). Evaluating the Effectiveness of Common Technical Trading Models. *arXiv preprint arXiv:1907.10407*.

12. Fu, X., Du, J., Guo, Y., Liu, M., Dong, T., & Duan, X. (2018). A machine learning framework for stock selection. *arXiv preprint arXiv:1806.01743*.

13. El-Khodary, I. A. (2009). A decision support system for technical analysis of financial markets based on the moving average crossover. *World Applied Sciences Journal*, *6*(11), 1457-1472.

14. Huang, X., Wu, C., Du, X., Wang, H., & Ye, M. (2024). A novel stock trading utilizing long short term memory prediction and evolutionary operating-weights strategy. *Expert Systems with Applications*, *246*, 123146.

15. Das, P., Füss, R., Hanle, B., & Russ, I. N. (2020). The cross-over effect of irrational sentiments in housing, commercial property, and stock markets. *Journal of banking & Finance*, *114*, 105799.