International Journal of Research Publication and Reviews, Vol 4, no 6, pp 656-662 June 2023

WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Stock Price Prediction using Moving Average Time Series

Dr. P. H. Zope

Department of Computer Engineering
SSBT's COET, Bambhori, Jalgaon, KBC N.M.U. Jalgaon
phzope@gmail.com

ABSTRACT

The moving average method is widely used in time-series forecasting, it is one of widely known technical indicator used to predict the stock market future data in time series analysis. During its' development, many variation and implementation have been made by researchers. One of its' widely used variation is Exponential Moving Average (EMA). Basically, EMA is an improvement of Weighted Moving Average (WMA) that gives a special weighting to more recent data than the older data, which could not be found in Simple Moving Average (SMA) method. This paper aims to introduce comparative approach of moving average method in time series analysis.

Keyword:- Moving average, weighted moving average, Machine learning, Stock, Trading, Trend, time series analysis etc.

I. Introduction

Time-series data analysis method is becoming very important in many industries like financial industries, pharmaceuticals, social media companies, web service providers, research, and many more. Some of them are using Simple Moving Average (SMA), Exponential Moving Average (EMA), Weighted Moving Average (WMA) etc[1-3]. Some others are using soft computing methods such as neural network, fuzzy interface system [4-8] to predict their future values of a given time series data set. Some other even combine and develop hybrid forecasting methods, such as nero-fuzzy, wavelet-neural networks and fuzzy-wavelet method[9-11].

In this study the comparative analysis of the moving average methods is discussed by comparing their performance.

II. Moving Average Methods

A. Simple Moving Average (SMA) A Simple Moving Average (SMA) is a common average of the previous n data points in time series data. Each point in the time series data is equally weighted, so there are no weighting factors applied to any of the data points. A simple moving average computes the mean of the past N data points and takes this value as the predicted N+1 value [12].

$$SMA = \frac{P_M + P_{M-1} + \cdots + P_{M-(n-1)}}{n} \tag{1}$$

Where $P_{M,}$ $P_{M-(n-1)}$ and n are immediate data points that occur before the present, to predict the present data point, the size of SMA is n. The SMA is our predicted value. The precision of the model will vary significantly with the choice of n. Higher n would mean that we are willing to go deeper into the past to compute the present value.

$$SMA_{Today} = \frac{P_M}{n} + SMA_{Yesterday} - \frac{P_{M-n}}{n} \tag{2}$$

B. Weighted Moving Average (WMA) A Weighted Moving Average (WMA) is an improvement form of SMA. It gives a greater weight to more recent data than the older ones. The weighting factors are calculated from the sum of days used in time series data, also known as the 'sum of digits '[13].

$$WMA = \frac{nP_M + (n-1)P_{M-1} + \cdots + 2P_{(M-n+2)} + P_{(M-n+1)}}{n + (n-1) + \cdots + 2 + 1} \tag{3}$$

C. Exponential Moving Average (EMA) An Exponential Moving Average (EMA) is a type of WMA which assigns a weighting factor to each value in the data series according to its age. Like WMA, in EMA the most recent data gets the greatest weight and each data value gets a smaller weight as we go back chronologically. But unlike WMA, in EMA the weighting for each older data point decreases exponentially, so its' never reaching zero [14]. The EMA for a time series can be calculated recursively as

$$S_1 = Y_1, \qquad (4)$$

$$\text{for } t > 1, S_t = \alpha \cdot Y_t + (1 - \alpha) \cdot S_{t-1},$$

where S_t is the value at time period , Y_t is the value of at time period , and represents the degree of weighting decrease, a constant smoothing factor between 0 and 1. Commonly, is calculated using the formula

$$\alpha = 2/(n + 1) \qquad (5)$$

D. Weighted Exponential Moving Average (WEMA) A new approach is introduced by combining the calculation of weighting factors for WMA and EMA. We will call this approach as Weighted Exponential Moving Average (WEMA) for easiness. First, we will use the WMA formula to get the new predicted value for a point in time series data using the 'sum of digits' weighting factor. Then, the new value will not be used as the forecasting result value, but as a base value to calculate with EMA weighting factors [15]. Here is the algorithm's procedure

1. Calculate the base value, ,H_t using equation (3), for a given time series data and periods.

2. Using the base value obtained, calculate the forecasting value using formula

$$WEMA_t = \alpha \cdot Y_t + (1 - \alpha) \cdot H_t \qquad (6)$$

where Y_t is the value at time period , H_t is the base value for a time period , and represents the degree of weighting decrease as in equation (5). 3. Back to step (1) until each data point in the time periods given ended. The main difference in this approach than EMA approach is that in EMA only the most recent data and one last ordered data will be used in the calculation; while in this new approach not only one last ordered data will be used, but some older data values in a given time periods will be considered.

E. Mean Square Error (MSE) Mean Square Error (MSE) denotes the average of the square of error sum between the forecasted data and the actual data. The formula can be written as follows,

$$MSE = \frac{\sum_{t=1}^{n} e_t^2}{n} \qquad (7)$$

where n denotes the number of data, and e_t denotes the forecasting error from X_t - $\ddot{X}_t$. Here, X_t is the actual data and $\ddot{X}_t$ is the forecasted data [16].

F. Mean Absolute Percentage Error (MAPE) Mean Absolute Percentage Error (MAPE) value gives us an indication about how much the average of absolute error of the forecasted data compare to the actual data, and denotes by the formula,

$$MAPE = \frac{\sum_{t=1}^{n} \left|\frac{e_t}{X_t}\right|}{n} \times 100, \qquad (8)$$

where n denotes the numbers of data, and et denotes the forecasting error from Xt - Ẍt . The actual data is denoted by Xt , while Ẍt is the forecasted data.

III. Moving Average in a Pandas Data Frame

In Python, calculate the moving average using .rolling() method. This method provides rolling windows over the data, and we can use the mean function over these windows to calculate moving averages. The size of the window is passed as a parameter in the function .rolling(window).

Step 1: Importing Libraries

```
# importing Libraries

# importing pandas as pd

import pandas as pd

# importing numpy as np

# for Mathematical calculations

import numpy as np

# importing pyplot from matplotlib as plt

# for plotting graphs

import matplotlib.pyplot as plt
```

```
plt.style.use('default')

%matplotlib inline
```

Step 2: Importing Data

```
# importing time-series data

reliance = pd.read_csv('RELIANCE.NS.csv', index_col='Date', parse_dates=True)

# Printing dataFrame

reliance.head()
```

Date	Open	High	Low	Close	Adj Close	Volume
2016-07-21	507.192322	507.687622	497.781525	498.450195	485.688568	3097170.0
2016-07-22	498.276825	504.121429	497.831055	502.660278	489.790863	3184126.0
2016-07-25	502.239258	507.068512	498.945496	506.573181	493.603546	3803482.0
2016-07-26	506.697021	512.541626	502.833649	507.217102	494.231049	4853316.0
2016-07-27	506.721771	508.182922	499.737976	501.793488	488.946259	5090697.0
2016-07-28	502.709808	510.411804	501.768738	508.356293	495.341064	7204399.0
2016-07-29	507.192322	507.638092	501.248657	502.685059	489.815002	4444057.0
2016-08-01	504.220490	505.359680	497.806305	500.010406	487.208862	3681961.0
2016-08-02	499.267456	505.211090	498.425415	501.644897	488.801483	3558227.0
2016-08-03	502.437408	502.437408	491.094910	492.308411	479.704041	5479056.0

Fig.1 dataframe output

Step 3: Calculating Simple Moving Average

To calculate SMA in Python we will use Pandas dataframe.rolling() function that helps us to make calculations on a rolling window. On the rolling window, we will use .mean() function to calculate the mean of each window.

Syntax: DataFrame.rolling(window, min_periods=None, center=False, win_type=None, on=None, axis=0).mean()

```
# updating our dataFrame to have only one column 'Close' as rest all columns are of no use for us at

# the moment using .to_frame() to convert pandas series into dataframe.

reliance = reliance['Close'].to_frame()

# calculating simple moving average using .rolling(window).mean() , with window size = 30

reliance['SMA30'] = reliance['Close'].rolling(30).mean()

# removing all the NULL values using dropna() method

reliance.dropna(inplace=True)

# printing Dataframe

Reliance
```

Date	Close	SMA30
2016-09-01	509.767914	505.689072
2016-09-02	501.917328	505.804643
2016-09-06	505.161560	505.888019
2016-09-07	504.047150	505.803818
2016-09-08	511.848175	505.958187
...	...	...
2021-07-14	2086.000000	2160.583325
2021-07-15	2082.350098	2156.339998
2021-07-16	2112.399902	2153.736662
2021-07-19	2098.949951	2149.454997
2021-07-20	2093.800049	2145.428328

1145 rows × 2 columns

Fig 2. Moving average calculation

IV. Result and Discussion

The statistics and chart indicates Simple Moving Average just calculates the average value by performing a mean operation on given data but it changes from interval to interval. But whereas in Exponential Moving Average also uses Simple Mean Average in calculating its average but gives more weightage to the newly added value as the latest value has more weightage. Fig 3 shows the simple moving average

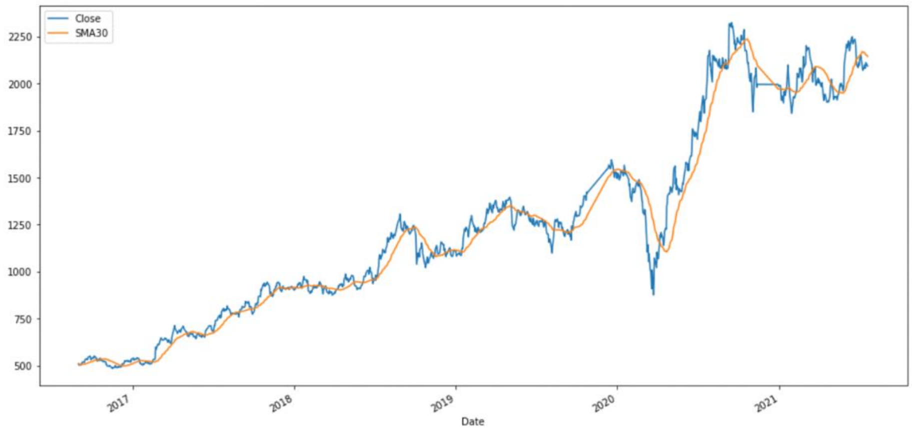

Fig 3. Simple moving average

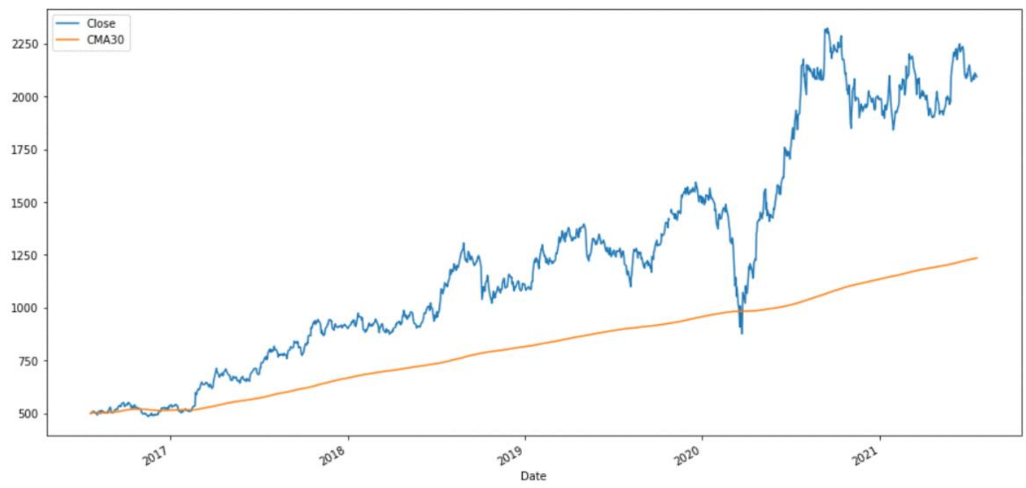

Fig 4. Exponential moving average

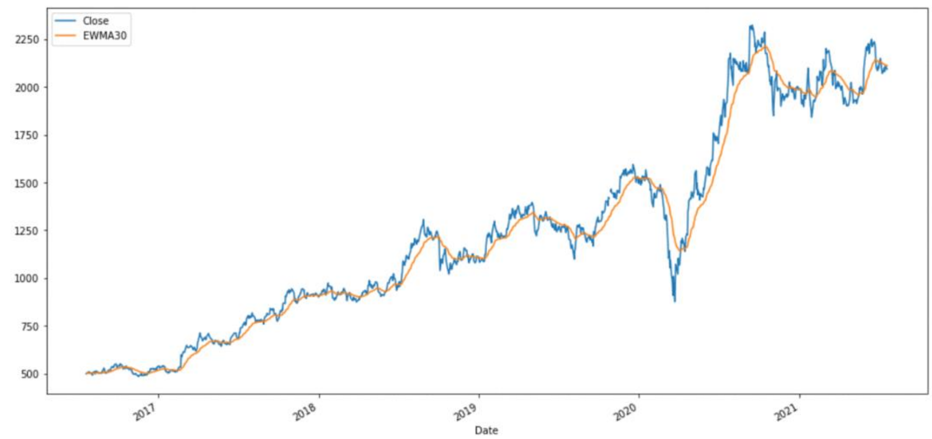

Fig 5. Cumulative moving average

EMA values is little smoothened when compared to Original Stock values indicates the nature of Exponential Moving Averages.

```
# import necessary packages
import pandas as pd
import matplotlib.pyplot as plt

# create a dataframe
stockValues = pd.DataFrame(
        {'Stock_Values': [60, 102, 103, 104, 101,105, 102, 103, 103, 102]})

# finding EMA
# use any constant value that results in
# good smoothened curve
ema = stockValues.ewm(com=0.4).mean()

# Comparison plot between stock values & EMA
```

```
plt.plot(stockValues, label="Stock Values")

plt.plot(ema, label="EMA Values")

plt.xlabel("Days")

plt.ylabel("Price")

plt.legend()

plt.show()
```

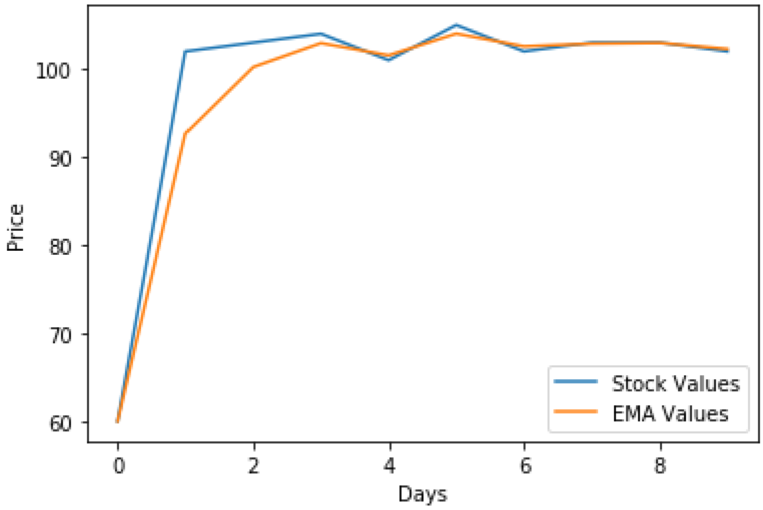

Fig 6. Comparative plot between stock values and EMA

V. Conclusion

In this study the experimental work is carried out on the SMA and EMA. The comparative analysis is represented using cumulative moving average. It is observed that EMA is best moving average method as compared to SMA for the presented financial time series data. To support the performance of this moving average the stochastic regression method is used to observe the good quality approach.

VI. References

1. C. Wang, "Quantitative Analysis on the Bullwhip Effect in a Supply Chain using Double Moving Average and Double Exponential Smoothing Forecast", Proc. of the International Symposiums on Information Processing (ISIP), pp. 114-118, May 2008.
2. F. Klinker, "Exponential Moving Average versus Moving Exponential Average", Math. Semesterber, vol. 58, no. 1, pp. 97-107, 2011.
3. F. Papailias and D. D. Thomakos, "Improved Moving Average (IMA) Strategies", Market Technician: Journal of the Society of Technical Analysis, vol. 72, pp. 12-17, 2012.
4. S-C. Kuo, C-C. Chen, H-Y. Chen, S-T. Li and H-J. Wang, "A Steady-State Probabilities Model for Fuzzy Time Series Forecasting", Proc. of the 5th IIAI International Congress on advanced Applied Informatics (IIAI-AAI), pp. 615-619, July 2016.
5. M. Dong, "Forecasting Language Test Performance with a Back Propagation Neural Network Model", Proc. of the 11th International Conference on Natural Computation (ICNC), pp. 813-819, August 2015.
6. S. Hansun, "Peramalan Data IHSG Menggunakan Fuzzy Time Series", Indonesian Journal of Computing and Cybernetic Systems (IJCCS), vol. 6, no. 2, pp. 79-88, 2012.
7. J. Faraway and C. Chatfield, "Time Series Forecasting with Neural Networks: A Comparative Study using the Airline Data", Journal of the Royal Statistical Society: Series C (Applied Statistics), vol. 47, no. 2, pp. 231-250, 1998.
8. B. Oancea and S. C. Ciucu, "Time Series Forecasting using Neural Networks", Proc. of 2013 International Conference on Challenges of the Knowledge Society, pp. 1402-1408, 2013.
9. Z. Zhiyuan, Z. Yanning, Z. Xiaoyuan and J. Yamin, "Research on the Evaluation of Enterprise Competitiveness based on the Wavelet Neural Network Forecasting System", Proc. of the 2016 Chinese Control and Decision Conference (CCDC), pp. 2674-2677, May 2016.

10. A. Draidi and D. Labed, "A Neuro-fuzzy Approach for Predicting Load Peak Profile", International Journal of Electrical and Computer Engineering (IJECE), vol. 5, no. 6, pp. 1304-1310, Dec 2015.
11. A. Popoola, S. Ahmad and K. Ahmad, "A Fuzzy-Wavelet Method for Analyzing Non-Stationary Time Series", Proc. of the 5th International Conference on Recent Advances in Soft Computing RASC 2004, pp. 231-236, 2004.
12. Yaxi Su, "Self-Attentive Moving Average for Time Series Prediction", Appl. Sci. 2022, 12(7), 3602; https://doi.org/10.3390/app12073602.
13. Ma, S. A hybrid deep meta-ensemble networks with application in electric utility industry load forecasting. Inf. Sci. 2021, 544, 183–196.
14. Karevan, Z.; Suykens, J.A. Transductive LSTM for time-series prediction: An application to weather forecasting. Neural Netw. 2020, 125, 1–9.
15. Cui, C.; Li, X.; Du, J.; Zhang, C.; Nie, X.; Wang, M.; Yin, Y. Temporal-Relational Hypergraph Tri-Attention Networks for Stock Trend Prediction. arXiv 2021, arXiv:2107.14033.
16. Gajbhiye, A.; Winterbottom, T.; Al Moubayed, N.; Bradley, S. Bilinear fusion of Ccommonsense knowledge with attention-based NLI models. In International Conference on Artificial Neural Networks; Springer: Berlin/Heidelberg, Germany, 2020; pp. 633–646.
17. P. H. Zope, "Stock Price Prediction using Machine Learning in Python", INTERNATIONAL JOURNAL OF SCIENCE, SPIRITUALITY, BUSINESS AND TECHNOLOGY (IJSSBT), Vol-8, No-2, 2022