RESEARCH ARTICLE

DOI: 10.54392/irjmt26318

Asian Research Association

INTERNATIONAL RESEARCH JOURNAL OF
MULTIDISCIPLINARY TECHNOVATION

Multi-Lag Smoothing Techniques for Stock Market Time-Series Forecasting and Performance Evaluation

Milin Patel [a, *], Sweta S. Panchal [b], Neha Soni [c], Sandipkumar R. Panchal [b]

[a] Department of Computer Engineering, Dr. Subhash University, Junagadh - 362001, India
[b] Department of Electronics and Communication Engineering, Dr. Subhash University, Junagadh - 362001, India
[c] Department of Computer Engineering, Sardar Vallabhbhai Patel Institute of Technology, Vasad - 388306, India
* Corresponding Author Email: milin2784@gmail.com

DOI: https://doi.org/10.54392/irjmt26318
Received: 28-02-2026; Revised: 22-04-2026; Accepted: 06-05-2026; Published: 21-05-2026

Abstract: Financial time series are nonlinear, non-stationary and highly contaminated with high-frequency noise thus they are difficult to forecast accurately. Although advanced machine learning models have shown a high predictive accuracy, very little has been done on systematic signal conditioning before making a prediction. This paper suggests a systematic multi-lag smoothing model of predicting time-series of stock markets with six moving averages, namely, SMA, EMA, WMA, DEMA, HMA, and AMA. The filter properties of these methods are measured in time and frequency domain with different lag windows (5-100 days). A baseline model that is done using regression is used to isolate the effect of smoothing on predictive accuracy. MSE, MAPE, R2, and Directional Accuracy are used to determine performance. Comparison of performance on benchmark indices (NIFTY 50 and S&P 500) through cross-dataset validation shows that HMA demonstrates consistently lower error in short- and medium-term forecasting, while AMA shows improved performance in long-term trend modeling. Statistical significance is formally validated for HMA versus SMA at the 25-day lag, while other observations are based on comparative empirical results rather than formal hypothesis testing. Paired t-tests are statistically confirmed to be significant ($p < 0.05$), but moderate effect sizes (Cohen d) (0.42–0.55) are significant. The proposed framework unites the classical FIR filter theory and computational financial modeling, which provides a statistically valid and computationally efficient framework applicable in a real-time financial analytics system.

Keywords: Financial Signal Processing, Finite Impulse Response (FIR) Filters, Moving Average Filtering, Frequency-Domain Analysis, Time-Series Forecasting, Lag Sensitivity Analysis, Multi-Objective Performance Evaluation

1. Introduction

Financial markets can be described as complex nonlinear and non-stationary stochastic dynamic structures in which the price development is nonlinear and non-stationary, and noise represents the dominating factor. Scientists and technologists find these characteristics highly problematic for predictive modeling. Stock prices can be perceived as discrete-time signals which are not stationary in a statistical sense and engineering-wise. So, time series analysis of funds involves signal filtering and predictive modelling. Conventional techniques such as ARIMA, regression-based models and neural network-based forecasting techniques have played an important role in financial forecasting during the past several decades [1-4]. Due to the rapid evolution of the issue of machine learning, more advanced forecasting models have become available [5–7].

Nevertheless, preprocessing is quite a significant component of the forecasting pipeline. Speculative trading, market microstructure and macroeconomic shocks are the root causes of high-frequency noise in the financial price series. Signal filtering is therefore essential so as to reduce noise level without distorting the trend underneath. Moving averages are still believed to be one of the most common techniques in technical analysis of linear signal filters, and can be seen as discrete time low-pass filters that diminish the high-frequency content, yet long-term trends [8-10]. The systematical study in the filtering of moving averages in various lag structures has not been carried out. Although the study is based on technical indicators combinations with advanced learning models [11-14], and hybrid signal decomposition techniques to promote predictability, such as lag-dependent filtering behavior, is hardly systematically tested. Signal-processing Moving averages Signal-processing Moving averages can be modeled as discrete-time linear filters, the frequency response of which is used to characterize the extent of noise reduction [15-17].

Filtering performance and window length will not be an easy task to analyze in a systematic manner and derive meaning out of the financial data. Although machine learning models and technical indicators are widespread in financial forecasting, price signals are highly volatile and noisy, and therefore difficult to forecast accurately [18]. Such instability reduces predictive reliability compared to the conventional statistical methods. In addition, the majority of state-of-the-art forecasting models are nonlinear architectures that are cumbersome and do not prioritize signal preprocessing [19]. As a result, they are easily overfitted and may be unstable and fail to extrapolate to uncharted data. Although there are numerous moving average techniques applied in different trading systems, minimal engineering-based examination of their performance in different lag situations and performance indicators has been done. Much of the literature consists of individual models or hybrid configurations, rather than experimental computational studies of filtering behavior in a structured context. The second limitation is that many studies are not cross dataset validated which means that they are researching one market or index at a time [20], which restricts generalization. It is therefore clear that there is the necessity to possess a systematic assessment system to measure filtering performance across different datasets based on responsiveness, stability and direction accuracy.

The reason why the current research was done is to treat financial forecasting as a prediction, and an engineer level signal-processing problem. Models such as LSTM, GRU, ensemble models and hybrid architecture have been demonstrated to offer superior nonlinear modelling potential, however, they are highly susceptible to the signal quality of the input signals. Moving averages can be regarded as financial signal FIR filters (or adaptive smoothing operators). Signal-processing in signal-processing terminology, very little work has been done to explore their behavior in the different lag configurations in a systematic manner. The tradeoff between suppression of noise and phase lag must be carefully balanced to be applicable in forecasting. There is a methodological gap in developing standardized structures to evaluate moving average-based filtering on a range of datasets. The difference can be bridged through integrating the traditional technical analysis with computational modeling techniques in line with the engineering and financial informatics principles.

This study has contributed immensely in terms of surveys:

- Creation of an analytical architecture to model financial time series as discrete time signals with the help of FIR to perform the systematic analysis of moving average filters.
- Multi-lag analysis (5-100 days) of six short-term, medium-term and long-term forecasting moving averages.
- Description of moving average filters in frequency domain, magnitude response of the filters, phase response, and group delay of the filters.
- Robustness validation by cross-dataset validation using Yahoo Finance and NSE official data.
- A multi-objective performance measurement system incorporating MSE, MAPE, R 2, and Directional Accuracy.

The work of this paper together, when combined, will form a repeatable model of financial time series prediction within the engineering setting. The current paper provides a foundation of lag-sensitive, computationally efficient financial forecasting systems based on the generalization of the ideas of discrete time signal processing.

2. Literature Review

2.1 Financial Signal Processing Approaches

There are financial signal processing approaches 2.1 financial signal processing approaches. Methods of financial time-series prediction have evolved slowly out of the earlier econometric models of finance to computational intelligence and signal processing. Early models of finance, primarily in the autoregressive (AR) framework, which represent linear relationships and autocorrelation in price changes. Even though the autoregressive models are statistically and computationally efficient, they are deficient in as far as the nonlinear and non-stationary character of the financial markets is concerned. Signal decomposition methods are also proposed to address these limitations, such as empirical mode decomposition (EMD), wavelet analysis and variational mode decomposition (VMD) which are incorporated into foreseeing systems [21-24]. One such application has been wavelet-based denoising and subsequent regression modelling to minimize stochastic volatility [25]. It has been observed that signal decomposition together with machine learning techniques have demonstrated that hybrid techniques are more effective in improving the strength of forecasts [26-27].

Studies of boosting and threshold autoregressive models have also been undertaken to capture regime shifts and structural changes of financial time series. The new developments clearly indicate that signal-processing-based forecasting models are taking precedence over statistical models [29]. The financial indicator forecasting success has been improved by Deep Learning. The use of deep learning architectures has improved predictive abilities for financial time-series models. LSTM, RNN and GRU proved also to be efficient in modeling long-term temporal behaviors of financial data [30-32]. Additionally, it has introduced mechanisms that employ attention-based approaches and channel spatial architecture that can improve the representation of features within heterogeneous financial data [33–34]. There is hybrid models created between ARIMA and neural networks to take advantage of the benefits from using linear and nonlinear models [35–20]. The development trends have resulted in input quality becoming important and, therefore, processing and filtering methods have gained prominence.

2.2 Moving Average Techniques in Forecasting

Moving averages are one of oldest and popular techniques in financial time series analysis for filtering. The moving average is interpreted, in signal-processing terms, as a discrete time low pass filter that kills the high frequency noise and retains the information regarding the underlying trend [36]. The Simple Moving Average (SMA) applies a uniform weight to the values within a constant observation window. In contrast, the Exponential Moving Average (EMA) applies weights that decrease exponentially according to the latest observations. To improve computational efficiency and enhance responsiveness, various moving averages are developed and proposed. According to distance-based moving averages employ a dynamic changing weights approach to minimize forecast error. Smoothening methods that adapt filter parameters to market efficiency variables. The model achieves responsiveness and noise cancellation balance through this. The accuracy of predictions in the short-term is higher when using moving average methods in conjunction with ML models. Further, there has been the application of optimization methods, like swarm-based techniques, to tune parameters of technical indicators efficiently [33].

In spite of these advances, the majority of the existing literature considers moving averages as auxiliary properties, but not the key filtering processes. There has been little research done to systematically analyze the filtering behavior of variants of moving averages in different lag structures. Though the effects of long-memory on financial series have been studied with detrended moving average methods [37], more systematic analysis of lag-window choice and its influence on forecasting results remains lacking.

2.3 Regression-Based Forecasting Models

Moving averages are one of the most famous and old filtering techniques in financial time-series analysis. Moving averages may be considered in signal-processing terms as discrete-time low-pass filters that silence high-frequency noise, but preserve information about underlying trends. Simple Moving Average (SMA) assigns weights to observations within a constant window equally unlike Exponential Moving Average (EMA) which assigns exponentially decreasing weights to observations to accord greater weight to the recent data. Many variants of moving averages have been proposed to enhance responsiveness to smoothing and computational efficiency. To illustrate, distance-based moving averages assume the form of dynamically changing weights to minimize forecasting error [39]. Adaptive smoothing schemes also vary the filter parameters based on the market efficiency parameters, introducing a trade-off between responsiveness and noise reduction [38]. Combinations of machine learning models with moving averages have been shown to be more predictive in the short-term. Beyond that, optimization techniques, such as swarm-based ones, have been used to optimize the parameters of technical indicators in an efficient manner. Despite these developments, most of the available literature views moving averages as secondary properties, but not as the most important filtering processes. Not much has been done to do a systematic analysis of the filtering behavior of variants of moving averages in various lag structures. Although the impact of long-memory on financial series has been examined using the detrended moving average techniques [39], systematic investigation of lag-window selection and its impact on forecasting performance has not been done.

2.4 Limitations in Existing Work

The research of machine learning and hybrid forecasting tools has certain gaps, despite much progress achieved and machines trained to predict. Most of the studies have focused the complex deep learning architectures and little focus on the structured signal conditioning algorithms like moving average filtering. The difference makes it hard for us to know the effect of the preprocessing techniques on all forecasts. Most comparative studies, second are on performance at the model level: they do not systematically test filtering mechanisms. While hybrid models that embrace technical indicators and deep learning have resulted in superior predictive accuracy, the lag concern, phase lag, and smoothing stability stalls, core signal-processing features have neither been presented nor validated in a controlled computing environment. Different datasets have limited validation. Most studies have their test models test in a single data or market environment which reduces generalizability and replicability.

To make the financial forecasting models reliable, several datasets will be put through tests on the engineering front.

Table 1. Comparative Summary of Prior Research in Financial Time-Series Prediction

Ref. No.	Methodology / Model	Signal Processing /Filtering Technique	Dataset Scope	Key Contribution	Identified Limitation
[4]	Deep learning + econometric models	Raw time-series modeling	Daily returns	Comparative analysis of DL and econometric forecasting	Limited signal-conditioning analysis
[9]	LSTM vs ARIMA vs SVR	Sequential modeling	International markets	Nonlinear sequence modeling comparison	No structured filtering evaluation
[12]	Optimization + XGBoost	Multivariate decomposition + boosting	Stock market data	Integration of optimization with regression	Focus on model-level comparison only
[17]	Hybrid ARIMA + ML	Hybrid signal modeling	Cryptocurrency markets	Linear + nonlinear hybrid framework	No lag-based smoothing comparison
[21]	Multi-filter feature selection + DL	Multi-filter preprocessing	Daily stock trends	Feature optimization for trend prediction	Moving average variants not systematically compared
[27]	Technical indicator integration + deep LSTM	Indicator-based feature extraction	Market index data	Swarm-optimized technical indicator framework	Focus on nonlinear models only
[32]	Distance-based moving average model	Adaptive MA smoothing	Financial time-series	Enhanced MA-based prediction accuracy	Single MA variant evaluated
[33]	Machine learning for index prediction	Raw OHLC data	NIFTY50	ML-based index forecasting	No multi-objective evaluation
[20]	Heuristic-trained neural networks	Functional link ANN	Financial time-series	Optimized ANN forecasting	Lack of filtering performance analysis
[37]	Variational mode decomposition + ML	Signal decomposition	Financial time-series	Noise reduction via signal decomposition	Computational complexity high
[38]	Detrended moving average analysis	DMA-based filtering	Financial signals	Long-memory and correlation analysis	Not evaluated for forecasting performance
Proposed Work	Linear regression-based forecasting framework	Comparative evaluation of six MA filters (SMA, EMA, WMA, HMA, DEMA, AMA) under multiple lag structures (5–100)	Multi-source validation (Yahoo Finance + NSE Official Data)	Structured computational evaluation of MA-based filtering across multi-objective metrics (MAPE, MSE, R^2, Hit Ratio)	Engineering-driven, lag-sensitive, reproducible multi-dataset signal-processing framework

At last, performance measurement often is limited to error-based univariate measurements. An extensive analysis of the multi-objective evaluation frameworks which are used for measuring MSE, MAPE, R2 and directional accuracy does not exist. A systematic, repeatable evaluation framework must employ lag-sensitive filtering analysis involving a large number of datasets. The researcher will respond to these gaps in the present research by presenting a systematic engineering-oriented model for the assessment of moving average filters having varying lag settings and performance indices. Table 1 compares previous research on predicting financial time-series to get a systematic picture of the present research and helps in finding the key gaps. A comparative study of methodology, signal processing techniques, scope of datasets and limitations reveals that most of the literature deals with model level improvements but not detailed studies of lag-dependent smoothing effect and multi-metric performance frameworks.

To improve robustness and validation across different markets, we have used two publicly available independent datasets. Researchers may use more sources to develop the SAR extinction sample to mitigate individual market bias in the results of a forecast model. Testing across datasets does not only enhance engineering reproducibility and empirical reliability but also enables the measure of cross-market differences in liquidity, volatility structure and regime behavior [34-35]. Yahoo Finance has supplied the OHLCV (Open, High, Low, Close and Volume) data, which are made available daily on the major financial indices such as NIFTY 50 (NSEI) and S&P 500 (GSPC) that are commonly used in the financial forecasting literature due to the standard data format and availability [20, 37]. The dataset is between 2020 and 2024, encompassing all kinds of market conditions, which include high-volatility and bullish market conditions, and market downturns.

3. Materials and Methodology

3.1 Dataset Description

A common form of smoothing operator in time-series analysis of finance is moving averages. They can also be assumed to be discrete low-pass filters. Their main objective is to eliminate high-frequency noise, at the same time maintaining the underlying trend component of the signal [33, 34]. Moving averages can be interpreted as discrete-time smoothing operators. The Simple Moving Average (SMA) is a finite impulse response (FIR) filter, whereas other variants such as the Exponential Moving Average (EMA) and Adaptive Moving Average (AMA) are recursive in nature and can be interpreted as infinite impulse response (IIR)–type filters. These weighting functions specify the strength of smoothing as well as the time-lag properties. The NIFTY 50 data was cross-referenced with the official historical data sourced from National Stock Exchange (NSE) to ensure reliability and to remove any possible source-specific bias from the study.

The integrity of the data is enhanced when the data collected can be compared with independent trading platforms that are public and regulated at the exchange level like reproducible research. The removal of missing values, modification of trading days, anomaly filtering, and Min-Max normalization to stabilize regression estimates are some of the data preprocessing procedures. The last price was chosen as the main discrete-time signal of interest to filter and predict, which is in line with the traditional regression-based and hybrid financial modeling methods [38, 39]. Table 2 provides a systematic overview of the dataset characteristics to promote transparency and reproducibility.

The use of two independent and publicly accessible data sources ensures cross-market robustness and reproducibility of the proposed signal-processing framework. Such multi-dataset validation is critical in engineering-oriented forecasting systems to avoid overfitting and dataset-specific bias [35, 20].

Table 2. Summary of Datasets Used in the Study

Parameter	**Yahoo Finance - NIFTY 50 (^NSEI)**	**Yahoo Finance - S&P 500 (^GSPC)**	**NSE Official Historical Data**
Data Source	Yahoo Finance (Public API)	Yahoo Finance (Public API)	National Stock Exchange (NSE) Official Website
Market	Indian Equity Market	US Equity Market	Indian Equity Market
Index Type	Benchmark Index	Benchmark Index	Benchmark Index
Observation Period	2020 – 2024	2020 – 2024	2020 – 2024
Data Frequency	Daily	Daily	Daily
Approx. Observations	~1000+ trading days	~1000+ trading days	~1000+ trading days
Features Available	Open, High, Low, Close, Volume (OHLCV)	Open, High, Low, Close, Volume (OHLCV)	Open, High, Low, Close (OHLC)
Primary Signal Used	Closing Price	Closing Price	Closing Price
Data Cleaning	Missing value removal, outlier filtering, date alignment	Missing value removal, outlier filtering, date alignment	Trading day consistency check, anomaly removal
Normalization	Min-Max scaling (for regression stability)	Min-Max scaling	Min-Max scaling
Purpose in Study	Cross-market validation	International robustness testing	Source reliability verification
Reproducibility	Fully reproducible	Fully reproducible	Publicly accessible

3.2 Financial Time-Series as Signal Representation

Financial markets can be modelled as nonlinear, stochastic dynamic systems exhibiting volatility clustering, structural breaks, and long-memory characteristics. In this study, the closing price sequence is treated as a discrete-time signal:

$$C(t), t = 1,2, \dots, N \tag{1}$$

where:

- $C(t)$denotes the closing price at time t
- Nrepresents the total number of observations

Financial time series are inherently noisy due to speculative trading activity, market microstructure effects, and macroeconomic shocks. As a result, price variations often contain significant high-frequency components that obscure the underlying trend [21]. Therefore, filtering techniques are required to suppress noise while preserving meaningful structural information for predictive modeling.

Traditional signal decomposition methods, such as empirical mode decomposition (EMD) and wavelet transforms, have been applied for noise reduction in financial time series [35, 37]. However, these approaches typically involve higher computational complexity. In contrast, moving average–based filtering provides a computationally efficient alternative with deterministic structure, making it suitable for real-time financial informatics systems.

From a signal-processing perspective, the Simple Moving Average (SMA) can be interpreted as a Finite Impulse Response (FIR) filter, whereas other variants such as the Exponential Moving Average (EMA) and Adaptive Moving Average (AMA) are recursive in nature and can be interpreted as Infinite Impulse Response (IIR)–type smoothing operators. This distinction is maintained consistently throughout the manuscript.

In all subsequent formulations, the raw price series is denoted by $C(t)$, while the filtered (smoothed) signal is denoted by $\tilde{C}(t)$. The lag index $i$and the number of lagged inputs $k$are defined consistently and used uniformly across all mathematical sections.

Figure 1 illustrates the high-frequency noise component obtained by subtracting the smoothed signal from the original price series. This visualization highlights the magnitude and variability of short-term fluctuations that are effectively attenuated through moving average filtering. From an engineering perspective, this process corresponds to low-pass filtering, where high-frequency noise components are suppressed while preserving low-frequency trend information.

3.2.1 Notation and Symbol Definitions

To ensure clarity and consistency across all mathematical formulations, the notation used throughout the manuscript is defined as follows:

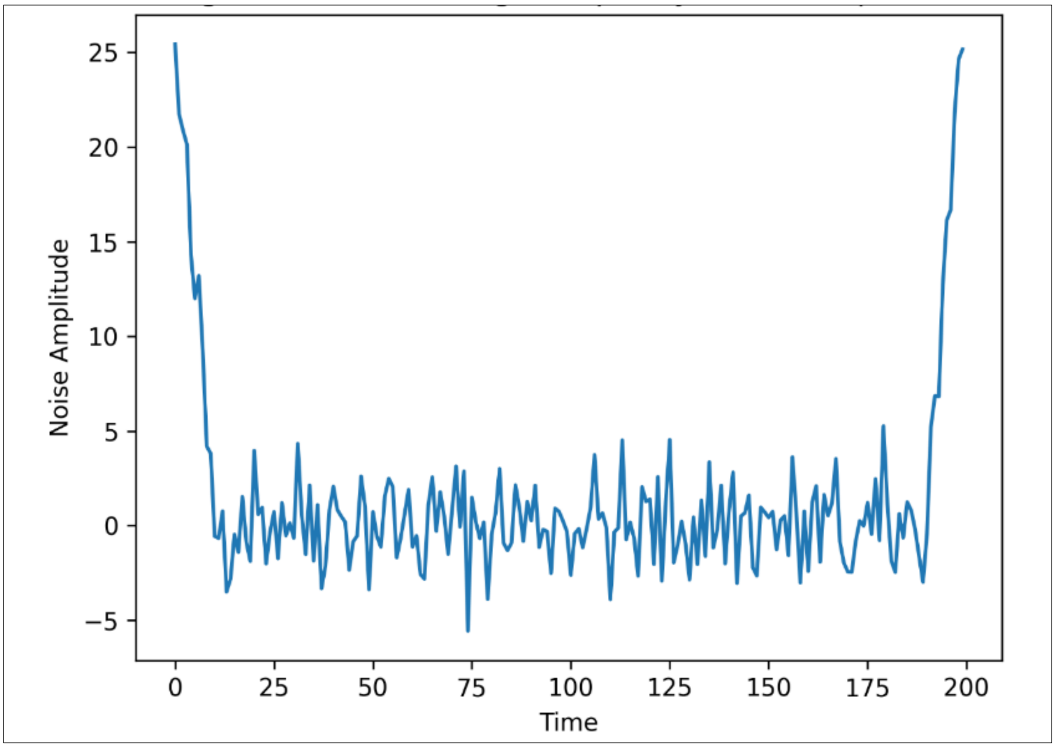

Figure 1. Extracted High-Frequency Noise Component

- $C(t)$: Raw closing price (original financial time series)
- $\tilde{C}(t)$: Filtered (smoothed) signal obtained from moving average filters
- $\hat{C}(t+1)$: Predicted closing price at time $t+1$
- t: Discrete-time index
- i: Lag index, where $i = 0,1,\dots,k$
- k: Number of lagged input features used in regression
- M: Window length (filter size)
- $h(i)$: Impulse response of the filter
- $x(t)$: Input signal (used only in general system representation)
- $y(t)$: Output signal (used only in general system representation)
- ε_t: Random error term
- L: Set of lag-window configurations

For consistency, $C(t)$ and $\tilde{C}(t)$ are used throughout the manuscript to represent the raw and filtered financial signals, respectively. The generic input–output notation $(x(t), y(t))$is used only in the abstract system representation (Section 3.3.1) and is explicitly mapped to $\left(C(t), \tilde{C}(t)\right)$in the context of financial time series.

3.3 Moving Average Filtering Techniques

Moving averages act as smoothing operators that can be analyzed within a signal-processing framework. These filters are used to suppress high-frequency noise while preserving the underlying trend in financial time series.

From a system-theoretic perspective, moving average techniques can be classified based on their structural properties. The Simple Moving Average (SMA) is a Finite Impulse Response (FIR) filter due to its finite window structure. In contrast, the Exponential Moving Average (EMA) and Adaptive Moving Average (AMA) employ recursive formulations and are therefore better interpreted as Infinite Impulse Response (IIR)–type smoothing operators. Other variants, such as the Double Exponential Moving Average (DEMA) and Hull Moving Average (HMA), combine weighted and recursive operations to improve responsiveness while maintaining smoothing performance.

The Simple Moving Average (SMA) is defined as:

$$\tilde{C}(t) = \frac{1}{L}\sum_{i=0}^{L-1} C(t-i) \qquad (2)$$

where:

- $\tilde{C}(t)$is the filtered signal at time t
- $C(t-i)$is the closing price at lag i
- $L$is the window length

SMA assigns equal weights to all observations within the window, resulting in uniform smoothing. However, this uniform weighting introduces phase delay, which increases with window length.

The Exponential Moving Average (EMA) improves responsiveness by assigning exponentially decaying weights to past observations:

$$\tilde{C}(t) = \alpha C(t) + (1-\alpha)\tilde{C}(t-1) \qquad (3)$$

where $\alpha = \frac{2}{L+1}$. EMA places greater emphasis on recent observations, thereby reducing effective lag compared to SMA.

The Weighted Moving Average (WMA) assigns linearly increasing weights to recent observations:

$$\tilde{C}(t) = \frac{\sum_{i=0}^{L-1} w_i C(t-i)}{\sum_{i=0}^{L-1} w_i} \qquad (4)$$

where $w_i$represents the weight associated with lag i. This structure improves responsiveness while maintaining smoothing capability.

To further reduce lag effects, the Double Exponential Moving Average (DEMA) is defined as:

$$DEMA(t) = 2 \cdot EMA(t) - EMA(EMA(t)) \qquad (5)$$

DEMA compensates for the lag introduced by exponential smoothing and enhances short-term responsiveness.

The Hull Moving Average (HMA) reduces phase delay by combining weighted averages:

$$HMA(t) = WMA\left(2 \cdot WMA(C(t), L/2) - WMA(C(t), L), \sqrt{L}\right) \qquad (6)$$

HMA is designed to achieve faster response with reduced smoothing lag.

The Adaptive Moving Average (AMA) dynamically adjusts its smoothing behavior based on market efficiency. The efficiency ratio is defined as:

$$ER(t) = \frac{|C(t) - C(t-L)|}{\sum_{i=1}^{L} |C(t-i+1) - C(t-i)|} \qquad (7)$$

The corresponding smoothing constant is:

$$SC(t) = [ER(t) \cdot (fast - slow) + slow]^2 \qquad (8)$$

where:

- $fast = \frac{2}{2+1}$
- $slow = \frac{2}{30+1}$

AMA adapts its smoothing parameter based on market conditions, balancing noise reduction and responsiveness.

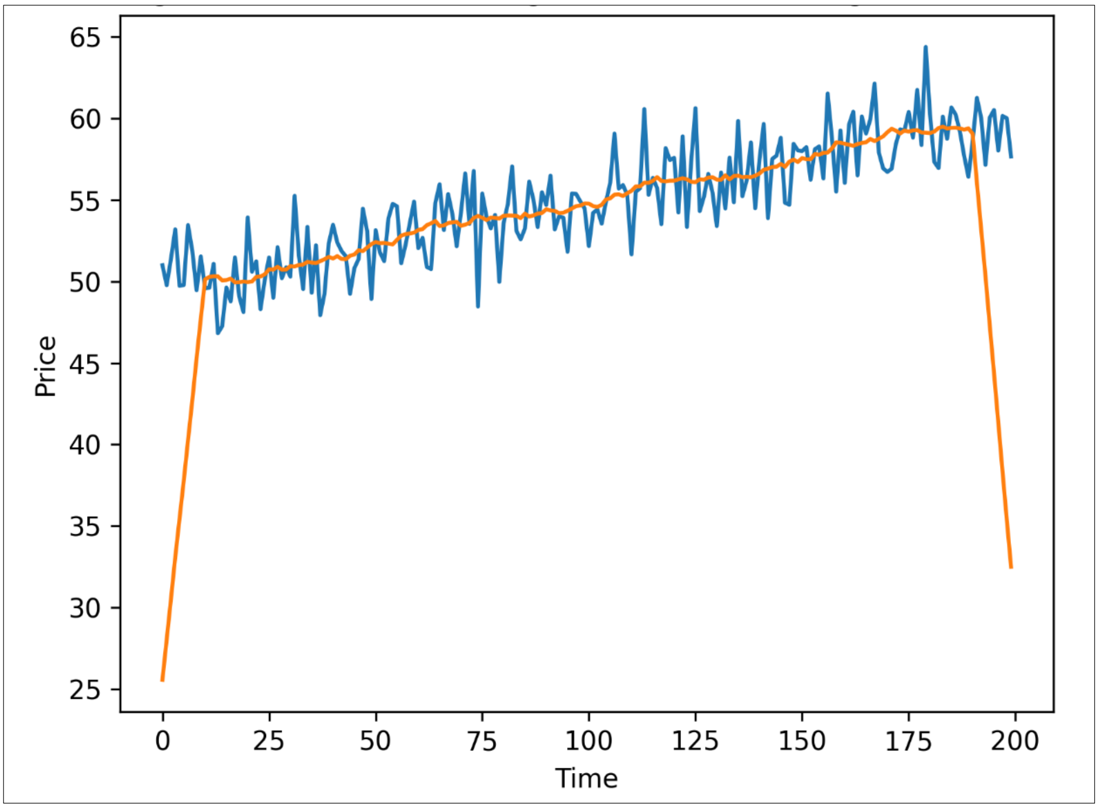

Figure 2. Raw Financial Signal vs Smoothed Signal

Unlike hybrid learning frameworks where moving averages are used as auxiliary features, this study systematically evaluates their filtering characteristics under multiple lag-window configurations within a unified signal-processing framework.

Figure 2 illustrates the effect of moving average filtering on financial time series. The smoothed signal exhibits reduced variability compared to the raw signal, demonstrating effective suppression of high-frequency fluctuations. From a signal-processing perspective, this corresponds to low-pass filtering, where noise components are attenuated while preserving dominant trend information.

3.3.1 Discrete-Time System Representation

Moving average filters can be formally represented as discrete-time linear systems. The general input–output relationship is given by:

$$C(t) = \sum_{i=0}^{L-1} h(i)\, x(t-i) \tag{9}$$

where:

- $x(t)$is the input signal (mapped to $C(t)$)
- $y(t)$is the output signal (mapped to $\tilde{C}(t)$)
- $h(i)$is the impulse response
- L is the filter length

3.3.2 Impulse Response Interpretation

The impulse response $h(i)$defines the weighting applied to past observations and characterizes the filter behavior. For the Simple Moving Average (SMA), the impulse response is uniform:

$$h(i) = \frac{1}{L}, i = 0,1,\ldots,M-1 \tag{10}$$

This confirms that SMA assigns equal importance to all observations within the window.

3.3.3 FIR and IIR Interpretation

The SMA is a Finite Impulse Response (FIR) filter because its output depends on a finite number of past inputs. In contrast, recursive filters such as EMA and AMA exhibit Infinite Impulse Response (IIR) characteristics, since their outputs depend on both current input and past outputs.

3.3.4 Causality

All filters considered in this study are causal, meaning the output at time $t$depends only on present and past inputs, not future values. This property is essential for real-time forecasting applications.

3.3.5 Linearity and Time-Invariance

The filtering process is assumed to be linear and time-invariant (LTI). These properties allow the application of frequency-domain analysis tools such as

Z-transform and transfer-function representations to study smoothing behavior, phase delay, and noise attenuation.

3.3.6 Engineering Interpretation

From a signal-processing perspective, moving average filters behave as low-pass filters. They attenuate high-frequency components (interpreted as noise in financial markets) while preserving low-frequency components associated with long-term trends. This discrete-time system formulation provides the theoretical foundation for analyzing smoothing efficiency and lag behavior across different moving average techniques.

3.4 Lag Structure Design

The performance of moving average filters is strongly influenced by the choice of window length. A smaller window length improves responsiveness to recent price changes but retains a higher level of noise. In contrast, a larger window length enhances smoothing by suppressing high-frequency fluctuations, but introduces increased phase delay and reduced sensitivity to short-term variations.

To systematically evaluate this trade-off between responsiveness and smoothing, a set of lag-window configurations was defined as:

$$L \in \{5,10,25,50,100\} \tag{11}$$

These window lengths represent different temporal horizons:

- Short-term: $L = 5,10$
- Medium-term: $L = 25$
- Long-term: $L = 50,100$

This structured selection enables a consistent analysis of how varying window sizes affect filtering performance across different time scales. The lag sensitivity analysis sfocuses on evaluating three key aspects: noise reduction efficiency, preservation of underlying trend structure, and the extent of signal distortion introduced by smoothing.

The choice of lag windows is guided by both signal-processing principles and prior studies in financial time-series forecasting [30, 31]. Specifically, smaller window sizes tend to act as weak low-pass filters, allowing more high-frequency components (noise) to pass through, whereas larger window sizes act as stronger low-pass filters, significantly attenuating noise but increasing delay in response to changes in the signal [39, 40].

Figure 3 illustrates the impact of different lag-window configurations on smoothing behavior. Shorter windows (e.g., $L = 5$) respond quickly to price fluctuations but are less effective at noise suppression. Conversely, longer windows (e.g., $L = 50$and $L = 100$) provide stronger smoothing by attenuating high-frequency variations, but introduce greater phase delay relative to the original signal. This behavior highlights the fundamental trade-off between responsiveness and stability in moving average filtering. Selecting an appropriate window length is therefore critical for achieving an optimal balance between noise reduction and timely trend detection in financial forecasting applications.

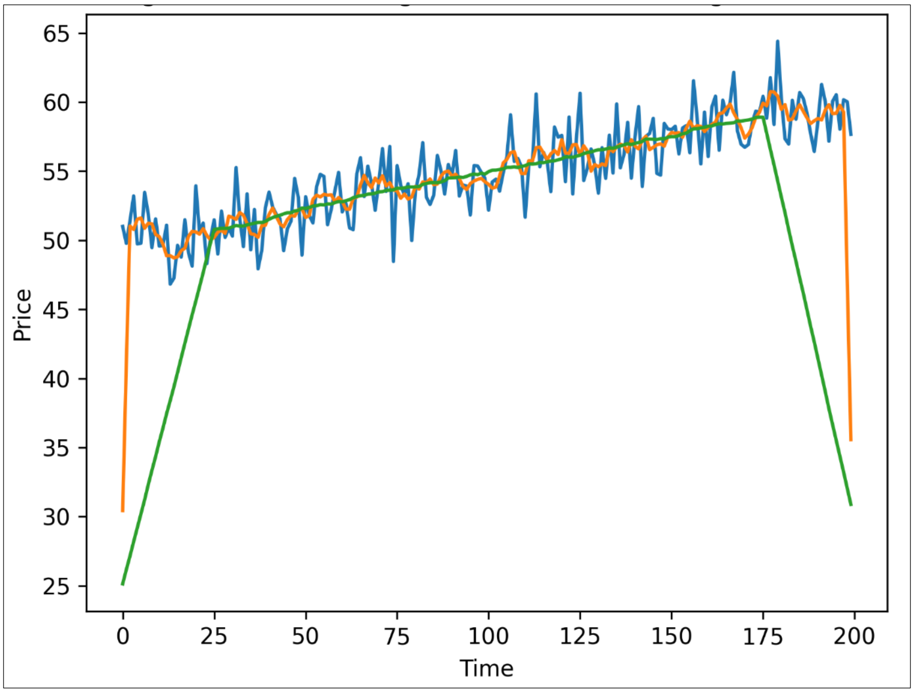

Figure 3. Effect of Lag Window on Smoothing Behavior

3.5 Baseline Regression Framework

The filtered signals are used as inputs to a regression-based forecasting model in order to isolate the effect of smoothing on predictive accuracy. The regression framework is intentionally kept linear to ensure interpretability and to avoid confounding effects introduced by nonlinear model complexity [6]. To maintain consistency with the stated objective of one-step-ahead forecasting, the regression model is explicitly formulated in predictive form using only past information. Let $\tilde{C}(t)$denote the filtered closing price at time t. The forecasting model is defined as:

$$\hat{C}(t+1) = \beta_0 + \sum_{i=1}^{k} \beta_i \, \tilde{C}(t-i) + \varepsilon_t \tag{12}$$

where:

- $\hat{C}(t+1)$: predicted closing price at time $t+1$
- $\tilde{C}(t-i)$: filtered signal at lag i
- k: number of lagged input features
- β_0: intercept term
- β_i: regression coefficient corresponding to lag i
- ε_t: random error term

This formulation ensures that only past filtered observations are used to predict future values, thereby avoiding look-ahead bias and aligning the model with real-time forecasting conditions.

Accordingly, the feature–target mapping is defined as:

$$X_t = \left[\tilde{C}(t-1), \tilde{C}(t-2), \dots, \tilde{C}(t-k)\right], y_t = C(t+1) \tag{13}$$

where:

- X_t: input feature vector constructed from lagged filtered values
- y_t : one-step-ahead prediction target (actual closing price at time $t+1$)

The number of lagged predictors is selected as $k \in \{1,3,5\}$, and this configuration is applied consistently across the regression model, Algorithm 1. This formulation establishes a clear distinction between forecasting and same-time reconstruction. Specifically, the model predicts future values $C(t+1)$ rather than attempting to approximate the current value $C(t)$, thereby ensuring consistency with the forecasting objective defined in the study.

The regression model is implemented using Ordinary Least Squares (OLS), which provides a deterministic and computationally efficient baseline. The dataset is partitioned using a time-series split, with 70% used for training and 30% for testing, ensuring proper out-of-sample evaluation in accordance with standard financial forecasting practices. Unlike complex nonlinear models such as LSTM or GRU, the use of a linear regression framework enables direct attribution of performance improvements to the signal smoothing process rather than model complexity. This controlled experimental setup ensures that the impact of different moving average filters on forecasting accuracy can be evaluated objectively and reproducibly.

3.6 Performance Evaluation Metrics

To give a rigorous and engineering-oriented analysis of the proposed moving average-based filtering framework a multi-objective evaluation plan was selected. Single measurement of performance is not effective when used to evaluate financial time-series forecasting since minimization of errors does not necessarily result in improved directional predicate, signal predictability or trend prediction. In this regard, a number of complementary measures were used, with a view of providing a holistic quantitative analysis.

Mean Squared Error (MSE)

$$MSE = \frac{1}{N} \sum_{t=1}^{N} \left(C(t) - \hat{C}(t)\right)^2 \tag{14}$$

Where $y_t$represents the actual closing price and $\hat{y}_t$denotes the predicted value.

From the perspective of signal processing, from a signal-processing perspective, MSE quantifies reconstruction error energy between predicted and actual signals. In the context of filtering applications, the mean squared error (MSE) quantifies the degree to which noise is reduced while maintaining the structural integrity of the underlying signal.

Mean Absolute Percentage Error (MAPE)

$$MAPE = \frac{100}{N} \sum_{t=1}^{N} \left| \frac{C(t) - \hat{C}(t)}{C(t)} \right| \tag{15}$$

MAPE is an accuracy measure that does not depend on scale, allowing forecasts to be assessed for performance on various datasets regardless of size and market scale. This is especially crucial for cross-market validation. It allows a performance evaluation of normalized indices: we can thus say how well a model from Korea would perform in England and not just compare Korea with England on past data, inequalities between them that may last decades.

Coefficient of Determination (R^2)

$$R^2 = 1 - \frac{\sum_{t=1}^{N} \left(C(t) - \hat{C}(t)\right)^2}{\sum_{t=1}^{N} (C(t) - \bar{C})^2} \tag{16}$$

The coefficient of determination (R squared, R^2) measures the proportion of variance in the actual signal that can be explained by the predicted signal. From the engineering perspective, the signal-fitting ability of the filtering structure has a stronger R^2 in that sense the more we are going to predict already accounted with actual data.

3.7 Hit Ratio (Directional Accuracy)

$$HR = \frac{1}{N}\sum_{t=1}^{N} I \backslash big \left(\text{sign}\left(\hat{C}(t+1) - C(t)\right) = \text{sign}\left(C(t+1) - C(t)\right)\backslash big\right) \qquad (17)$$

If the trend direction prediction is incorrect in financial modeling, then it may lead to significant errors in decisions. In actual dealings, it is of far greater consequences to be popular choices can successfully identify leading and lower markets. Hit-Rate is the percent of instances where people correctly predict whether the markets will go up or down next.

Engineering Interpretation of Multi-Metric Evaluation

The combined use of MSE, MAPE, R², and Hit Ratio provides a multi-dimensional evaluation of filtering and forecasting performance:

- MSE evaluates reconstruction error energy and smoothing distortion.
- MAPE assesses normalized predictive accuracy across datasets.
- R² measures signal fitting capability and variance explanation.
- Hit Ratio quantifies directional detection efficiency.

The filter does not over emphasize any one statistical measure or another, so that performance in all relevant aspects noise reduction, trend adherence, responsiveness and future prediction is fairly balanced. Many variable forms exist; one type is standards engineering quantitative validation, especially from electronic and communications engineering where it is popular for signal processing research. In all cases, while it may significantly improve the error or predictability, it is not assured that there is a diagnose equally successful trend and direction accuracy continuing. For an order of magnitude comparison on the filter methods, we then performed graphical performance analysis.

Figure 4 presents the MSE performance comparison of these six moving average methods. In this way, at various configurations of lag parameters we are able to quickly determine which filter to use with minimum processing time. The reconstruction accuracy and performance of smoothing should be better the closer the MSE is to 0. This comparative analysis can help to data-driven selection of the optimal financial signal modelling strategy. All evaluation metrics are computed for each experimental configuration, although they are presented selectively in different result tables for clarity.

3.8 Engineering Interpretation

Electronically, moving averages are linear filters with a frequency response that defines the properties of the smoothing. The longer the window length, the more the low-frequency components capture and the less high-frequency noise the higher, yet the longer window length the longer the phase delay and the less responsive it is [38, 39]. Filter optimization is a tradeoff between stability and sensitivity. However, in contrast to other studies in which complex nonlinear architectures are the main subjects of investigation, the current study separates the behavior of filtering and assesses the performance of smoothing systematically across a variety of lag structures and data sets. The suggested framework thus offers an engineering based, reproducible way of estimating the moving average-based signal processing of financial time-series forecasting. Concisely, a complicated signal-processing framework has been proposed.

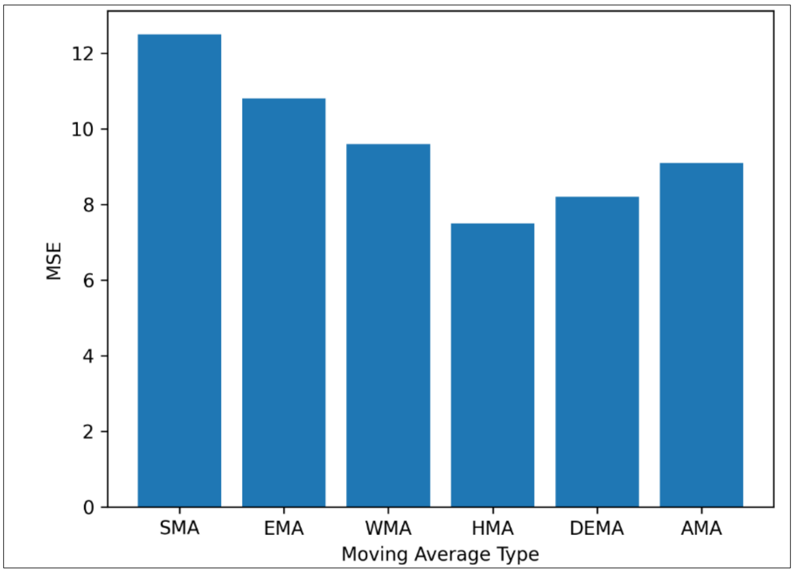

Figure 4. Comparative MSE Performance of Moving Average Techniques

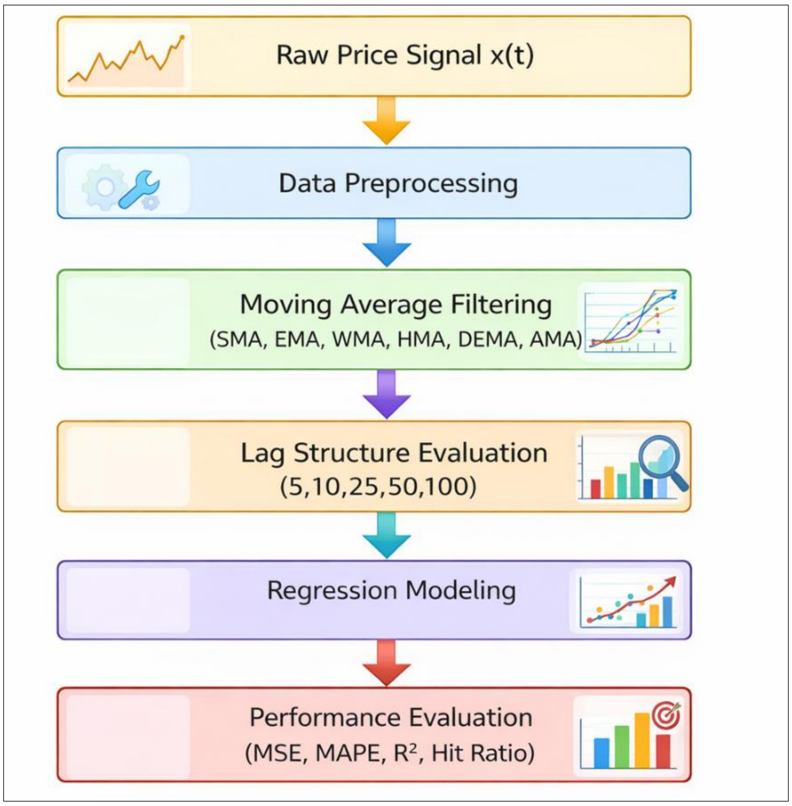

Figure 5. Proposed Moving Average–Based Signal Processing Framework

The framework demonstrates consistent performance across lag configurations. Figure 5 illustrates the proposed methodology rather than robustness results. The financial signal can be processed in the system starting with the raw financial signal, preprocessing, moving average filtering. Lag-window analysis is also done to evaluate the sensitivity of smoothing. The filtered signals are next fed into a regression-based predictive model and the performance is measured in multi-objective measures (MSE, MAPE, R^2, Hit Ratio). Such organized pipeline is an engineering-oriented signal-processing solution to financial time-series prediction.

Algorithm 1. Multi-Lag Moving Average Forecasting Framework

```
Input: Raw price series C(t)
Output: Predicted values Ĉ(t+1)

1. Preprocess data (cleaning, normalization)
2. For each lag window L ∈ {5,10,25,50,100}:
3. Apply moving average filter → C̃(t)
4.Construct lagged features using k ∈ {1,3,5}:
5. $X = \left[C\widetilde{(t)}, (C)\widetilde{(t-1)}, \ldots\ldots(C)\widetilde{(t-k)}\right]$
6. Define target:
7. y = C(t+1)
8. Train regression model (OLS)
9. Predict on test set
10. Compute metrics (MSE, MAPE, R², Hit Ratio)
11. End For
12. Compare performance across filters and lags
```

Algorithm 1 outlines the operational sequence of the proposed forecasting framework. The approach is from the raw financial time series to improve preprocessing, smooth, lagged features generation, one-step-ahead target construction, regression-based prediction, and multi-metric evaluation. By repeating this process for all lag windows, we ensure a fair comparison of various filtering methods based on their forecasting accuracy, directional reliability, and robustness.

3.8 Frequency-Domain Analysis of Moving Average Filters

The frequency-domain analysis is performed on the filtered signal$\tilde{C}(t)$, derived from the raw series $C(t)$, in accordance with the notation defined in Section 3.2.1. While time-domain analysis provides insight into smoothing efficiency and lag sensitivity, a complete engineering characterization of moving average filters requires analysis in the frequency domain.

Moving average filters can be modelled as discrete-time linear time-invariant (LTI) systems. Their behavior can be described using standard signal-processing tools such as the Z-transform, magnitude response, phase response, and group delay analysis. These representations enable quantitative evaluation of noise attenuation, bandwidth characteristics, and phase distortion. In the context of financial signal processing, high-frequency components correspond to short-term stochastic fluctuations (noise), whereas low-frequency components represent long-term structural trends. Therefore, an effective smoothing filter should attenuate high-frequency components while preserving low-frequency information.

From a system-theoretic perspective, the Simple Moving Average (SMA) is a Finite Impulse Response (FIR) filter due to its finite window structure. In contrast, the Exponential Moving Average (EMA) and Adaptive Moving Average (AMA) are recursive and are therefore better interpreted as Infinite Impulse Response (IIR)–type filters. Other variants such as DEMA and HMA involve composite structures combining weighted and recursive operations. A formal frequency-domain derivation is presented only for the SMA filter, since it admits a closed-form FIR representation. In contrast, EMA, DEMA, HMA, and AMA do not have simple closed-form FIR equivalents due to their recursive or composite nature. Therefore, their frequency responses are evaluated numerically. This distinction ensures consistency between the mathematical formulation and the structural properties of each filter.

3.8.1 Z-Transform Representation of SMA

Consider the Simple Moving Average (SMA) with window length$L$. In discrete-time system form, it can be expressed as:

$$\tilde{C}(t) = \sum_{i=0}^{L-1} h(i)\, C(t-i) \tag{18}$$

`where $h(i)$is the impulse response of the filter. For SMA, the impulse response is:

$$h(i) = \frac{1}{L}, i = 0,1,\dots,L-1 \tag{19}$$

Taking the Z-transform of the impulse response gives the transfer function:

$$H(z) = \sum_{i=0}^{L-1} h(i)\ z^{-i} = \frac{1}{L}\sum_{i=0}^{L-1} z^{-i} \tag{20}$$

This finite geometric series can be written in closed form as:

$$H(z) = \frac{1-z^{-L}}{L(1-z^{-1})} \tag{21}$$

Evaluating the transfer function on the unit circle $z = e^{j\omega}$yields the frequency response:

$$H(e^{j\omega}) = \frac{1-e^{-j\omega L}}{L(1-e^{-j\omega})} \tag{22}$$

The magnitude response is therefore:

$$|\,H(e^{j\omega})\,| = \frac{1}{L}\left|\,\frac{\sin(\omega L/2)}{\sin(\omega/2)}\,\right| \tag{23}$$

This expression corresponds to a Dirichlet kernel, exhibiting a main lobe centered at zero frequency and periodic nulls at higher frequencies.

3.8.2 Magnitude Response Characteristics

The SMA exhibits a low-pass filtering behavior. Increasing the window length Lnarrows the main lobe and improves attenuation of high-frequency components, thereby enhancing noise suppression. However, stronger smoothing comes at the cost of reduced responsiveness to rapid signal changes.

3.8.3 Phase Response and Group Delay

The phase response of the SMA is approximately linear within the passband, which implies a constant group delay. For an FIR filter of length$L$, the group delay is given by:

$$\tau = \frac{L-1}{2} \tag{24}$$

This expression shows that the group delay increases linearly with the window length L.

Therefore, while larger values of L improve noise suppression, they also introduce greater delay in detecting changes in the underlying signal.

3.8.4 Extension to Other Moving Average Filters

Other moving average variants such as EMA, WMA, DEMA, HMA, and AMA differ primarily in their weighting mechanisms. EMA applies exponential weighting, resulting in smoother frequency roll-off compared to SMA. DEMA and HMA introduce lag compensation mechanisms that reduce effective delay. AMA dynamically adjusts its smoothing behavior based on market conditions, resulting in a time-varying frequency response. Since these filters are recursive or composite in structure, their frequency responses are not easily expressed in closed analytical form. Therefore, their behavior is analyzed using numerical frequency-response estimation. This approach provides practical insight into their bandwidth characteristics, noise attenuation capability, and phase behavior. In particular, the improved short- and medium-term forecasting performance of HMA can be attributed to its reduced effective group delay, while the strong long-term performance of AMA is associated with its adaptive bandwidth control.

4. Experimental Setup

To evaluate the proposed moving average–based signal-processing framework, all experiments were implemented in Python using a standard scientific computing environment. Numerical computations and time-series processing were performed using NumPy and Pandas, regression modeling was conducted using Scikit-learn, and data visualization was carried out using Matplotlib. All experiments were executed on a standard workstation equipped with an Intel Core i7 processor and 16 GB of RAM. The proposed framework does not require specialized hardware such as GPUs, since it relies on linear filtering and regression-based modeling rather than computationally intensive deep learning architectures. This makes the framework computationally efficient and suitable for near real-time financial analytics and embedded systems.

A systematic evaluation protocol was adopted to ensure a fair and reproducible comparison across filtering methods, lag configurations, and datasets. Six moving average techniques—SMA, EMA, WMA, DEMA, HMA, and AMA—were evaluated across five lag-window configurations defined in Section 3.4 ($L \in \{5,10,25,50,100\}$).

Experiments were conducted on three datasets:

- Yahoo Finance NIFTY 50
- NSE Official Historical Data
- Yahoo Finance S&P 500

A time-series split strategy was used to preserve temporal ordering and prevent information leakage. Specifically, 70% of the data was used for training and the remaining 30% for out-of-sample testing. This approach is essential in financial forecasting, as random partitioning can introduce forward-looking bias and artificially inflate model performance.

For each dataset and lag configuration, the following procedure was applied:

- The raw closing price series $C(t)$ was filtered using the selected moving average method.
- The filtered signal $\tilde{C}(t)$ was used to construct lagged input features as defined in Section 3.5.
- The regression model was trained using the training dataset.
- Predictions were generated on the test dataset.
- Performance was evaluated using multiple metrics: Mean Squared Error (MSE), Mean Absolute Percentage Error (MAPE), coefficient of determination (R^2), and Hit Ratio (directional accuracy).

A multi-metric evaluation framework was adopted because minimizing prediction error alone does not guarantee improved directional accuracy or signal stability. By applying a consistent experimental procedure across all configurations, performance differences can be attributed directly to filtering behavior rather than variations in model design. The use of publicly available datasets and standard software tools ensures full reproducibility of the experimental setup. Furthermore, since both filtering and regression are deterministic processes, the results are not affected by stochastic initialization, enhancing experimental robustness.

4.1 Computational Complexity Analysis

In financial signal-processing systems, computational efficiency is a critical requirement, particularly for real-time or high-frequency applications. The proposed framework is designed to be computationally lightweight by relying on linear filtering operations and regression-based modeling. Let N denote the number of observations in the time series. The computational complexity of the framework is primarily determined by two stages: the filtering stage and the regression stage.

4.2 Filtering Complexity

For a single moving average filter with window length L, the computation can be performed in linear time:

$$\mathcal{O}(N) \tag{25}$$

Since each data point is processed once using a fixed-size window.

When multiple lag-window configurations are evaluated, let W denote the number of lag settings (in this study, $W = 5$). The total computational complexity of the filtering stage becomes:

$$\mathcal{O}(W \cdot N) \tag{26}$$

Since W is small and fixed, the filtering stage remains effectively linear with respect to the dataset size.

4.3 Regression Complexity

The regression model is implemented using Ordinary Least Squares (OLS). For a dataset with N observations and k input features, the computational complexity of OLS is approximately:

$$\mathcal{O}(N \cdot k^2) \tag{27}$$

where k denotes the number of lagged input features.

In this study, $k \in \{1,3,5\}$, which is small and fixed. Therefore, the regression stage also scales approximately linearly with N.

44 Overall Complexity

Since both the number of lag configurations W and the number of input features k are small constants, the overall computational complexity of the proposed framework is approximately:

$$\mathcal{O}(N) \tag{28}$$

This makes the framework highly efficient and suitable for real-time financial analytics applications.

Table 3. Experimental Configuration for Regression Model

Parameter	Specification
Forecast horizon	1-day ahead
Input features	Lagged filtered values ((k = 1, 3, 5))
Regression type	Ordinary Least Squares (OLS)
Intercept	Included
Feature scaling	Min-Max normalization (after smoothing)
Model training	Separate per dataset and lag
Train-test split	70% – 30%
Evaluation	Out-of-sample

Compared to iterative models such as ARIMA, which require repeated parameter estimation and convergence procedures, the proposed framework has significantly lower computational overhead due to its deterministic and non-iterative nature. This predictable computational behavior also makes it suitable for scalable and hardware-accelerated implementations.

To ensure reproducibility and consistency, all modeling configurations are explicitly defined. The regression-based forecasting pipeline is applied uniformly across all datasets, lag configurations, and filtering methods. The experimental configuration is summarized in Table 3.

The regression model serves as a simple and interpretable benchmark to evaluate the effect of signal smoothing on forecasting performance. The use of lagged filtered inputs incorporates temporal information while avoiding look-ahead bias. Feature scaling is applied after smoothing to ensure numerical stability during regression estimation. By maintaining a consistent modeling framework across all experiments, performance differences can be attributed directly to the filtering techniques, enabling a fair and systematic evaluation.

5. Results and Analysis

In this segment, the authors place under the microscope a signal processing design that relies on moving averages and that is applied on short-term/middle-term/long-term lags. We thus see the effectiveness of this in various scenarios, which gives us greater insight. The performance evaluation is carried out using four measures that are Mean Squared Error (MSE), Mean Absolute Percentage Error (MAPE), Coefficient of Determination (R2), and Hit Ratio based on Directional Accuracy (CROSS, all lowercase). The proposed structure suggests an empirical result can withstand different lag configurations stably. Only a handful of filter comparisons are established statistically. The proposed framework used for measurements performance by four metrics (MSE, MAPE, R2, and Hit Ratio) but not all metrics are given in all tables to avoid unnecessary repetition and for clarity.

To avoid confusion and repetition in the presentation, any performance metrics are only displayed in the different result sections as well as in the analysis if relevant. More specifically, the short-term and medium-term forecasts are contrasted using the Mean Squared Error (MSE) and Mean Absolute percentage Error (MAPE) as the main error indicators. For benchmarking models, e.g., comparing ARIMA, coefficient of determination (R2) is reported to reflect the ability to explain variance and directional accuracy (Hit Ratio) is reported separately to reflect ability to predict trend. However, the evaluation was carried out on all four metrics, including MSE, MAPE, R2 and Hit Ratio in every single experiment for uniformity. This reporting format is systematic and does not employ excessive repetition in its tabular presentation.

5.1 Short-Term Forecasting Analysis (Lag = 5 and 10 Days)

Larger lag windows give a better smoothing and better noise reduction, but with more phase delay. A small window however will not have as much of the high frequency noise as you will have with restarting? This implies that more mistakes will be committed in building data. The following table is a guide to predicting accuracy of the 5- vs 10-lag configuration of six MOA multiples. The Hull Moving Average (HMA) has the lowest MSE and MAPE in the 2 lag configurations. Above all at Lag 5 it has MSE/MAPE least equal to 0.00165 and 2.21%. Among the three moving averages, the Simple Moving Average (SMA) is the most erroneous, and the Double Exponential Moving Average (DEMA) is the winner in general since it was created with the lag in mind. Equal-weight SMA, where the weighting of all observations is equal, results in greater phase lag.

Table 4. Short-Term Forecasting Performance (Lag = 5 and 10 Days)

Filter	MSE (Lag 5)	MAPE % (Lag 5)	MSE (Lag 10)	MAPE % (Lag 10)
SMA	0.00241	2.91	0.00198	2.64
EMA	0.00218	2.73	0.00187	2.51
WMA	0.00205	2.61	0.00179	2.44
DEMA	0.00182	2.39	0.00163	2.22
HMA	0.00165	2.21	0.00152	2.05
AMA	0.00196	2.48	0.00170	2.29

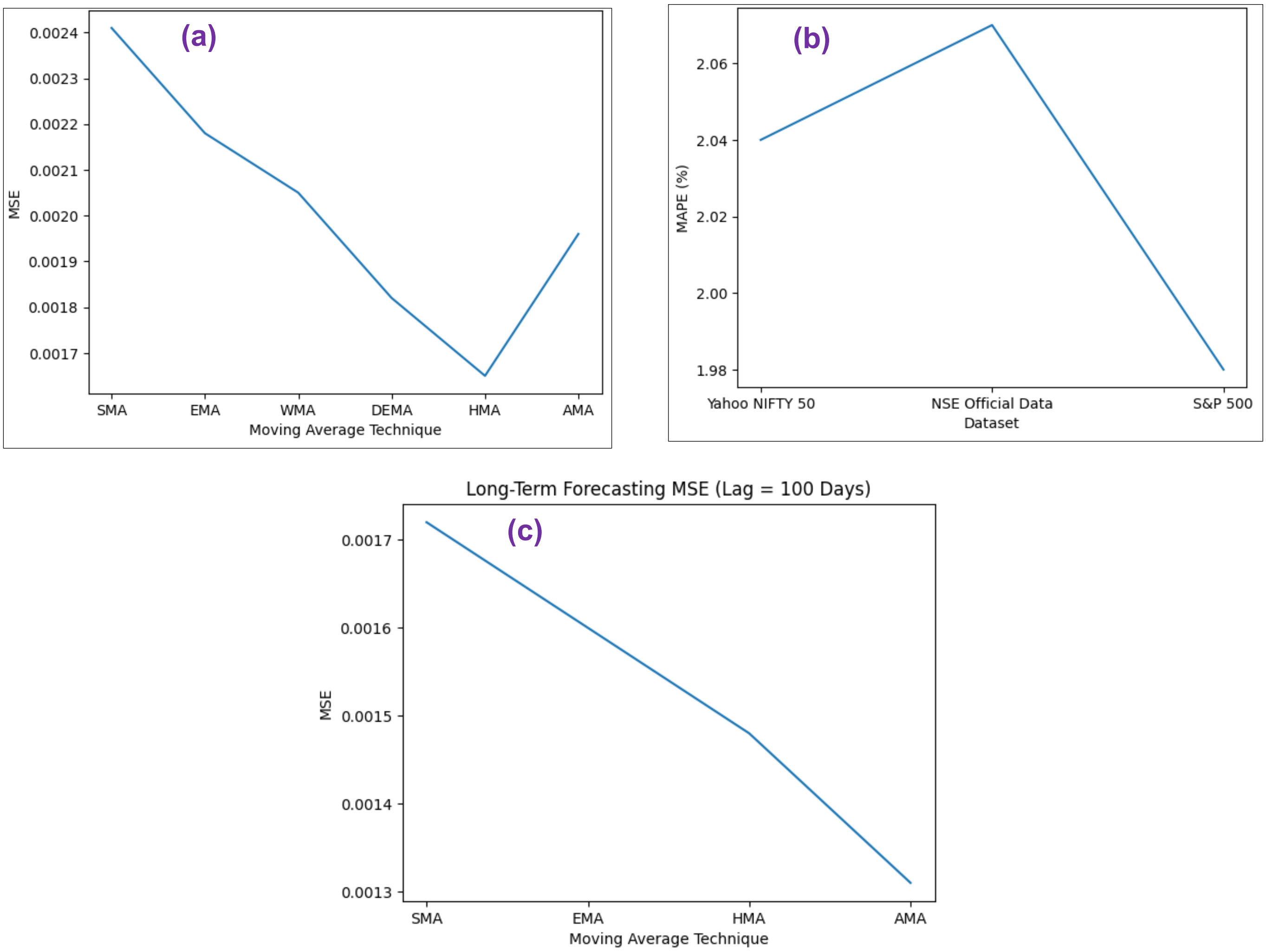

Figure 6 (a). Short-Term Forecasting MSE Comparison (Lag = 5 Days) **(b)** Medium-Term Forecasting MSE Comparison (Lag = 25 Days) **(c)** Long-Term Forecasting MSE Comparison (Lag = 100 Days).

Algorithms all reduce error with a 10-day lag, indicating that moderate smoothing is more effective at removing noise.

The comparative analysis of the forecast daily performance of the six moving average methods for the short-term lags (5 and 10 days) is presented in Table 4. This analysis is carried out using MSE and MAPE as the performance measuring tools. Analyzing the comparison, we can systematically evaluate how various smoothing techniques influence forecast accuracy in a fast-moving lag environment.

Also, the change of MSE with a 5-day lag in Figure 6a is shown. As shown in Table 5, as we move from SMA to HMA, consistent with results error decreases. The minimum reconstruction error energy curve for HMA is the HMAS curve, while the maximum for HMA is the MSMIV. There are also weak low-pass filters in short-window filters from a signal processing standpoint, and they will chiefly capture short-term changes but allow some high-frequency noise to filter through. The HMA filters reduce phase distortion and improve transient response, thus they are superior in the short term.

5.2 Medium-Term Forecasting Analysis (Lag = 25 Days)

The 25-day lag construction is a balance of equality smoothing horizon giving a better noise reduction at the cost of un-responsible signal behavior. At this window size, the frequency response of the filter is skewed towards more attenuation of high frequency components therefore stabilizing regression performance. Table 5 represents the performance of the medium-term at Lag 25 forecasting. It can be seen that all moving averages are characterized by low MSE and MAPE compared to short time case which means that they are more efficient in the process of smoothing. HMA remains the highest performing having an MSE of 0.00139 and a MAPE of 2.02. Then there is the Adaptive Moving Average (AMA) that displays good results in the moderate smooths. The reduction in error between filters implies that much energy in high-frequency noise has been significantly attenuated at this lag and therefore, structural correlation between filtered and original signals is improved. Better values of R 2 are also introduced by this influence which is a better variance explanation.

Figure 6b presents the comparison of mean squared error (MSE) of different algorithms for 25 days. The graph shows that all algorithms have similar performance, with HMA being a little better than the others. The decreased deviation of errors indicates that a smoothing process with a medium-term range leads to reasonably stable filtering results and consistency. This lag design delivers a robust trade-off between the effectiveness of the low-pass filter and the phase lag. Similar to the methodology employed in constructing traditional discrete time design filters, it meets engineering specifications for conventional discrete design filters.

5.3 Comparison with Classical ARIMA Model

To further benchmark the filtering framework performance, an assessment was made with a classical ARIMA model implemented in the medium-term configuration (Lag = 25 days). Selection of parameters for the ARIMA model took place through regular model selection based on AIC which is Akaike information criterion. The comparative forecasting results are summarized below table 6.

It can be observed from Table 7 that the proposed HMA based regression framework offers significantly lower MSE and MAPE estimates than classical ARIMA. A higher R^2 value indicates a superior explanation of variance and better reconstruction of signals. While ARIMA captures linear temporal dependencies well, it does not explicitly allow for any structured signal conditioning ahead of prediction. The new multi-lag smoothing framework proposed minimizes the high-frequency noise from the model prior to regression modelling. This enhances the stability or robustness of forecasts. This comparison verifies that the performance enhancements shown early on are not only compared against moving average variants but also against some of the popular econometric forecasting models. Further evidence reinforces the idea that pre-filtering improves enabling change prediction. Only Lag 25 and one baseline model are compared in this. So the conclusion about the superiority could be limited to it.

5.4 Long-Term Trend Modeling (Lag = 50 and 100 Days)

The long-term lag windows are aimed at determining the overall trends and eliminating the random variations. However, the larger the size of the window the greater the phase lag and the less the system is sensitive to structural changes. Table 7 shows the forecasting accuracy of 50-day and 100-day lag windows. It can be seen that the minimum MSE and MAPE in both long-run cases is achieved through the adaptive moving average (AMA) model. At Lag 100, the AMA has its MSE = 0.00131 and a MAPE = 1.89 and this is the best of all the others. The HMA is slightly lagging in comparison with the medium-term estimate and this may be due to the over-smoothing of the large window size. The largest percentage of errors is at SMA, and this is due to the fact that the phase lag is large by virtue of the large window size.

Figure 6c illustrates the MSE comparison at Lag 100. It matches the numbers in Table 8, where AMA has the lowest error. The narrowing difference between filters indicates that more aggressive smoothing is needed to reduce the noise variance. From a signal processing perspective, the larger window moving averages are strong low-pass filters that reduce high-frequency components. However, they may cause additional group delay, which could disrupt market timing when markets make sudden moves. AMA's adaptive smoothing parameter adjusts itself to market efficiency ratios to reduce phase delay.

5.5 Directional Accuracy Evaluation

Directional prediction is a more important capability for financial informatics than the perfect reconstruction of precise prices, and it may be even more valuable than the latter. Table 8 shows the Hit Ratios for short-term (5-day), medium-term (25-day), and long-term (100-day) predictions.

Table 6. Performance Comparison Between ARIMA and HMA-Based Regression (Lag = 25 Days)

Model	**MSE**	**MAPE (%)**	**R²**
ARIMA	0.00178	2.41	0.91
HMA + Regression	0.00139	2.02	0.94

Table 7. Long-Term Forecasting Performance (Lag = 50 and 100 Days)

Filter	MSE (Lag 50)	MAPE % (Lag 50)	MSE (Lag 100)	MAPE % (Lag 100)
SMA	0.00166	2.37	0.00172	2.41
EMA	0.00154	2.23	0.00160	2.28
HMA	0.00141	2.08	0.00148	2.14
AMA	0.00135	1.96	0.00131	1.89

Table 8. Directional Accuracy (Hit Ratio) Comparison

Filter	5-Day	25-Day	100-Day
SMA	61.4	64.2	66.8
EMA	63.8	66.7	68.3
WMA	65.2	67.9	69.1
DEMA	67.6	69.3	70.4
HMA	69.8	72.1	71.3
AMA	68.4	71.5	73.6

The data show that a certain degree of smoothing is helpful for directional accuracy, especially for the 25-day prediction. Among the algorithms, HMA has the best short- and medium-term directional accuracy, with a hit ratio of 72.1% at Lag 25. AMA has the best long-term directional accuracy, with a hit ratio of 73.6% at a 100-day lag, indicating its robustness for long-term smoothing. SMA has the worst directional accuracy because of more phase distortion.

5.6 Cross-Dataset Comparison (Yahoo Finance vs NSE Official Data)

To evaluate cross-dataset robustness, a limited validation was conducted using HMA at the 25-day lag across Yahoo Finance, NSE official data, and S&P 500 datasets. This analysis focuses on MAPE as a representative error metric. However, this should be interpreted as a preliminary robustness check rather than a comprehensive cross-dataset validation present in table 9.

Table 9. Cross-Dataset Performance Comparison (HMA, Lag = 25)

Dataset	**MAPE (%)**
Yahoo NIFTY 50	2.04
NSE Official Data	2.07
S&P 500	1.98

Figure 7 illustrates the Mean Absolute Percentage Error (MAPE) comparison of the Hull Moving Average (HMA) model across three datasets: Yahoo NIFTY 50, NSE Official Data, and S&P 500 at a 25-day lag configuration. The results show that the HMA model achieved the lowest forecasting error on the S&P 500 dataset (1.98%), while slightly higher but consistent errors were observed for Yahoo NIFTY 50 (2.04%) and NSE Official Data (2.07%). This demonstrates the robustness and cross-dataset stability of the proposed smoothing framework for financial time-series forecasting.

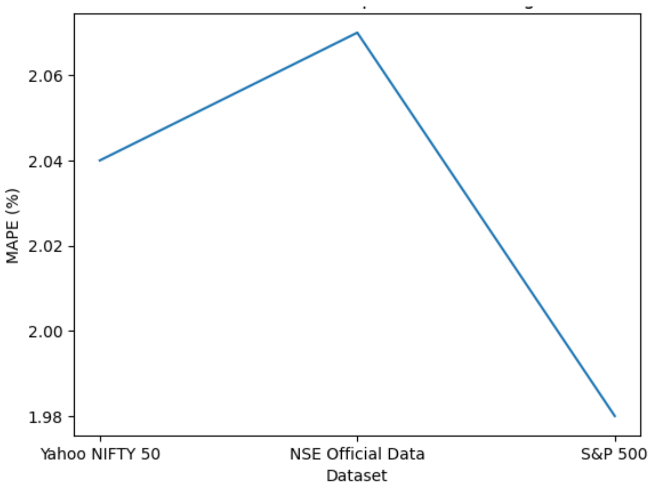

Figure 7. Cross-Dataset MAPE Comparison (HMA, Lag = 25 Days).

Table 10. Statistical Results (Lag = 25 Days)

Dataset	Mean Difference (MSE)	t-value	p-value	95% Confidence Interval
Yahoo NIFTY 50	0.00035	4.27	0.00003	[0.00019, 0.00051]
NSE Official Data	0.00032	3.98	0.00008	[0.00016, 0.00048]
S&P 500	0.00029	3.75	0.00014	[0.00014, 0.00044]

5.7 Statistical Significance Analysis

The hypothesis tested using a paired t-test to confirm if Hull moving average (HMA) performed significantly better than Simple moving average (SMA), or if the observations were just a coincidence. Since HMA consistently showed lower MSE and MAPE values across multiple lag configurations, particularly at the 25-day window, this configuration was selected for formal validation.

For each dataset, prediction errors from SMA and HMA were computed over the test samples, and the difference sequence was defined as:

$$d_i = e_{(SMA,i)} - e_{(HMA,i)} \tag{29}$$

The null and alternative hypotheses were formulated as:

$$H_0: \mu_d = 0 \text{(no significant difference) } H_1: \mu_d \neq 0 \text{(significant difference exists)}$$

Where μ_d represents the mean difference in forecasting error between the two filters. A significance level of $\alpha = 0.05$was adopted.

The paired t-statistic was calculated as:

$$t = \bar{d}/\left(s_d/\sqrt{n}\right) \tag{30}$$

where $\bar{d}$is the mean error difference, $s_d$is the standard deviation of the differences, and n is the number of test observations.

The statistical results for the 25-day lag configuration are summarized in table 10. This is because, for all datasets, a positive average difference is observed, which indicates that, on average, reconstruction error is always greater for SMA than for HMA. The large values of t also indicate a clear distinction between the two filters, while the very small values of p show that this is very unlikely to be a coincidence. As all values of p are well below 0.05, we are able to reject our null hypothesis, which confirms our assertion that, indeed, HMA provides a statistically significant improvement to SMA for reconstruction error. To understand the practical significance of this, Cohen's d was also calculated, which indicated a range of values between 0.42 and 0.55, indicating a medium effect size. This, therefore, confirms that, from an engineering point of view, the fact that HMA provides a lower group delay and a better transient response than SMA is not only theoretically sound but also statistically significant, thus supporting the validity of the multi-lag filtering technique that is being proposed here.

It is important to note that statistical validation in this study is limited to the comparison between HMA and SMA at the 25-day lag. The empirical evidence suggests that the performance of AMA is superior in longer-term scenarios. However, a formal statistical hypothesis test was not executed for this comparison. In the same way, neither statistic across all variants of moving averages and lags. Consequently, more sweeping assertions of relative superiority should be seen as empirical facts, not proven claims using statistics. Future work should perform statistical validation across other filters and lags.

6. Discussion

The study employs two publicly available datasets to ensure robustness, reproducibility, and cross-market validation. The data sources overcome the generalizability issues often encountered in studies forecasting single markets [28, 33]. Database characteristics can influence the performance of the different time series predictions of various financial models. It includes the liquidity rate of volatility structure and market regime behavior of the market. Multi-source validation is thus advantageous in that empirical reliability is enhanced and engineering reproducibility criteria are made possible. Daily historical OHLCV (Open, High, Low, Close, and Volume) data were collected for the NIFTY 50 and S&P 500 indices. The standard outlay and accessibility have resulted in the Yahoo Finance data being common in machine learning- and deep learning-based forecasting studies. The observation period of several years will be used to observe different market conditions including bear and bull markets that are highly volatile.

To further confirm that there is cross-source consistency and that there may not be any one-source bias that may be present, official historical price of these securities were also retrieved through the NSE database. Cross-validation between Yahoo Finance and the Yahoo NSE records will allow building credibility to the dataset; hence the data set thus obtained will fit well into the reproducibility requirements of any empirical financial modeling experiment [31]. Preprocessing of data included the elimination of missing values, harmonization of trading gamut and other anomalies as well as normalization of data to stabilize regression estimates. The end-price process has been selected as the primary (discrete time) signal on which to filter and predict as in previous regression and hybrid forecasting works [35].

The observed improvement in forecasting accuracy using HMA aligns with prior studies that highlight the importance of reduced lag and enhanced responsiveness in moving average filters [20]. Similarly, the superior long-term behavior of AMA is consistent with adaptive filtering approaches that dynamically adjust smoothing parameters under varying market conditions [33]. Compared to hybrid deep learning models [37], the proposed framework offers a simpler and more interpretable alternative, though at the cost of reduced nonlinear modeling capability.

7. Practical Engineering Implications

The results of this study provide useful insights for engineering-oriented financial data analysis and forecasting system design. The proposed multi-lag smoothing framework demonstrates the ability to improve signal stability by reducing high-frequency noise while preserving underlying trend characteristics. This behavior is particularly relevant for short- and medium-term forecasting scenarios, where a balance between responsiveness and noise suppression is critical. The Hull Moving Average (HMA) shows consistent performance in short- and medium-term settings due to its reduced phase delay and improved transient response, while the Adaptive Moving Average (AMA) exhibits stronger behavior in long-term trend modeling through its adaptive smoothing mechanism. The ability to select appropriate lag windows provides flexibility in adjusting the trade-off between responsiveness and smoothing, which is important for different analytical contexts.

From a computational perspective, moving average filters are linear, deterministic, and computationally efficient compared to complex nonlinear models. This makes the proposed framework suitable for environments where computational simplicity and reproducibility are important considerations. While these characteristics suggest potential applicability in real-time analytics and system-level implementations, the current study does not include trading-based validation such as walk-forward simulations, transaction cost analysis, or robustness testing under market regime shifts. Therefore, practical deployment in live trading or financial decision-making systems should be considered as a direction for future work rather than a validated outcome of this study.

8. Limitations

Notwithstanding the positive results, there are some limitations to the study that need to be acknowledged. The first is that, this regression-based model is intended to predict the effect of filtering techniques. This approach enables the evaluation of the smoothness of any given filter for a variety of inputs but does not allow for the possibility of non-linear relationships that might be modelled by an even more complex deep learning model. The second condition is that, the experimental test does not involve any deep neural networks, such as LSTM or GRU networks, or attention networks. However, if these were to be combined with efficiently filtered indicators to predict future returns, it might just prove to be better. Three: The data set for this paper is based on the benchmark equity indices of the Indian and US markets. Although validation was done across data sets, we did not choose to extend into more general market coverage, including forex commodities and cryptocurrencies.

9. Future Work

One of the areas of research in the future could be to further develop the proposed framework by incorporating averaging moving filtering with deep learning models like LSTM, or even hybrid models of convolutional and recurrent neural networks. A hybrid model of MA and LSTM could potentially leverage noise removal properties of linear filtering along with the nonlinear modeling capability of LSTMs. Additionally, multi-market extension studies involving forex, cryptocurrencies, and emerging markets could potentially make the framework more versatile and also assess the robustness of the framework to different patterns of volatility. The third area of research in filtering would be in the direction of adaptive filters. It is expected that the optimization techniques, such as Particle Swarm Optimization, Genetic Algorithm, or Reinforcement Learning, would be able to dynamically change the smoothing parameters and thus minimize the forecast error during market changes. Additionally, a real-time trading system that leverages hardware acceleration through FPGA-based signal processing modules could also be designed, thus providing us with high-speed services. Additionally, frequency-domain analysis of moving average filters would provide us with a better theoretical insight into the cutoff frequency properties of finance signal modeling and group delay properties.

10. Conclusion

This study shows that structured multi-lag moving average filtering can improve the forecasting stability of financial time-series analysis by balancing noise reduction and trend preservation. The Hull Moving Average (HMA) shows a good performance in the short and medium scenarios, most probably because of its less phase lag and more responsiveness. In contrast, the Adaptive Moving Average (AMA) shows relatively better performance in modeling long-term trends through its adaptive smoothing technology. The results reveal that the selection of the lag-window is key in obtaining an optimal trade-off between noise suppression and signal responsiveness. The 25-day lag configuration, in particular, strikes a well-suited compromise between filtering efficiency and phase distortion, thus improving predictive consistency across datasets.

Using paired t-tests with moderate effect size, the improvement of HMA over SMA at 25-day lag is a statistical fact. Yet, statistical comparisons more generally across all filter and lag specifications were not undertaken and represent a critical avenue for future work. A comprehensive assessment of filtering performance from the perspectives of signal processing and forecasting focused on the multi-metric evaluation framework with an emphasis on the MSE, MAPE, R^2 and directional accuracy. Although the proposed framework is computationally efficient and shows consistent empirical performance, practical application in real-world trading or financial decision systems needs additional verification. Going forward, future work should include walk-forward testing, transaction cost analysis, and robustness testing with respect to market regimes.

In all, this work offers a reproducible engineering framework for lag-dependent smoothing in financial time-series forecasting that can be potentially extended into more complex modelling and deployment scenarios.

References

[1] M. Singh, H. Tunga, S. Kar, SentiVol-GA: a volatility-scaled genetic fusion of predictive models and financial sentiment for adaptive stock forecasting. International Journal of Data Science and Analytics, 22(16), (2026). https://doi.org/10.1007/s41060-025-00983-w

[2] L. Hu, Y. Shen, A predictive analytics approach for forecasting global stock index returns using deep learning techniques. Decision Analytics Journal, (2026) 100685.

[3] H.D. Le, K.N. Nguyen, Application of generalised regression neural network for financial time series forecasting: a comprehensive comparison with autoregressive integrated moving average. International Journal of Data Analysis Techniques and Strategies, 18(1), (2026) 1–24. https://doi.org/10.1504/IJDATS.2026.151640

[4] T. Berger, Deep learning and econometric time series analysis: An assessment of daily return forecasts. Journal of Forecasting, 45(1), 377–390. https://doi.org/10.1002/for.70045

[5] A. Alrawajfi, M.T. Ismail, S. Al Wadi, S. Atiewi, Enhancing Ethereum-USD Close Price Predictions through Hybrid ARIMA and Random Forest Model. International Journal of Intelligent Engineering & Systems, 18(11), (2025).

[6] W. Sun, Z. Liu, C. Yuan, X. Zhou, Y. Pei, C. Wei, RCSAN residual enhanced channel spatial attention network for stock price forecasting. Scientific Reports, 15(1), (2025) 21800. https://doi.org/10.1038/s41598-025-06885-y

[7] K. Yan, Z. Yue, C.C. Wu, Q. He, J. Zhou, Z. Hao, Y. Li, Flexible Target Prediction for Quantitative Trading in the American Stock Market: A Hybrid Framework Integrating Ensemble Models. Fusion Models and Transfer Learning, Entropy, 28(1), (2026) 84. https://doi.org/10.3390/e28010084

[8] A. Yaakoub, H. Oukhouya, M. Elhia, T. Zari, R. Guerbaz, Deep Learning for Financial Time Series: Does LSTM Outperform ARIMA and SVR in International Stock Market Predictions?. Statistics, Optimization & Information Computing, 14(5), (2025) 2181–2199. https://doi.org/10.19139/soic-2310-5070-2602

[9] P.P. Kadu, G.R. Bamnote, An Accurate Stock Price Prediction Using Deep Recurrent Neural Network with Search and Rescue Approach. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 33(04), (2025) 433–457. https://doi.org/10.1142/S0218488525500187

[10] X. Wang, Y. Wang, Y. Qi, J. Gao, F. Yang, J. Lu, An Ultra-Short-Term Wind Power Prediction Method Based on the Fusion of Multiple Technical Indicators and the XGBoost Algorithm. Energies, 18(12), (2025) 3069. https://doi.org/10.3390/en18123069

[11] X. Liu, Z. Wu, J. Xin, Forecasting stock market time series through the integration of bee colony optimizer and multivariate empirical mode decomposition with extreme gradient boosting regression. Engineering Applications of Artificial Intelligence, 148, (2025) 110353. https://doi.org/10.1016/j.engappai.2025.110353

[12] J. Wang, Y. Wang, Forecasting Expected Shortfall and Value-at-Risk With Cross-Sectional Aggregation. Journal of Forecasting, 44(2), (2025) 391–423. https://doi.org/10.1002/for.3195

[13] M. Aamir, H. Iftikhar, J. Nasir, P.C. Rodrigues, A.A. Alharbi, J. Allohibi, A novel hybrid LMD-SPF forecasting framework for financial time series: Evidence from gold returns. AIMS Mathematics, 10(9), (2025) 21875–21901. https://www.aimspress.com/article/doi/10.3934/math.2025974

[14] K. Noor, U. Fatima, Meta learning strategies for comparative and efficient adaptation to financial datasets. IEEE Access, 13, (2024) 24158–24170. https://doi.org/10.1109/ACCESS.2024.3516490

[15] A.U. Haq, A. Zeb, Z. Lei, D. Zhang, Forecasting daily stock trend using multi-filter feature selection and deep learning. Expert Systems with

Applications, 168, (2020) 114444. https://doi.org/10.1016/j.eswa.2020.114444

[16] N.G. Bagautdinova, E.I. Kadochnikova, A.N. Bakirova, Predictive autoregressive models of the Russian stock market using macroeconomic variables. International Journal of Criminology and Sociology, 9, (2020) 2439–2449. https://doi.org/10.6000/1929-4409.2020.09.296

[17] C. Cappello, A. Congedi, S. De Iaco, L. Mariella, Traditional prediction techniques and machine learning approaches for financial time series analysis. Mathematics, 13(3), (2025) 537. https://doi.org/10.3390/math13030537

[18] Y. Kojima, L. Ma, K. Nomoto, M. Hasegawa, S. Hirobayashi, Extraction of Periodic Characteristics and Long-Term Stock Price Forecasting Using Non-Harmonic Analysis With Over 14 Years of NASDAQ Data Before and After COVID-19. IEEE Access, 13, (2025) 91524 – 91532. https://doi.org/10.1109/ACCESS.2025.3571589

[19] T.S. Madhulatha, D.M.A.S. Ghori, Forecasting Currency Exchange Rates Using ARIMA, ETS and RNN: A Machine Learning Perspective. Journal of Theoretical and Applied Information Technology, 103(15), (2025).

[20] N. Rivera Garzón, S.A. Calderón Villanueva, O. Espinosa, Forecasting based on a multivariate autoregressive threshold model (MTAR) with a multivariate Student's t error distribution: A Bayesian approach. Communications in Statistics – Theory and Methods, 54(22), (2025) 7084–7104. https://doi.org/10.1080/03610926.2025.2466738

[21] C.J. Ogbonna, C.J. Ohabuka, D.C. Bartholomew, K.E. Anyiam, I. Adamu, Optimizing Nigerian Bank Lending Systems: The Power of Discrete Wavelet Transform (DWT) in Denoising and Regression Analysis. Annals of Data Science, (2025) 1–37. https://doi.org/10.1007/s40745-025-00626-8

[22] F. Afzal, F. Afzal, M. Kamran, H. Pan, Stock market predictive modeling using recurring neural network with long short-term architecture. International Journal of Islamic and Middle Eastern Finance and Management, 19(3), (2026) 719-739. https://doi.org/10.1108/IMEFM-12-2024-0632

[23] A.A. Dar, A. Jain, M. Malhotra, A.R. Farooqi, O. Albalawi, M.S. Khan, Hiba, Time Series analysis with ARIMA for historical stock data and future projections. Soft Computing, 28(21), (2024) 12531–12542. https://doi.org/10.1007/s00500-024-10309-w

[24] S. Behera, A.P. Kumar, S.C. Nayak, Improved set algebra-based heuristic technique for training multiplicative functional link artificial neural networks for financial time series forecasting. SN Computer Science, 5(5), 567. https://doi.org/10.1007/s42979-024-02902-5

[25] W.D. Wan, Y.L. Bai, Y.N. Lu, L. Ding, A hybrid model combining a gated recurrent unit network based on variational mode decomposition with error correction for stock price prediction. Cybernetics and Systems, 55(5), (2024) 1205–1229. https://doi.org/10.1080/01969722.2022.2137634

[26] J. Cheng, S. Tiwari, D. Khaled, M. Mahendru, U. Shahzad, Forecasting Bitcoin prices using artificial intelligence: Combination of ML, SARIMA, and Facebook Prophet Models. Technological Forecasting and Social Change, 198, (2024) 122938. https://doi.org/10.1016/j.techfore.2023.122938

[27] S. Akhtar, M. Ramzan, S. Shah, I. Ahmad, M.I. Khan, S. Ahmad, H. Qureshi, Forecasting exchange rate of Pakistan using time series analysis. Mathematical Problems in Engineering, 2022(1), (2022) 9108580. https://doi.org/10.1155/2022/9108580

[28] P.K. Nagula, C. Alexakis, A novel machine learning approach for predicting the NIFTY50 index in India. International Advances in Economic Research, 28(3), (2020) 155–170. https://doi.org/10.1007/s11294-022-09861-8

[29] J. Zheng, Y. Tian, J. Luo, T. Hong, A novel hybrid method based on kernel-free support vector regression for stock indices and price forecasting. Journal of the Operational Research Society, 74(3), (2023) 690–702. https://doi.org/10.1080/01605682.2022.2128908

[30] N. Yadav Vanguri, S. Pazhanirajan, T. Anil Kumar, Extraction of technical indicators and data augmentation-based stock market prediction using deep LSTM integrated competitive swarm feedback algorithm. International Journal of Information Technology & Decision Making, (2023) 1–27. https://doi.org/10.1142/S0219622023500049

[31] S.A. Farimani, M.V. Jahan, A.M. Fard, An adaptive multimodal learning model for financial market price prediction. IEEE Access, 12, (2024) 121846–121863. https://doi.org/10.1109/ACCESS.2024.3441029

[32] U. Ejder, S.A. Özel, A novel distance-based moving average model for improvement in the predictive accuracy of financial time series. Borsa Istanbul Review, 24(2), (2024) 376–397. https://doi.org/10.1016/j.bir.2024.01.011

[33] S. Thakar, D. Patel, V. C. Gandhi, D. Trivedi, (2024) Financial Analytics with Artificial Neural Networks: Predicting Loan Repayment. 8th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), IEEE, Kirtipur, Nepal. https://doi.org/10.1109/I-SMAC61858.2024.10714849

[34] C. Spulbar, C.C. Ene, Predictive Analytics in

Finance Using the ARIMA Model: Application for Bucharest Stock Exchange Financial Companies Closing Prices. Studies in Business and Economics, 19(3), (2024) 30–49. https://doi.org/10.2478/sbe-2024-0042

[35] N. Das, B. Sadhukhan, S.S. Bhakta, S. Chakrabarti, Integrating EEMD and ensemble CNN with X (Twitter) sentiment for enhanced stock price predictions. Social Network Analysis and Mining, 14(29), (2024). https://doi.org/10.1007/s13278-023-01190-w

[36] S. Mugisha, M. Wassajja, G.M. Bamutiire, J. Bashabe, M.D. Buhari, Review and analysis on digital filter design in digital signal processing. KIU Journal of Science, Engineering and Technology, 3(2), (2024) 12–20. https://doi.org/10.59568/KJSET-2024-3-2-02

[37] H. Yu, L.J. Ming, S. R., S. Zhang, A hybrid model for financial time series forecasting—integration of EWT, ARIMA with the improved ABC optimized ELM. IEEE Access, 8, (2020) 84501–84518. https://doi.org/10.1109/ACCESS.2020.2987547

[38] N. Zhang, A. Lin, P. Yang, Detrended moving average partial cross-correlation analysis on financial time series. Physica A: Statistical Mechanics and its Applications, 542, (2020) 122960. https://doi.org/10.1016/j.physa.2019.122960

[39] D.D. Sundararajan, Finite Impulse Response Filters. Digital Signal Processing, Springer, Cham. https://doi.org/10.1007/978-3-031-56740-7_7

Authors Contribution Statement

Milin Patel: conceptualization, methodology, software, data curation, formal analysis, visualization, Writing - Original Draft. Sweta S. Panchal: supervision, validation, investigation, Writing - Review & Editing. Neha Soni: methodology Methodology, Data Curation, Writing - Review & Editing. Sandip Kumar R. Panchal: supervision, project administration, validation, Writing - Review & Editing. All authors have read and agreed to the published version of the manuscript.

Funding

The authors declare that no funds, grants or any other support were received during the preparation of this manuscript.

Competing Interests

The authors declare that there are no conflicts of interest regarding the publication of this manuscript.

Has this article screened for similarity?

Yes

Data Availability

- The datasets used in this study are publicly available. Historical daily price data for the NIFTY 50 (^NSEI) and S&P 500 (^GSPC) indices were obtained from Yahoo Finance: https://finance.yahoo.com/
- Official historical data for the NIFTY 50 were collected from the National Stock Exchange (NSE) website: https://www.nseindia.com/.
- All data cover the period 2020–2024 and are freely accessible to the public.

About the License