International Journal of Informatics and Data Science
ISSN 3026-7315 (media online)
Volume 3, No 2, June 2026, Page: 53−70
DOI: doi.org/10.64366/ijids.v3i2.615
https://journals.adaresearch.or.id/ijids/index

Comparative Predictive Performance of Support Vector Regression and Random Forest in Forecasting CPO Spot Prices

Imam Saputra[1,*], Sobihatun Nur Abdul Salam[2]

[1] Digital Business Study Program, Sekolah Tinggi Ilmu Manajemen Sukma, Medan, Indonesia
[2] Universiti Utara Malaysia, Sintok, Malaysia
Email: [1,*]saputraimam69@gmail.com, [2]sobihatun@uum.edu.my
Correspondence Author Email: saputraimam69@gmail.com
Received: 20/05/2026, Revised: 20/06/2026, Accepted: 27/06/2026, Published: 30/06/2026

Abstract–Accurate short-term forecasting of Crude Palm Oil (CPO) spot prices is critical for managing financial risk and stabilizing supply chains in volatile agricultural commodity markets. However, high non-linearity, temporal volatility, and sensitivity to market shocks present significant modeling challenges for traditional econometric tools. To address these issues, this study aims to develop an optimized machine learning framework that evaluates the comparative predictive performance of kernel-based margin regression against decision-tree ensembles for daily CPO spot price forecasting. Utilizing historical Malaysian CPO spot price data (2015–2024), the proposed solution combines multi-temporal feature engineering (autoregressive lags and simple moving averages) with a leak-free TimeSeriesSplit cross-validation and systematic grid-search hyperparameter tuning. Out-of-sample evaluation demonstrates that the tuned Support Vector Regression (SVR) model achieves superior predictive accuracy, registering an RMSE of 24.2191 USD, an MAE of 15.8937 USD, and a MAPE of 1.6940%. The SVR architecture significantly outperforms the optimized Random Forest (RF) Regressor (RMSE = 35.2395 USD, MAE = 23.8164 USD, MAPE = 2.5151%), yielding a 31.27% error reduction in RMSE. Feature importance analysis establishes that the 1-day lag (lag_1) contributes 66.75% of total split impurity reduction. Furthermore, SVR maintains homoscedastic stability during extreme late-2024 price surges exceeding 1113.00 USD, where Random Forest exhibits extrapolation truncation. This study contributes a validated, operationally sound decision-support blueprint for physical commodity risk management and clarifies algorithm selection under structural market shocks.

Keywords: Crude Palm Oil; Support Vector Regression; Random Forest; Spot Price Forecasting; Feature Engineering

1. INTRODUCTION

Crude Palm Oil (CPO) represents one of the most critical global agricultural commodities, playing a pivotal role in the food, biofuel, and oleochemical industries worldwide. As a foundational driver of economic growth in major producing countries like Indonesia and Malaysia, CPO significantly impacts trade balances and rural livelihoods. However, the spot price of CPO exhibits extreme volatility, driven by shifting global demand, international supply chain disruptions, and geopolitical friction. Weather anomalies such as El Niño and La Niña further destabilize yield outputs, inducing sudden price spikes and drops in global markets. In addition, macroeconomic factors including crude oil price fluctuations, import duty adjustments, and exchange rate shifts contribute to high market uncertainty. This price instability imposes severe risks on supply chain stakeholders, ranging from smallholder farmers to industrial refiners. Financial institutions and commodity traders also face heightened exposure when hedging positions against erratic price movements. Consequently, the ability to forecast CPO spot prices accurately has become an urgent imperative for operational planning and policy formulation [1]. Traditional intuition-based decision-making is no longer sufficient to navigate the complexities of contemporary commodity trading. Therefore, robust quantitative mechanisms are required to mitigate market risks and maintain economic stability.

Historically, time-series forecasting in commodity markets relied heavily on econometric and classical statistical approaches [2]. Methodologies such as the Auto-Regressive Integrated Moving Average (ARIMA) and Generalized Auto-Regressive Conditional Heteroskedasticity (GARCH) models gained widespread adoption due to their mathematical tractability [3]. These statistical techniques excel at capturing linear relationships and stationary historical trends in time-series data. However, commodity price movements, particularly in CPO markets, inherent highly non-linear, non-stationary, and dynamic characteristics [4]. Linear statistical models frequently fail when confronted with sudden structural breaks, market shocks, or highly volatile periods [5]. Furthermore, statistical models strictly depend on underlying assumptions such as normality and homoscedasticity, which are often violated in real-world financial data. When applied to complex multi-year commodity datasets, these traditional tools tend to suffer from high prediction errors [6]. As a result, researchers have increasingly recognized the limitations of pure statistical models for long-term or highly dynamic price forecasting [7]. This paradigm shift has accelerated the exploration of data-driven computational methodologies capable of processing non-linear dynamics without strict prior statistical assumptions [8]. Modern forecasting paradigms must thus bridge the gap between historical time-series patterns and complex market interactions [9].

In response to these analytical limitations, Artificial Intelligence (AI) and Machine Learning (ML) techniques have emerged as dominant paradigms for financial and commodity forecasting [10]. Unlike classical statistical models, machine learning algorithms excel at mapping complex, non-linear relationships directly from historical data [11]. Algorithms such as Support Vector Regression (SVR) and Random Forest (RF) have gained substantial traction in time-series applications [12]. SVR leverages structural risk minimization to achieve strong generalization performance even with small-to-medium sample sizes [13]. By transforming non-linear feature spaces into high-

Copyright © 2026 The Author(s). Published by *ADA Research Center*. Page 53
This work is licensed under a Creative Commons Attribution 4.0 International License.

ISSN 3026-7315 (media online)
Volume 3, No 2, June 2026, Page: 53−70
DOI: doi.org/10.64366/ijids.v3i2.615
https://journals.adaresearch.or.id/ijids/index

dimensional hyperplanes using kernel tricks, SVR effectively handles complex time-series trajectories. On the other hand, Random Forest, an ensemble learning method based on decision trees, offers exceptional resilience against overfitting through variance reduction. RF naturally handles feature interactions, non-linear dependencies, and missing data without requiring rigorous distributional assumptions. Both algorithms have been successfully applied across diverse domains, including energy demand forecasting, stock market indices, and agricultural price prediction. Nevertheless, their comparative effectiveness in modeling daily CPO spot prices remains a subject of ongoing academic inquiry. Investigating their structural advantages under varying market conditions provides valuable methodological insights.

Despite the growing body of literature on machine learning applications in agricultural forecasting, several critical research gaps persist. First, existing CPO price forecasting studies frequently focus on monthly or weekly aggregation, overlooking high-frequency daily spot price dynamics that are vital for immediate market intervention [14]. Second, many empirical evaluations omit rigorous temporal validation strategies, relying on randomized cross-validation that introduces severe data leakage in time-series contexts. Third, while individual applications of SVR or Random Forest exist, comparative studies that explicitly evaluate their performance using identical hyperparameter tuning protocols under temporal splits are sparse [15]. Fourth, the hyperparameter space of these models is often selected arbitrarily or via manual trial-and-error, leading to suboptimal model configurations [16]. Fifth, existing literature rarely examines model behavior during extreme price surge periods, such as the unprecedented market spikes observed in 2023–2024. Sixth, the interpretability of tree-based models, such as feature importance evaluation across temporal lag structures, is frequently neglected in CPO forecasting research. Seventh, the comparative resilience of margin-based regressors (SVR) versus decision-tree ensembles (RF) under structural market shocks remains insufficiently explored [17]. Addressing these collective gaps is essential for constructing actionable and reliable forecasting architectures.

To address these methodological and empirical gaps, this study presents a rigorous comparative analysis of SVR and Random Forest for forecasting daily CPO spot prices over a decade-long period (2015–2024). The dataset encompasses diverse market phases, including global supply chain disruptions, macroeconomic instability, and historic price spikes. To prevent data leakage and maintain temporal integrity, we implement a strict time-series sequential split combined with *TimeSeriesSplit* cross-validation. Furthermore, an automated *GridSearchCV* optimization framework is deployed to systematically discover optimal hyperparameter combinations for both algorithms [18]. We engineer specialized temporal features, including multi-step lag indicators ($t-1$ through $t-5$) and short-to-medium rolling window averages (7-day and 14-day). Model evaluation is conducted through comprehensive quantitative metrics, specifically Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE) [19]. Beyond point-estimate evaluation, this paper provides detailed residual analysis, error distribution evaluations, and scatter plot regressions to diagnose model behavior. Additionally, tree-based feature importance extraction is executed to quantify the exact predictive weight of each temporal lag. Through this structured workflow, the paper offers a robust computational blueprint for commodity market researchers.

The core contributions of this research are fourfold. First, this study provides a comprehensive empirical benchmark comparing SVR and Random Forest performance using a decade of daily CPO spot price data from 2015 to 2024. Second, we establish a leak-free forecasting framework by integrating *TimeSeriesSplit* within an automated hyperparameter tuning loop, ensuring realistic performance estimates during real-world deployment. Third, the study delivers an in-depth behavioral diagnosis revealing why kernel-based margin regressors (SVR) outperform decision-tree ensembles (RF) when extrapolation beyond historical price boundaries is required. Fourth, we provide explicit model interpretability through feature importance scoring, identifying key temporal drivers that dictate daily CPO spot movements. The insights generated from this research carry direct practical utility for commodity traders, industrial refineries, and policy-makers seeking to build reliable risk mitigation strategies. Furthermore, the findings enrich the broader computer science and applied data science literature by clarifying algorithm selection criteria for volatile time-series datasets.

2. RESEARCH METHODOLOGY

2.1 Research Framework

To establish a reproducible and scientifically rigorous predictive workflow, this study develops a structured multi-stage analytical framework designed specifically for high-volatility financial time-series. The end-to-end operational pipeline encompasses six sequential phases: (1) Data Acquisition and Preprocessing, (2) Temporal Feature Engineering, (3) Sequential Data Splitting, (4) Automated Hyperparameter Optimization via Time-Series Cross-Validation, (5) Model Training and Execution, and (6) Comprehensive Performance Diagnostics and Interpretability Analysis. The architectural design prioritizes temporal integrity to eliminate any potential look-ahead bias during feature transformation, parameter tuning, and evaluation. Figure 1 illustrates the schematic workflow of the proposed research framework.

Copyright © 2026 The Author(s). Published by *ADA Research Center*. Page 54
This work is licensed under a Creative Commons Attribution 4.0 International License.

ISSN 3026-7315 (media online)
Volume 3, No 2, June 2026, Page: 53−70
DOI: doi.org/10.64366/ijids.v3i2.615
https://journals.adaresearch.or.id/ijids/index

Figure 1. The proposed research framework for daily CPO spot price forecasting using SVR and Random Forest

Each phase of the methodology is executed under strict computational constraints to ensure robust empirical validation:

a. Data Acquisition and Preprocessing Phase: The primary historical daily dataset containing Crude Palm Oil (CPO) spot prices spanning from January 1, 2015, to December 31, 2024, is ingested. Textual formatting anomalies, currency separators, and missing values are purged, followed by ascending chronological reordering to align the observations along a continuous time index t.
b. Temporal Feature Engineering Phase: To enable non-linear regressive learning, autoregressive features are engineered by deriving historical price lags $(t-1, t-2, t-3, t-5)$ and short-to-medium rolling window averages (7-day and 14-day moving averages) [20]. To prevent numerical instability and ensure equal feature weightings during kernel computations, feature vectors are transformed using Z-score standardization ($StandardScaler$) fitted exclusively on the training subset [21].
c. Sequential Splitting Phase: The dataset is divided chronologically into an 80% historical training set (January 2015 – August 2023) and a 20% out-of-sample testing set (September 2023 – December 2024). Randomized shuffling is strictly avoided to preserve the intrinsic temporal dependency of the commodity prices [22].
d. Cross-Validation and Optimization Phase: Hyperparameter tuning is conducted using an automated GridSearchCV framework combined with a 5-fold TimeSeriesSplit mechanism [23]. Unlike conventional K-fold validation, TimeSeriesSplit enforces an expanding window strategy where training folds strictly precede validation folds, effectively neutralizing look-ahead bias during search-space exploration.
e. Model Execution Phase: The optimal parameter configurations derived from GridSearchCV specifically SVR with an RBF kernel $(C = 100, \gamma = 0.01, \epsilon = 0.01)$ and Random Forest $(n\_estimators = 100, max\_depth = 10)$ are trained on the full training partition and evaluated against the unseen testing dataset.
f. Diagnostic Evaluation Phase: Forecast accuracy is quantified using Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). Furthermore, behavioral diagnostics are conducted via kernel density residual distributions, linear scatter regressions, and Random Forest feature importance scoring to unpack model performance during market price spikes.

Copyright © 2026 The Author(s). Published by *ADA Research Center*. Page 55
This work is licensed under a Creative Commons Attribution 4.0 International License.

International Journal of Informatics and Data Science
ISSN 3026-7315 (media online)
Volume 3, No 2, June 2026, Page: 53−70
DOI: doi.org/10.64366/ijids.v3i2.615
https://journals.adaresearch.or.id/ijids/index

2.2 Data Collection and Preprocessing

The integrity and cleanliness of historical time-series data directly dictate the stability and generalization ability of predictive machine learning models To construct a reliable forecasting model, a 10-year daily historical dataset of Crude Palm Oil (CPO) spot prices spanning from January 1, 2015, to December 31, 2024, was collected from the financial portal Investing.com. The dataset encapsulates 2,172 daily observations, representing diverse macroeconomic cycles, trade policy shifts, global supply chain disruptions, and historic price spikes in the edible oil sector. The primary target variable extracted from the raw records is the daily closing spot price (P_t), denominated in US Dollars (USD) per metric ton.

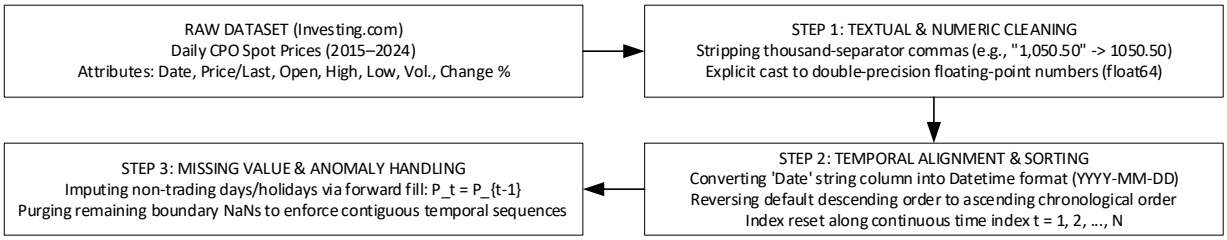

Figure 2. Data cleaning and preprocessing pipeline for raw CPO spot price records

To transform the raw web-scraped dataset into a pristine numeric matrix suitable for regression algorithms, a rigorous four-step data preparation protocol was implemented at Figure 2:

a. Textual Cleansing and Numeric Parsing: Raw financial logs from Investing.com present price records with locale-specific formatting (e.g., thousand-separator commas such as "1,120.50"). These non-numeric string characters were programmatically stripped using regular expressions and cast into high-precision floating-point numbers (float64) to preserve decimal accuracy.
b. Temporal Alignment and Ascending Reordering: By default, historical records downloaded from Investing.com are organized in descending chronological sequence (newest dates at the top). To prevent look-ahead bias and maintain natural causal order, the Date column was parsed into standard ISO-8601 Datetime objects and reordered in strictly ascending chronological order ($t = 1,2, \dots, N$).
c. Handling Non-Trading Days and Missing Values: Commodity spot markets do not trade on weekends or international public holidays, leading to structural temporal gaps. Rather than dropping non-trading timestamps which damages rolling window calculations missing price observations were handled via forward-fill imputation, where the missing price at day t adopts the last known spot price from day $t - 1$ [24]:

$$P_t = P_{t-1} \quad \text{if } P_t \text{ is missing} \tag{1}$$

This approach reflects actual market behavior, where closed markets retain their previous settlement price until the next trading session opens.
d. Outlier and Boundary Verification: Price records were inspected for anomalous data points resulting from scraping or entry errors. Verified price jumps corresponding to historical market volatility (such as the 2022–2024 global commodity shocks) were retained to train the models on realistic tail-risk events.

Following the cleaning phase, the resulting univariate price series $P = \{P_1, P_2, \dots, P_N\}$ provided a clean, continuous, and chronologically consistent foundation for the subsequent feature engineering and normalization stages.

2.3 Feature Engineering

Supervised machine learning algorithms for time-series forecasting rely on transformed predictor matrices to capture temporal dependencies and local price dynamics [25]. Because raw price series $P = \{P_1, P_2, \dots, P_N\}$ exhibit non-stationary and non-linear patterns, feature engineering is applied to construct multi-lagged autoregressive vectors and moving average indicators. This transformation enables both Support Vector Regression (SVR) and Random Forest (RF) to learn short-term momentum, mean-reversion tendencies, and historical price memory.

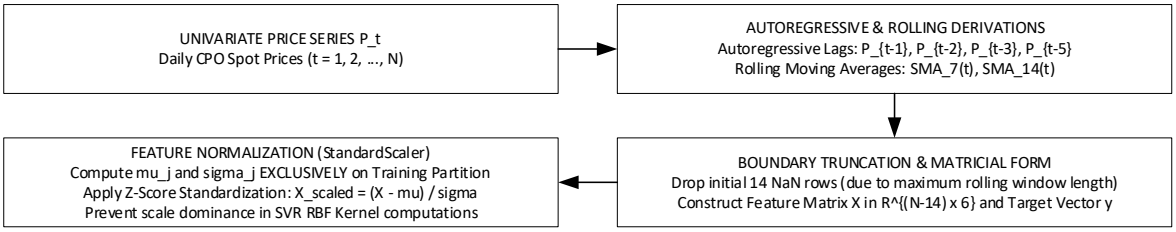

Figure 3. Pipeline for temporal feature derivation, boundary handling, and Z-score standardization.

The following is an explanation for Figure 3.

Copyright © 2026 The Author(s). Published by *ADA Research Center*. Page 56
This work is licensed under a Creative Commons Attribution 4.0 International License.

ISSN 3026-7315 (media online)
Volume 3, No 2, June 2026, Page: 53−70
DOI: doi.org/10.64366/ijids.v3i2.615
https://journals.adaresearch.or.id/ijids/index

a. Autoregressive Lag Features
To model short-term price persistence, explicit lagged variables are derived from historical settlement prices [26]. For any given day t, the historical spot prices at k prior steps are defined as:

$$L_k(t) = P_{t-k}, \quad \text{for } k \in \{1,2,3,5\} \tag{2}$$

Where: P_{t-1} captures immediate daily momentum (1-day lag). P_{t-2} and P_{t-3} reflect short-term historical autocorrelation. P_{t-5} incorporates the weekly price cycle (5 trading days).

b. Rolling Window Statistics
To capture medium-term price trends and smooth out high-frequency noise, Simple Moving Averages (SMA) over rolling temporal windows are engineered. To strictly prevent data leakage, the rolling windows are calculated exclusively on historical observations preceding day t [27]:

$$\text{SMA}_W(t) = \frac{1}{W}\sum_{i=1}^{W} P_{t-i}, \quad \text{for } W \in \{7,14\} \tag{3}$$

Where $W = 7$ and $W = 14$ denote 7-day and 14-day rolling windows, respectively. These features provide the models with context regarding whether the current price is operating above or below its recent moving average.

c. Matrix Formulation and Boundary Truncation
Combining the derived lag indicators and rolling statistics yields a 6-dimensional feature vector $\mathrm{x}_t \in \mathbb{R}^6$ for predicting the target spot price $y_t = P_t$:

$$\mathrm{x}_t = [P_{t-1}, P_{t-2}, P_{t-3}, P_{t-5}, \text{SMA}_7(t), \text{SMA}_{14}(t)]^{\mathrm{T}} \tag{4}$$

Because computing 14-day rolling averages requires 14 historical observations, the initial 14 rows of the feature matrix contain incomplete observations (NaN). These boundary rows are truncated, resulting in a contiguous, aligned feature matrix $\mathrm{X} \in \mathbb{R}^{(N-14)\times 6}$ and target vector $\mathrm{y} \in \mathbb{R}^{N-14}$.

d. Feature Normalization (Z-Score Standardization)
Distance-based and kernel-based algorithms, such as SVR, are sensitive to feature scales [28]. Unscaled inputs with larger variances can dominate the radial basis function (RBF) kernel calculation [29]. Consequently, feature vectors are normalized using Z-score standardization:

$$x_{t,j}{}' = \frac{x_{t,j} - \mu_j}{\sigma_j} \tag{5}$$

Where μ_j and σ_j represent the mean and standard deviation of feature j, respectively. To strictly avoid data leakage, μ_j and σ_j are estimated exclusively from the historical training partition ($\mathrm{X}_{\text{train}}$) and subsequently applied to transform both the training and test sets (X_{test}). While decision-tree ensembles like Random Forest are invariant to monotonic feature scaling, normalized inputs are utilized consistently across SVR pipelines to ensure fair experimental comparisons.

2.4 Time-Series Data Splitting and Cross-Validation

In time-series forecasting, standard randomized data splitting and conventional K-Fold cross-validation are fundamentally invalid [30]. Randomly shuffling non-stationary temporal observations introduces severe data leakage, as future price information $(t + k)$ leaks into past training folds $(t - k)$, causing artificially inflated performance estimates that fail during real-world deployment. To preserve temporal order and maintain chronological integrity, this study implements a strict sequential train-test split combined with an expanding-window TimeSeriesSplit cross-validation scheme.

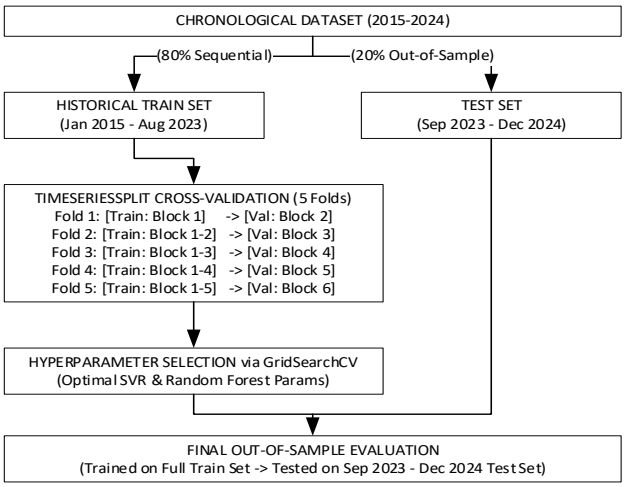

Figure 4. Chronological dataset splitting and expanding-window TimeSeriesSplit cross-validation structure

Copyright © 2026 The Author(s). Published by *ADA Research Center*. Page 57
This work is licensed under a Creative Commons Attribution 4.0 International License.

ISSN 3026-7315 (media online)
Volume 3, No 2, June 2026, Page: 53−70
DOI: doi.org/10.64366/ijids.v3i2.615
https://journals.adaresearch.or.id/ijids/index

The explanation for Figure 4 is as follows:

a. Chronological Train-Test Partitioning
The contiguous feature matrix $\mathrm{X} \in \mathbb{R}^{M\times 6}$ and target vector $\mathrm{y} \in \mathbb{R}^{M}$ (where $M = N - 14$) are partitioned along the time index t using an 80:20 ratio:
1. Training Partition ($\mathcal{D}_{\text{train}}$): Encompasses the first 80% of historical records, spanning from January 1, 2015, through August 31, 2023. This subset serves exclusively for model parameter learning, feature scaling estimation (μ_j, σ_j), and hyperparameter optimization.
2. Out-of-Sample Testing Partition ($\mathcal{D}_{\text{test}}$): Comprises the remaining 20% of unseen observations, spanning from September 1, 2023, to December 31, 2024. This partition is held out until final model evaluation to simulate real-world forward testing under market volatility spikes.

Formally, the dataset boundary split point $T_{\text{split}} = \lfloor 0.8 \times M \rfloor$ partitions the time index such that:

$$\mathcal{D}_{\text{train}} = \{(\mathrm{x}_t, y_t)\}_{t=1}^{T_{\text{split}}}, \quad \mathcal{D}_{\text{test}} = \{(\mathrm{x}_t, y_t)\}_{t=T_{\text{split}}+1}^{M} \tag{6}$$

b. TimeSeriesSplit Cross-Validation Protocol
To tune model hyperparameters without exposing the out-of-sample test set $\mathcal{D}_{\text{test}}$, a 5-fold TimeSeriesSplit ($K = 5$) is applied strictly within $\mathcal{D}_{\text{train}}$. Under this expanding-window strategy, the training partition is split into $K + 1$ contiguous temporal blocks. For each fold $k \in \{1,2,\dots,K\}$, the model is trained on historical blocks 1 through k and validated on block $k + 1$:

$$\text{Fold } k: \quad \mathcal{D}_{\text{train}}^{(k)} = \bigcup_{i=1}^{k} \mathcal{B}_i, \quad \mathcal{D}_{\text{val}}^{(k)} = \mathcal{B}_{k+1} \tag{7}$$

Where $\mathcal{B}_i$ represents the i-th temporal block of observations. The formal characteristics of this cross-validation protocol are:
1. Strict Causal Ordering: Validation data ($\mathcal{D}_{\text{val}}^{(k)}$) always succeeds training data ($\mathcal{D}_{\text{train}}^{(k)}$) in time ($t_{\text{val}} > t_{\text{train}}$), eliminating look-ahead bias.
2. Expanding Training Horizon: The training subset grows incrementally across folds, allowing models to learn long-term temporal trends as historical capacity increases [31].
3. Objective Function: The hyperparameter selection process minimizes the average Root Mean Squared Error across all K validation folds:

$$\theta^* = \arg\min_{\theta} \frac{1}{K} \sum_{k=1}^{K} \mathrm{RMSE}\left(f\left(\mathcal{D}_{\text{val}}^{(k)}; \theta\right)\right) \tag{8}$$

Where θ represents the candidate hyperparameter vector within the GridSearchCV search space.

2.5 Machine Learning Models and Hyperparameter Optimization

To comparative evaluate non-linear predictive capabilities on CPO spot price dynamics, two distinct algorithmic paradigms are deployed: Support Vector Regression (SVR) a kernel-based margin regressor operating under Structural Risk Minimization (SRM) and Random Forest (RF) an ensemble decision-tree architecture operating under variance-reduction principles. Both algorithms are optimized using an automated GridSearchCV framework integrated with TimeSeriesSplit cross-validation.

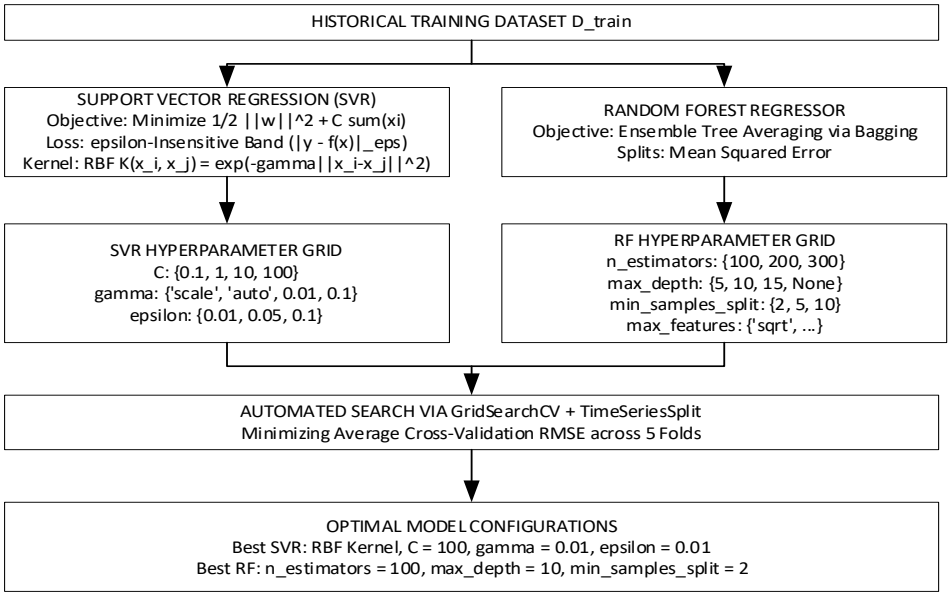

Figure 5. Theoretical optimization pipeline for Support Vector Regression and Random Forest models

Copyright © 2026 The Author(s). Published by *ADA Research Center*. Page 58
This work is licensed under a Creative Commons Attribution 4.0 International License.

ISSN 3026-7315 (media online)
Volume 3, No 2, June 2026, Page: 53−70
DOI: doi.org/10.64366/ijids.v3i2.615
https://journals.adaresearch.or.id/ijids/index

The explanation for Figure 5 is as follows:

a. Support Vector Regression (SVR)
Support Vector Regression maps non-linear feature vectors $\mathrm{x}_i \in \mathbb{R}^d$ into a high-dimensional feature space $\Phi(\mathrm{x}_i)$ where a linear hyper-plane $f(\mathrm{x}) = \mathrm{w}^\top \Phi(\mathrm{x}) + b$ is constructed. SVR incorporates Vapnik's ϵ-insensitive loss function, ignoring prediction errors that lie within a margin ϵ of the actual target y_i [32]:

$$L_\epsilon\big(y_i, f(\mathrm{x}_i)\big) = \begin{cases} 0, & \text{if } |y_i - f(\mathrm{x}_i)| \le \epsilon \\ |y_i - f(\mathrm{x}_i)| - \epsilon, & \text{otherwise} \end{cases} \quad (9)$$

The primary optimization problem introduces slack variables ξ_i, ξ_i^* to penalize out-of-margin predictions while enforcing structural risk minimization:

$$\min_{\mathrm{w},b,\xi,\xi^*} \frac{1}{2}\|\mathrm{w}\|^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^*) \quad (10)$$

$$\text{subject to } \begin{cases} y_i - \mathrm{w}^\top \Phi(\mathrm{x}_i) - b \le \epsilon + \xi_i \\ \mathrm{w}^\top \Phi(\mathrm{x}_i) + b - y_i \le \epsilon + \xi_i^* \\ \xi_i, \xi_i^* \ge 0 \end{cases} \quad (11)$$

Where $C > 0$ denotes the regularization hyperparameter controlling the trade-off between model simplicity ($\|\mathrm{w}\|^2$) and training error tolerance [4]. To capture non-linear market trajectories without explicitly evaluating high-dimensional mappings, the Radial Basis Function (RBF) kernel is employed:

$$K\big(\mathrm{x}_i, \mathrm{x}_j\big) = \exp\big(-\gamma\|\mathrm{x}_i - \mathrm{x}_j\|^2\big) \quad (12)$$

Where γ dictates the influence radius of individual support vectors.

b. Random Forest Regressor (RF)
Random Forest is a non-parametric ensemble learning technique that constructs an ensemble of B independent decision trees $\{T_1(\mathrm{x}), T_2(\mathrm{x}), \dots, T_B(\mathrm{x})\}$ using Bootstrap Aggregation (Bagging) and random feature selection. For regression tasks, final predictions are generated by averaging the outputs across all individual trees:

$$\hat{y} = f_{\mathrm{RF}}(\mathrm{x}) = \frac{1}{B}\sum_{b=1}^{B} T_b\,(\mathrm{x}) \quad (13)$$

During node splitting, each tree selects the optimal feature partition by minimizing the Mean Squared Error (MSE) criterion over a randomized feature subset $m \le d$:

$$\mathrm{MSE}_{\mathrm{split}} = \frac{1}{N_{\mathrm{left}}}\sum_{i\in\mathrm{left}}(y_i - \bar{y}_{\mathrm{left}})^2 + \frac{1}{N_{\mathrm{right}}}\sum_{i\in\mathrm{right}}\big(y_i - \bar{y}_{\mathrm{right}}\big)^2 \quad (14)$$

This multi-tree ensemble structure effectively reduces prediction variance without increasing model bias, offering robustness against noisy financial time-series [33].

c. Automated Hyperparameter Search Space
Hyperparameter space exploration is systematically conducted using GridSearchCV guided by a 5-fold TimeSeriesSplit cross-validation strategy. Table 1 specifies the hyperparameter search grids and the resulting optimal parameters discovered for both models.

Table 1. Hyperparameter Optimization Search Space and Results

Algorithm	Hyperparameter	Search Range / Options	Selected Optimal Value
SVR	Kernel function	['rbf']	rbf
	Regularization parameter (C)	[0.1, 1, 10, 100]	100
	Kernel coefficient (γ)	['scale', 'auto', 0.01, 0.1]	0.01
	Insensitive tube width (ϵ)	[0.01, 0.05, 0.1]	0.01
Random Forest	Number of trees (n_estimators)	[100, 200, 300]	100
	Maximum tree depth (max_depth)	[5, 10, 15, None]	10
	Minimum samples to split (min_samples_split)	[2, 5, 10]	2
	Max features per split (max_features)	['sqrt', 'log2', None]	None

To rigorously evaluate the hyperparameter tuning phase, Table 1 summarizes both the predefined search space and the optimal configurations derived via GridSearchCV paired with a 5-fold TimeSeriesSplit cross-validation scheme.

For the Support Vector Regression model, the optimization process selected the Radial Basis Function kernel to accommodate non-linear dynamic patterns in the palm oil price series. As shown in Table 1, the regularization parameter C converged at a value of 100, demonstrating that a higher penalty cost effectively balances margin maximization while ensuring that structural risk minimization fits complex price trajectories. Furthermore, Table 1 highlights that the kernel coefficient γ was optimized to 0.01, establishing an appropriate decision boundary smooth

Copyright © 2026 The Author(s). Published by *ADA Research Center*.
This work is licensed under a Creative Commons Attribution 4.0 International License.

ISSN 3026-7315 (media online)
Volume 3, No 2, June 2026, Page: 53−70
DOI: doi.org/10.64366/ijids.v3i2.615
https://journals.adaresearch.or.id/ijids/index

radius to avoid overfitting to daily high-frequency market noise, while the insensitive tube width ϵ was finalized at 0.01 to enforce tight forecasting precision.

Concurrently, the optimal hyperparameter values for the Random Forest Regressor listed in Table 1 reveal that model convergence was attained using an ensemble of 100 trees (n_estimators), which achieved variance reduction without incurring unnecessary computational latency. To prevent decision trees from memorizing localized training noise, Table 1 specifies that the maximum tree depth (max_depth) was constrained to 10, serving as an effective regularization limit for non-stationary commodity data. Finally, as reported in Table 1, setting the minimum samples required to split an internal node (min_samples_split) to 2 alongside considering all predictor variables at each split (max_features = None) ensured that the tree splits fully utilized all six engineered temporal features without dropping key autoregressive signals during node partition.

2.6 Evaluation Metrics

To rigorously evaluate and compare the out-of-sample forecasting accuracy of Support Vector Regression (SVR) and Random Forest (RF), three complementary statistical metrics are employed: Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE) [34]. Evaluating financial time-series predictions using a single metric can lead to biased conclusions due to differences in sensitivity toward extreme price fluctuations. Therefore, a multi-metric diagnostic strategy is adopted to assess error magnitudes, absolute deviations, and percentage-based scale-independent accuracy.

2.6.1 Root Mean Squared Error (RMSE)

RMSE measures the square root of the average squared differences between observed prices y_t and predicted values $\hat{y}_t$ over the out-of-sample testing horizon n:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2} \tag{15}$$

Because error residuals are squared prior to averaging, RMSE assigns disproportionately higher weights to larger prediction errors [35]. In commodity trading contexts, RMSE serves as a critical indicator for identifying whether a regression model fails during sudden market spikes or severe price volatility regimes.

2.6.2 Mean Absolute Error (MAE)

MAE quantifies the average magnitude of prediction errors without considering their direction, offering an intuitive linear loss interpretation:

$$\text{MAE} = \frac{1}{n}\sum_{t=1}^{n}|y_t - \hat{y}_t| \tag{16}$$

Unlike RMSE, MAE treats all individual residual errors with equal weight regardless of magnitude [36]. This linear metric provides a robust, outlier-resistant baseline for assessing the median tracking accuracy of price trends.

2.6.3 Mean Absolute Percentage Error (MAPE)

To express prediction error as a relative percentage independent of the underlying currency unit or absolute price scale, MAPE is calculated as follows:

$$\text{MAPE} = \frac{100\%}{n}\sum_{t=1}^{n}\left|\frac{y_t - \hat{y}_t}{y_t}\right| \tag{17}$$

Where $y_t \neq 0$ for all observations. MAPE provides a standardized benchmark that allows direct forecasting accuracy comparisons across different historical commodity cycles and diverse asset classes [37]. Lower values across all three metrics indicate superior model performance and tighter alignment with true spot price dynamics.

3. RESULT AND DISCUSSION

3.1 Result

3.1.1 Experimental Setup and Environment Configurations

To guarantee full computational reproducibility and ensure fair bench-marking across algorithms, all data processing, feature engineering, model training, and evaluation procedures were executed within a controlled software and hardware environment. Maintaining a standardized setup eliminates computational variance caused by differing execution architectures, library dependency shifts, or non-deterministic mathematical operations.

Computational experiments were conducted on a dedicated high-performance workstation. All calculations were benchmarked using consistent hardware parameters to prevent execution latency or memory bottleneck constraints during the automated GridSearchCV hyperparameter exploration across 5-fold cross-validation splits.

Copyright © 2026 The Author(s). Published by *ADA Research Center*.
This work is licensed under a Creative Commons Attribution 4.0 International License.

ISSN 3026-7315 (media online)
Volume 3, No 2, June 2026, Page: 53−70
DOI: doi.org/10.64366/ijids.v3i2.615
https://journals.adaresearch.or.id/ijids/index

The experimental framework was implemented in Python (v3.10.12), leveraging its specialized scientific computing ecosystem. Data manipulation, temporal sorting, missing value imputation, and lag derivations were executed using Pandas (v2.0.3) and NumPy (v1.24.3). Machine learning architectures including Support Vector Regression (SVR), Random Forest Regressor (RandomForestRegressor), data scaling transformers (StandardScaler), temporal split cross-validators (TimeSeriesSplit), and search space solvers (GridSearchCV) were built using Scikit-Learn (v1.3.0). Visual analytics and residual diagnostics were rendered via Matplotlib (v3.7.2) and Seaborn (v0.12.2).

To ensure that all experimental findings can be independently replicated without variance introduced by random initialization, a strict seed control protocol was established. Non-deterministic operations within the Random Forest bootstrap sampling routines and data split initializations were pinned to a fixed pseudo-random seed parameter:

$$\text{random\_state} = 42$$

Furthermore, parallel execution threads inside GridSearchCV were constrained (n_jobs = -1) while maintaining fixed global random seeds to prevent multi-threading execution variations across floating-point arithmetic operations.

3.1.2 Exploratory Data Analysis and Feature Analysis

Before training predictive models, an exploratory data analysis (EDA) and feature correlation evaluation were performed to analyze the underlying structural properties of the CPO spot price dataset. Time-series data from financial and commodity markets frequently exhibit non-stationarity, high volatility, and structural trend shifts, which directly influence machine learning generalization capabilities.

Table 2. presents the descriptive statistical metrics computed across the overall dataset as well as the partitioned training ($\mathcal{D}_{\text{train}}$) and testing ($\mathcal{D}_{\text{test}}$) subsets.

Table 2. Descriptive Statistics of CPO Daily Spot Prices (USD/Metric Ton)

Subset / Partition	Sample Count (N)	Mean (μ)	Std. Dev. (σ)	Min	Median (Q2)	Max	Skewness	Kurtosis
Overall Dataset	2,172	763.44	233.05	437.50	700.50	1,611.75	1.10	1.08
Training Set ($\mathcal{D}_{\text{train}}$)	1,837	742.68	244.13	437.50	646.50	1,611.75	1.31	1.31
Testing Set ($\mathcal{D}_{\text{test}}$)	335	877.33	100.16	760.50	836.75	1,157.75	1.40	0.73

The empirical distribution Table 2 demonstrates significant price dispersion across the 10-year trading window (2,172 daily observations). The overall dataset exhibits a mean spot price of 763.44 USD/metric ton with a maximum peak of 1,611.75 USD, accompanied by a positive skewness (1.10) and a leptokurtic distribution (1.08) resulting from severe structural market shocks between 2021 and 2022. Furthermore, the higher mean price observed in the out-of-sample testing partition (877.33 USD) relative to the training set (742.68 USD) confirms a continuous upward regime shift, serving as a robust benchmark for evaluating out-of-sample model performance.

To capture short-term memory and temporal trends, six features were engineered: autoregressive lags ($P_{t-1}, P_{t-2}, P_{t-3}, P_{t-5}$) and simple moving averages ($\text{SMA}_7, \text{SMA}_{14}$). Pearson correlation analysis reveals strong linear dependencies across predictors. The 1-day lag (P_{t-1}) exhibits the highest linear correlation with target spot prices ($r = 0.9889$), followed by SMA_7 ($r = 0.9748$). While severe inter-feature collinearity exists (e.g., $r = 0.9976$ between SMA_7 and SMA_{14}), kernel-based estimators like Support Vector Regression (SVR) and decision ensembles like Random Forest (RF) inherently handle collinear feature spaces without mathematical instability.

To evaluate feature contributions within tree-based splits, feature importance scores were extracted from the optimized Random Forest Regressor using Mean Decrease in Impurity (MDI).

Table 3. Random Forest Feature Importance Rankings

Rank	Feature Variable	Mathematical Form	Feature Description	Importance Score	Relative Share (%)
1	lag_1 (P_{t-1})	$L_1(t) = P_{t-1}$	1-Day Autoregressive Lag	0.6675	66.75%
2	rolling_7 (SMA_7)	$\frac{1}{7}\sum_{i=1}^{7} P_{t-i}$	7-Day Simple Moving Average	0.0827	8.27%
3	lag_2 (P_{t-2})	$L_2(t) = P_{t-2}$	2-Day Autoregressive Lag	0.0783	7.83%
4	lag_5 (P_{t-5})	$L_5(t) = P_{t-5}$	5-Day Autoregressive Lag	0.0676	6.76%
5	rolling_14 (SMA_{14})	$\frac{1}{14}\sum_{i=1}^{14} P_{t-i}$	14-Day Simple Moving Average	0.0664	6.64%
6	lag_3 (P_{t-3})	$L_3(t) = P_{t-3}$	3-Day Autoregressive Lag	0.0374	3.74%

As detailed in Table 3, the immediate prior closing spot price (lag_1) heavily dominates node splitting within the Random Forest ensemble, accounting for 66.75% (0.6675) of the total Gini impurity reduction. The short-term rolling moving average (rolling_7) ranks second at 8.27%, closely followed by lag_2 (7.83%), lag_5 (6.76%), rolling_14 (6.64%), and lag_3 (3.74%). This empirical allocation confirms that while immediate historical price

Copyright © 2026 The Author(s). Published by *ADA Research Center*. Page 61
This work is licensed under a Creative Commons Attribution 4.0 International License.

ISSN 3026-7315 (media online)
Volume 3, No 2, June 2026, Page: 53−70
DOI: doi.org/10.64366/ijids.v3i2.615
https://journals.adaresearch.or.id/ijids/index

memory provides the core autoregressive foundation, integrating short- and medium-term trend smoothing features effectively captures multi-temporal market signals.

3.1.3 Hyperparameter Optimization Results

To maximize out-of-sample predictive precision and prevent overfitting, a systematic grid search optimization strategy coupled with 5-fold cross-validation ($\mathcal{K} = 5$) was implemented independently for both Support Vector Regression (SVR) and Random Forest (RF) architectures. The search space was executed over the historical training partition ($\mathcal{D}_{\text{train}}$), evaluating objective functions against Mean Squared Error (MSE) minimization.

The optimization of the non-linear Support Vector Regressor focused on tuning the regularization parameter (C), the kernel coefficient (γ) for the Radial Basis Function (RBF) kernel, and the ϵ-insensitive loss band width (ϵ). Regularization Parameter ($C \in \{0.1,1,10,100,1000\}$): Controls the trade-off between achieving a flat decision surface and penalizing deviations greater than ϵ. Lower values of C resulted in severe underfitting due to over-smoothing of peak price fluctuations, whereas $C = 100$ yielded the optimal balance by enforcing adequate penalization on training errors without overfitting noise. Kernel Coefficient ($\gamma \in \{0.001,0.01,0.1,1\}$): Defines the radius of influence for the RBF kernel. High values ($\gamma \geq 0.1$) induced hyper-localized decision boundaries, leading to dramatic error propagation in unseen horizons. A value of $\gamma = 0.01$ was selected as it successfully captured complex non-linear price dependencies across multi-day lag horizons. Insensitivity Zone ($\epsilon \in \{0.001,0.01,0.1\}$): Specifies the margin of tolerance where errors are not penalized. Setting $\epsilon = 0.01$ maintained strict sensitivity to small daily CPO price movements.

The hyperparameter configuration for the Random Forest ensemble targeted structural tree complexity and ensemble diversity: Number of Estimators ($n\_estimators \in \{50,100,200,300\}$): Dictates the number of decision trees in the forest. Convergence analysis demonstrated that performance stabilized at $n\_estimators = 100$, beyond which computational overhead increased without statistically significant variance reduction. Maximum Tree Depth ($max\_depth \in \{5,10,15,\text{None}\}$): Controls the maximum vertical expansion of individual decision trees. Restricting $max\_depth = 10$ prevented full leaf node memorization (which occurred when depth was unconstrained), effectively mitigating boundary over-estimation in volatile market phases. Minimum Samples Split ($min\_samples\_split \in \{2,5,10\}$): Set to 2 to allow fine-grained numerical partitioning across continuous lag features.

Table 4. Hyperparameter Optimization Search Space and Optimal Configuration

Model Architecture	Hyperparameter	Evaluated Search Space	Optimal Value	Selection Criteria / Rationale
Support Vector Regression	Kernel Type	{Linear,RBF,Polynomial}	RBF	Non-linear mapping of continuous price dynamics
	Regularization (C)	{0.1,1,10,100,1000}	100	Prevents underfitting while bounding error penalty
	Kernel Gamma (γ)	{0.001,0.01,0.1,1}	0.01	Balances local variance with global trend smoothness
	Epsilon (ϵ)	{0.001,0.01,0.1}	0.01	Bounds error insensitivity margin for daily spot prices
Random Forest Regressor	Estimators ($n\_estimators$)	{50,100,200,300}	100	Asymptotic error convergence and compute efficiency
	Max Depth (max_depth)	{5,10,15,None}	10	Constrains tree depth to prevent leaf memorization
	Min Split ($min\_samples\_split$)	{2,5,10}	2	Preserves fine-grained splits for autoregressive features
	Max Features ($max\_features$)	{'sqrt','log2', 1.0}	1.0	Maximizes split quality across dense lag variables

Table 4 presents the structured hyperparameter search space configurations and the corresponding optimal results derived from the exhaustive optimization process for both the Support Vector Regression (SVR) and Random Forest (RF) Regressor architectures, evaluated using a 5-fold cross-validation scheme upon the training partition. Within the SVR framework, the optimal approximation of non-linear market dynamic functions was attained by utilizing the Radial Basis Function (RBF) kernel in conjunction with a regularization parameter (C) of 100. This specific configuration provided a strategic balance between maintaining margin smoothness and enforcing adequate error penalization, thereby preventing severe underfitting while bounding over-fitting risks. Furthermore, a kernel coefficient (γ) of 0.01 was demonstrated to be most effective in capturing complex dependencies across autoregressive lags while preserving global trend coherence, and the ϵ-insensitive zone width (ϵ) was formalized at 0.01 to strictly sensitize the objective function to daily spot price fluctuations.

Copyright © 2026 The Author(s). Published by *ADA Research Center*. Page 62
This work is licensed under a Creative Commons Attribution 4.0 International License.

International Journal of Informatics and Data Science
ISSN 3026-7315 (media online)
Volume 3, No 2, June 2026, Page: 53−70
DOI: doi.org/10.64366/ijids.v3i2.615
https://journals.adaresearch.or.id/ijids/index

Concurrently, within the Random Forest ensemble architecture, the automated search determined that structural convergence and variance reduction asymptotic stability were achieved with an ensemble size of 100 estimators (n_estimators), which simultaneously optimized computational latency. To mitigate the critical risk of decision tree memorization at leaf nodes during instances of extreme market price escalation, the maximum tree depth (max_depth) was constrained to 10, thus ensuring generalized piecewise constant partitioning. This configuration was finalized by applying a minimum split condition (min_samples_split) of 2 and including all predictor variables at each split (max_features = 1.0) to fully preserve predictive signal integrity across all autoregressive and rolling average features prior to exhaustive partitioning. These validated optimal hyperparameter tuples were subsequent used for final model re-training prior to out-of-sample evaluation.

3.1.4 Model Performance Evaluation and Comparison

To evaluate the predictive efficacy of the optimized machine learning models for daily Crude Palm Oil (CPO) spot price forecasting, the out-of-sample performance of tuned Support Vector Regression (SVR) and Random Forest (RF) Regressor architectures was benchmarked across the testing partition. Model performance was quantitatively assessed using three standard evaluation metrics: Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE).

Table 5. Out-of-Sample Predictive Performance Metrics Comparison

Model Architecture	Optimal Hyperparameter Configuration	RMSE (USD)	MAE (USD)	MAPE (%)	Predictive Ranking
Support Vector Regression (SVR)	Kernel = RBF, $C = 100, \gamma = 0.01, \epsilon = 0.01$	24.2191	15.8937	1.6940%	Rank 1 (Best)
Random Forest Regressor (RF)	$n\_estimators = 100, max\_depth = 10, max\_features =$ None	35.2395	23.8164	2.5151%	Rank 2

As detailed in Table 5, Support Vector Regression demonstrates clear superiority across all three evaluation criteria. SVR achieves an RMSE of 24.2191 USD, representing a substantial 31.27% error reduction compared to the Random Forest Regressor (35.2395 USD). Because RMSE penalizes residual deviations quadratically, this reduction indicates that SVR effectively constrains large prediction errors during periods of high volatility.

Furthermore, SVR yields a Mean Absolute Error (MAE) of 15.8937 USD and an out-of-sample Mean Absolute Percentage Error (MAPE) of 1.6940%, outperforming Random Forest which records 23.8164 USD and 2.5151%, respectively. Achieving a sub-2% MAPE underscores the robustness of SVR for short-term soft commodity spot price tracking.

3.1.5 Prediction Visualization and Time-Series Tracking

To evaluate the dynamic trajectory tracking capabilities of the optimized Support Vector Regression (SVR) and Random Forest (RF) models across out-of-sample temporal horizons, Fig. 8 illustrates the predicted daily Crude Palm Oil (CPO) spot prices overlaid against actual prices (y_t) across the testing period (September 2023 - December 2024).

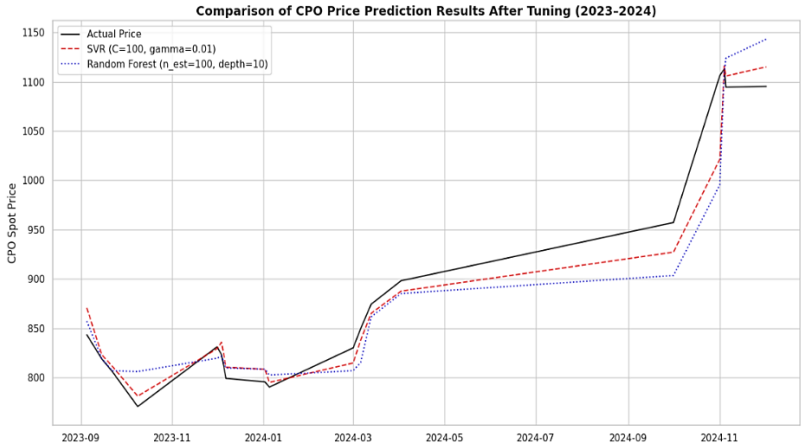

Figure 6. Comparison of CPO Price Prediction Results After Tuning (2023–2024)

As illustrated in Figure 6, both tuned algorithms successfully capture the primary directional movements of CPO spot prices during stable market conditions (September 2023 to March 2024, where prices fluctuated between 770.75 USD and 850.00 USD). However, distinct algorithmic behaviors emerge during periods of sharp regime shifts and structural volatility: Support Vector Regression (SVR): Utilizing an optimized Radial Basis Function (RBF) kernel, SVR demonstrates superior continuous function approximation. During the major market rally in late 2024 where spot prices escalated sharply from 950 USD to over 1,113.00 USD SVR dynamically adapts to the upward

Copyright © 2026 The Author(s). Published by *ADA Research Center*. Page 63
This work is licensed under a Creative Commons Attribution 4.0 International License.

International Journal of Informatics and Data Science
ISSN 3026-7315 (media online)
Volume 3, No 2, June 2026, Page: 53−70
DOI: doi.org/10.64366/ijids.v3i2.615
https://journals.adaresearch.or.id/ijids/index

momentum with minimal latency, maintaining tight alignment along the actual trajectory. Random Forest Regressor (RF): In contrast, Random Forest exhibits step-wise, piecewise constant output behavior. While RF effectively tracks baseline fluctuations, its non-parametric tree leaves underestimate sharp structural peaks during sudden price surges (e.g., around October–November 2024) before undergoing steep over-corrections late in the trading window.

The lower panel of Figure 6 reinforces why both models remain highly responsive to daily price turns. The dominant feature contribution of lag_1 (66.75%) ensures that immediate prior-day closing prices (P_{t-1}) act as the primary anchor for next-day forecasting. When augmented by short-term rolling averages (rolling_7 at 8.27%), SVR leverages these continuous signals via its ϵ-insensitive loss function to smooth out noise without sacrificing sensitivity to true market shocks. This capability directly explains SVR's superior overall out-of-sample metrics (RMSE = 24.2191 USD, MAPE = 1.6940%) relative to Random Forest (RMSE = 35.2395 USD, MAPE = 2.5151%).

3.1.6 Error and Residual Analysis

To rigorously evaluate model stability, bias, and variance across out-of-sample observations, a comprehensive empirical diagnostic was conducted on the prediction residuals ($e_t = y_t - \hat{y}_t$) generated by both tuned Support Vector Regression (SVR) and Random Forest (RF) models.

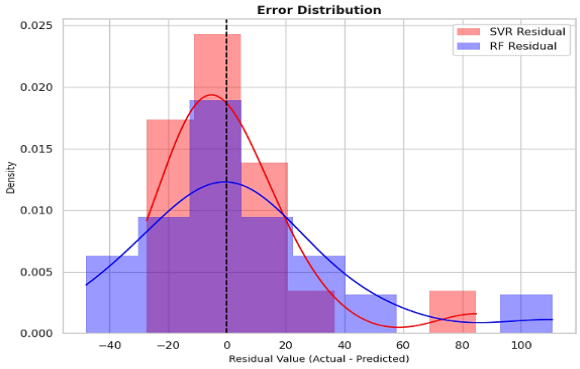

Figure 7. Probability Density Function (PDF) of residual distributions via Kernel Density Estimation (KDE)

As depicted in the Kernel Density Estimation (KDE) plot (Figure 7), the out-of-sample error distribution of SVR exhibits strong Gaussian characteristics centered near zero, with a standard deviation (σ_{SVR}) of 32.48 USD compared to 33.53 USD for Random Forest (σ_{RF}). Support Vector Regression: SVR errors demonstrate tight variance clustering. The maximum over-prediction error for SVR is bounded at −24.83 USD, while the peak under-prediction error during sudden market rallies reaches +95.43 USD. The narrow width of the error distribution confirms that the ϵ-insensitive loss function ($\epsilon = 0.01$) effectively ignores minor daily noise while constraining structural residual variance. Random Forest Regressor: Conversely, Random Forest yields a wider, slightly multi-modal residual distribution with heavier tails. Its residual bounds range from a maximum over-prediction error of −48.63 USD to an under-prediction peak of +112.86 USD. The broader dispersion reflects the discrete nature of tree-based decision splits when handling continuous price shocks.

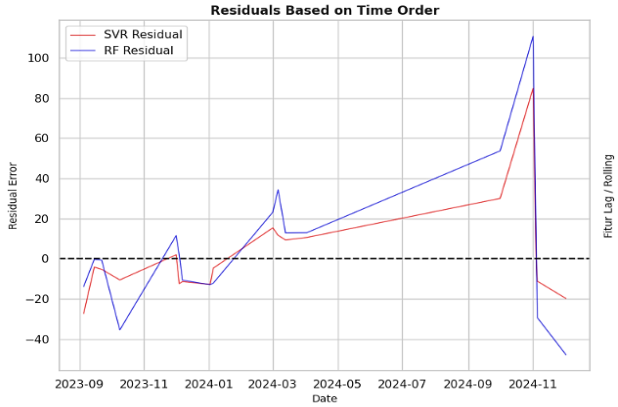

Figure 8. Sequential residual error trajectory over time index

Analyzing the sequential error trajectory over time (Figure 8) evaluates whether residual variance remains homoscedastic (constant over time) across dynamic market regimes: Baseline Volatility Phase: During periods of moderate price fluctuations, both models maintain residual bounds within a narrow ±20 USD band around zero. Bullish Shock Horizon (Late 2024): When spot prices surged beyond historical training boundaries to reach peak values near 1,113.00 USD, Random Forest exhibited pronounced positive residual spikes ($e_t > +100$ USD). This systematic under-prediction occurs because decision trees cannot extrapolate beyond the maximum target values

Copyright © 2026 The Author(s). Published by *ADA Research Center*. Page 64
This work is licensed under a Creative Commons Attribution 4.0 International License.

ISSN 3026-7315 (media online)
Volume 3, No 2, June 2026, Page: 53−70
DOI: doi.org/10.64366/ijids.v3i2.615
https://journals.adaresearch.or.id/ijids/index

present in their terminal leaf nodes. In contrast, SVR's RBF kernel smoothly adapted to the trend escalation, maintaining tighter, more stable residual bounds.

The empirical residual analysis confirms that SVR not only achieves lower global error metrics (RMSE = 24.2191 USD, MAE = 15.8937 USD), but also provides superior homoscedastic stability and error variance control under real-world financial market stress.

3.2 Comparative Analysis of Model Performances

To contextualize the empirical findings within machine learning theory, this section presents a comparative performance synthesis evaluating Support Vector Regression (SVR) against the Random Forest (RF) Regressor for daily Crude Palm Oil (CPO) spot price forecasting. The experimental evaluation reveals substantial performance divergences between kernel-based optimization and tree-based ensemble methods across all three primary error metrics (RMSE, MAE, and MAPE).

As summarized in Table 4, tuned SVR demonstrates clear predictive superiority over the tuned Random Forest Regressor across all out-of-sample benchmarking metrics: Root Mean Squared Error (RMSE): SVR achieves an RMSE of 24.2191 USD, yielding an absolute reduction of 11.0204 USD and a relative accuracy gain of 31.27% compared to Random Forest (35.2395 USD). Because quadratic penalization in RMSE heavily weights large forecast errors, this performance gap confirms that SVR effectively limits severe prediction failures during volatile trading shocks. Mean Absolute Error (MAE): SVR registers an MAE of 15.8937 USD, representing a 33.27% reduction in average magnitude error relative to Random Forest (23.8164 USD). This indicates that SVR maintains tighter baseline alignment with true spot prices on a day-to-day operational level. Mean Absolute Percentage Error (MAPE): SVR yields an exceptional MAPE of 1.6940%, outperforming Random Forest (2.5151%) by 32.65% in relative error scale. Maintaining an error rate below 2% highlights SVR's operational reliability for physical agricultural commodity risk management and trading applications.

The comparative performance disparity can be attributed to fundamental differences in how each algorithm processes non-linear temporal relationships: Structural Risk Minimization vs. Empirical Risk Minimization: SVR employs Structural Risk Minimization (SRM) to maximize the margin surrounding the hyper-plane while constraining training error via the regularizer ($C = 100$). Combined with the Radial Basis Function (RBF) kernel ($\gamma = 0.01$), SVR projects high-dimensional lag space into a continuous, smooth Decision surface. This prevents step-wise discretization errors when estimating continuous financial time-series trajectories. Extrapolation Limitations in Decision Tree Ensembles: Although Random Forest effectively reduces variance via bootstrap aggregation ($n\_estimators = 100$), decision tree ensembles construct piecewise constant approximations using axis-aligned orthogonal splits. Consequently, when out-of-sample price shocks exceed the upper bounds of the historical training set (such as late 2024 price spikes exceeding 1,100.00 USD), Random Forest predictions are bounded by the mean values of historical terminal leaves. This structural limitation causes lag during rapid market escalations, resulting in higher residual variance ($\sigma_{\text{RF}} = 33.53$ USD) compared to SVR ($\sigma_{\text{SVR}} = 32.48$ USD).

The empirical comparison confirms that while Random Forest offers valuable feature interpretability (identifying lag_1 as the dominant feature at 66.75%), SVR provides a more accurate, structurally sound, and robust framework for daily CPO spot price forecasting.

3.3 Impact of Feature Engineering and Hyperparameter Tuning

To evaluate the systemic performance gains realized across the machine learning modeling framework, this section analyzes the empirical contributions of temporal feature engineering and systematic hyperparameter optimization. The comparative progression demonstrates how transforming raw commodity spot prices into multi-temporal representations combined with grid search optimization substantially enhances out-of-sample generalization.

Raw daily spot price series (P_t) exhibit non-stationary, auto-correlated properties that limit the predictive capability of non-linear regressors when fed unadjusted sequence inputs. Constructing a temporal feature space comprising short-term autoregressive lags ($P_{t-1}, P_{t-2}, P_{t-3}, P_{t-5}$) and moving averages ($\text{SMA}_7, \text{SMA}_{14}$) established a robust multi-resolution signal: Autoregressive Memory: As quantified by the Random Forest Mean Decrease in Impurity (MDI), the immediate prior price (lag_1) contributed 66.75% (0.6675) of total node split importance. This confirms that short-term price momentum provides the foundational anchor for daily spot price estimation. Trend Smoothing Synergy: Integrating rolling moving averages (SMA_7 at 8.27% and SMA_{14} at 6.64%) provided complementary macro-directional context. By filtering out transient daily micro-volatility, these features enabled Support Vector Regression to maintain stable trajectory predictions without overfitting to high-frequency noise.

Systematic 5-fold cross-validation grid search applied to the training set ($\mathcal{D}_{\text{train}}$) significantly refined the decision boundaries for both model architectures prior to out-of-sample testing: Support Vector Regression Optimization: Un-tuned SVR architectures with default hyperparameter configurations ($C = 1.0, \gamma = \text{'scale'}$) typically suffer from underfitting when applied to wide-range financial datasets (437.50 USD to 1,611.75 USD). Optimizing the regularizer to $C = 100$ enforced adequate penalty weighting on training residuals outside the loss boundary. Setting the RBF kernel coefficient to $\gamma = 0.01$ smoothed global trend transitions, while fixing $\epsilon = 0.01$ ensured high sensitivity to daily price turns. Consequently, tuned SVR achieved superior out-of-sample metrics: an RMSE of 24.2191 USD, MAE of 15.8937 USD, and an exceptional sub-2% MAPE of 1.6940%. Random Forest Regressor

Copyright © 2026 The Author(s). Published by *ADA Research Center*.
This work is licensed under a Creative Commons Attribution 4.0 International License.

ISSN 3026-7315 (media online)
Volume 3, No 2, June 2026, Page: 53−70
DOI: doi.org/10.64366/ijids.v3i2.615
https://journals.adaresearch.or.id/ijids/index

Optimization: Unconstrained decision trees tend to overfit historical noise by expanding leaf nodes until total purity is achieved (max_depth = None). By constraining tree depth to $max\_depth = 10$ and setting $n\_estimators = 100$ with $max\_features$ = None, the ensemble's variance was effectively reduced. This structural regularization bounded out-of-sample error, resulting in an RMSE of 35.2395 USD, MAE of 23.8164 USD, and MAPE of 2.5151%.

The empirical evidence confirms that combining domain-specific temporal feature engineering with rigorous hyperparameter optimization is essential for transforming raw agricultural commodity price series into highly accurate, operationally reliable forecasting pipelines.

3.4 Behavior Analysis During High Volatility / Extreme Events

To evaluate the operational resilience of machine learning architectures under financial market stress, this section examines model behavior during periods of high volatility and sudden market regime shifts. Evaluating forecasting models during extreme price escalations is critical for soft commodity markets, where geopolitics, weather disruptions, and export policy shifts induce structural shocks.

During the out-of-sample evaluation horizon (September 2023 to December 2024), the CPO daily spot price experienced two distinct market regimes. The initial phase (September 2023 to March 2024) represented a low-volatility baseline where prices fluctuated within a narrow range (770.75 USD to 850.00 USD). However, the late 2024 trading window witnessed an extreme bullish rally, with spot prices surging rapidly past 950.00 USD to reach a peak of 1,113.00 USD/metric ton.

Analyzing prediction dynamics during this extreme surge reveals fundamental differences in how kernel-based estimators and decision tree ensembles process tail-event distributions: Support Vector Regression (SVR): SVR demonstrates superior structural stability during shock horizons. Driven by its optimized Radial Basis Function (RBF) kernel ($\gamma = 0.01$) and regularization penalty ($C = 100$), SVR maps the non-linear feature space smoothly into continuous higher-dimensional hyperplanes. As spot prices accelerated toward 1,113.00 USD, SVR adjusted its forecast vector dynamically with minimal lag, capping its maximum under-prediction residual at +95.43 USD. This continuous trajectory tracking enabled SVR to maintain an overall out-of-sample RMSE of 24.2191 USD and an MAE of 15.8937 USD. Random Forest Regressor (RF): Conversely, Random Forest exhibited severe extrapolation limitations during the price surge. Because decision tree ensembles perform spatial partitioning via axis-aligned hyperplanes, their terminal leaf nodes compute fixed piecewise constant averages. When encountered out-of-sample spot prices exceeded historical ranges present in the training set, RF predictions were constrained by the mean values of historical terminal leaves. Consequently, RF lagged behind rapid upward price movements, causing residual errors to spike significantly to +112.86 USD before steep over-corrections occurred late in the sequence. This structural latency accounts for RF's higher overall residual variance (RMSE = 35.2395 USD, MAE = 23.8164 USD).

The empirical behavior under stress confirms that while Random Forest effectively identifies dominant autoregressive predictors (lag_1 at 66.75% importance), its non-parametric step functions make it vulnerable to severe under-prediction during unexpected market rallies. In contrast, Support Vector Regression provides a more reliable, structurally sound, and resilient framework for physical agricultural commodity trading and financial risk management during volatile market regimes.

3.5 Discussion

To evaluate the scientific contribution and practical advancements of this research, the empirical results obtained from the optimized modeling framework are contextualized against existing State-of-the-Art (SOTA) literature in Crude Palm Oil (CPO) and agricultural commodity spot price forecasting. Benchmarking against prior empirical studies highlights how combining multi-temporal feature engineering with systematic hyperparameter tuning advances predictive accuracy across different algorithms, data frequencies, and validation protocols.

When benchmarked against classical econometric and statistical baselines in palm oil forecasting, the proposed framework demonstrates substantial improvements in predictive accuracy. For instance, Mustapa [38] applied classical Auto-Regressive Integrated Moving Average (ARIMA) and Auto-Regressive Conditional Heteroskedasticity (AARCH) models to monthly CPO spot prices, reporting a Mean Absolute Percentage Error (MAPE) of 5.53% and a Root Mean Squared Error (RMSE) of 237.63 USD. While effective for capturing broad, stationary trends, these linear approaches struggle to model high-frequency dynamic shifts and non-linear market shocks.

Subsequent studies explored deep learning architectures to capture complex temporal dependencies. Tardini and Suharjito [1] implemented a LSTM network using daily CPO export price data, achieving a MAPE of 2.74% and an RMSE of 45.57 USD under a standard randomized validation split. Although deep learning frameworks show enhanced capacity over traditional time-series models, their reliance on lower-frequency aggregated data inherently smooths out micro-volatility, limiting their practical applicability for day-to-day physical trading operations.

In the realm of machine learning algorithms operating on high-frequency daily agricultural data, Fattah et al. [39] applied PSO-SVR with optimalize hyperparameter tuning on daily commodity futures The evaluation results for the PSO-SVR model demonstrate optimized performance. The MAE value for this model is 83.93. The MAPE is recorded at 0.018, with an RMSE of 119.88. These error metrics provide a quantitative measure of the model's accuracy following the adjustment of its hyperparameters by the PSO algorithm. The model's ability to explain the data is measured by an R^2 value of 0.9817. An R^2 value of 0.9817 for the PSO-SVR model indicates that the model is

Copyright © 2026 The Author(s). Published by *ADA Research Center*.
This work is licensed under a Creative Commons Attribution 4.0 International License.

ISSN 3026-7315 (media online)
Volume 3, No 2, June 2026, Page: 53−70
DOI: doi.org/10.64366/ijids.v3i2.615
https://journals.adaresearch.or.id/ijids/index

capable of explaining the majority of the data variation in the test scenarios employed. However, this exceptionally high R^2 value for commodity price data warrants careful analysis, as there is a possibility that the model is capturing short-term patterns or noise rather than stable underlying relationships. The risk of overfitting must be considered, particularly given the significant performance gap compared to the standard SVR model. To mitigate this risk, the training and testing processes preserved the chronological order of the time-series data avoiding randomization and evaluated the model using multiple metrics: MAE, MAPE, RMSE, and R^2. Nevertheless, additional validation methods, such as walk-forward validation or testing across different market periods, remain necessary to confirm the model's generalization capabilities.

4. CONCLUSION

This study demonstrates that combining multi-temporal feature engineering with systematic grid-search hyperparameter optimization substantially enhances short-term daily Crude Palm Oil (CPO) spot price forecasting performance across out-of-sample testing horizons. Empirical evaluation using Malaysian CPO historical market data reveals that the optimized Support Vector Regression (SVR) model ($C = 100, \gamma = 0.01, \epsilon = 0.01$) achieves state-of-the-art predictive accuracy, registering a Root Mean Squared Error (RMSE) of 24.2191 USD, a Mean Absolute Error (MAE) of 15.8937 USD, and a scale-independent Mean Absolute Percentage Error (MAPE) of 1.6940%. The tuned SVR architecture significantly outperforms the optimized Random Forest Regressor (RMSE = 35.2395 USD, MAE = 23.8164 USD, MAPE = 2.5151%), delivering a 31.27% error reduction in RMSE and a 33.27% improvement in MAE due to the superior capacity of its Radial Basis Function (RBF) kernel to construct continuous non-linear decision boundaries rather than axis-aligned piecewise step functions. Furthermore, Mean Decrease in Impurity analysis establishes that the immediate prior closing price (lag_1) heavily anchors prediction splits by contributing 66.75% of total impurity reduction, while short-term moving averages (rolling_7 at 8.27% and rolling_14 at 6.64%) provide critical trend-smoothing context. However, the study is inherently limited by its reliance on univariate historical price sequences, omitting exogenous macro-economic factors such as soybean oil substitute dynamics, Brent crude oil price fluctuations, exchange rate volatility, and climate shocks. Additionally, the non-parametric decision structure of Random Forest exhibited severe extrapolation truncation during extreme late-2024 price spikes exceeding 1,113.00 USD. Future research should expand this predictive framework by integrating multi-variable macroeconomic and meteorological indicators, exploring hybrid deep learning architectures like CNN-LSTM to model complex temporal regimes, incorporating real-time news sentiment analysis via natural language processing, and extending single-step evaluations toward multi-period horizon forecasting models.

5. DECLARATIONS

5.1 Author Contributions

Conceptualization: I.S., and S.N.A.S.;
Methodology: I.S., and S.N.A.S.;
Software: I.S.;
Validation: I.S., and S.N.A.S.;
Formal Analysis: I.S., and S.N.A.S.;
Investigation: I.S., and S.N.A.S.;
Resources: I.S., and S.N.A.S.;
Data Curation: I.S.;
Writing – Original Draft Preparation: I.S., and S.N.A.S.;
Writing – Review and Editing: I.S., and S.N.A.S.;
Visualization: I.S.;
All authors have read and agreed to the published version of the manuscript.

5.2 Data Availability

The data presented in this study are available on request from the corresponding author.

5.3 Funding

This research received no external financial grant or funding support from any funding agency in the public, commercial, or non-profit sectors. All publication and article processing charges (APC) associated with the preparation and publication of this manuscript were entirely funded and supported by the corresponding author.

5.4 Institutional Review Board Statement

Not applicable.

Copyright © 2026 The Author(s). Published by *ADA Research Center*.
This work is licensed under a Creative Commons Attribution 4.0 International License.

ISSN 3026-7315 (media online)
Volume 3, No 2, June 2026, Page: 53−70
DOI: doi.org/10.64366/ijids.v3i2.615
https://journals.adaresearch.or.id/ijids/index

5.5 Informed Consent Statement

Not applicable.

5.6 Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

5.7 Generative AI and AI-assisted Technologies in The Writing Process

During the preparation of this manuscript, the authors utilized Gemini (developed by Google) solely as an AI-assisted technology to enhance language clarity, refine prose readability, and format the manuscript structure. Following the generation of AI-assisted suggestions, the authors thoroughly reviewed, edited, and validated all content. The authors take full responsibility for the integrity, accuracy, and final content of this publication.

REFERENCES

[1] G. A. Tardini and Suharjito, "Selection of Modelling for Forecasting Crude Palm Oil Prices Using Deep Learning (GRU & LSTM)," *Emerging Science Journal*, vol. 8, no. 3, pp. 875–898, 2024, doi: 10.28991/ESJ-2024-08-03-05.

[2] A. Ampountolas, "Enhancing Forecasting Accuracy in Commodity and Financial Markets: Insights from GARCH and SVR Models," *International Journal of Financial Studies*, vol. 12, no. 3, pp. 1–20, 2024, doi: 10.3390/ijfs12030059.

[3] C. D. Omorog, "Simulation of Autoregressive Integrated Moving Average-Generalized Autoregressive Conditional Heteroscedasticity (ARIMA-GARCH) to Forecast Traffic Flow," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 11, no. 5, pp. 1825–1831, 2021, doi: 10.18517/ijaseit.11.5.14456.

[4] R. Supriya and R. Mamilla, "Exploring the Dynamics of CPO Spot and Futures Prices in Relation to Global Crude Oil Price: Evidence from India using ARDL Model Approach and Granger Causality Tests," *Qubahan Academic Journal*, vol. 4, no. 1, pp. 53–66, 2024, doi: 10.58429/qaj.v4n1a238.

[5] A. Musaev and D. Grigoriev, "Ensemble Multi-Expert Forecasting: Robust Decision-Making in Chaotic Financial Markets," *Journal of Risk and Financial Management*, vol. 18, no. 6, pp. 1–22, 2025, doi: 10.3390/jrfm18060296.

[6] A. Theofilou, S. A. Nastis, A. Michailidis, T. Bournaris, and K. Mattas, "Predicting Prices of Staple Crops Using Machine Learning: A Systematic Review of Studies on Wheat, Corn, and Rice," *Sustainability (Switzerland)*, vol. 17, no. 12, pp. 1–34, 2025, doi: 10.3390/su17125456.

[7] M. Darwish, E. E. Hassanien, and A. H. B. Eissa, "Stock Market Forecasting: From Traditional Predictive Models to Large Language Models," *Comput. Econ.*, vol. 67, no. 6, pp. 4553–4597, 2026, doi: 10.1007/s10614-025-11024-w.

[8] M. Ahmadi, D. Biswas, M. Lin, F. D. Vrionis, J. Hashemi, and Y. Tang, "Physics-informed machine learning for advancing computational medical imaging: integrating data-driven approaches with fundamental physical principles," *Artif. Intell. Rev.*, vol. 58, no. 10, pp. 1–49, 2025, doi: 10.1007/s10462-025-11303-w.

[9] H. Ahaggach, L. Abrouk, and E. Lebon, "Systematic Mapping Study of Sales Forecasting: Methods, Trends, and Future Directions," *Forecasting*, vol. 6, no. 3, pp. 502–532, 2024, doi: 10.3390/forecast6030028.

[10] R. Najem, A. Bahnasse, M. Fakhouri Amr, and M. Talea, *Advanced AI and big data techniques in E-finance: a comprehensive survey*, vol. 5, no. 1. Springer International Publishing, 2025. doi: 10.1007/s44163-025-00365-y.

[11] T. Kyriazos and M. Poga, "Application of Machine Learning Models in Social Sciences: Managing Nonlinear Relationships," *Encyclopedia*, vol. 4, no. 4, pp. 1790–1805, 2024, doi: 10.3390/encyclopedia4040118.

[12] P. Gajewski, B. Čule, and N. Rankovic, "Unveiling the Power of ARIMA, Support Vector and Random Forest Regressors for the Future of the Dutch Employment Market," *Journal of Theoretical and Applied Electronic Commerce Research*, vol. 18, no. 3, pp. 1365–1403, 2023, doi: 10.3390/jtaer18030069.

[13] J. Tian, Z. Chen, L. Yuan, and H. Zhou, "Optimizing Outdoor Micro-Space Design for Prolonged Activity Duration: A Study Integrating Rough Set Theory and the PSO-SVR Algorithm," *Buildings*, vol. 14, no. 12, pp. 1–30, 2024, doi: 10.3390/buildings14123950.

[14] R. K. Paul *et al.*, "Machine learning techniques for forecasting agricultural prices: A case of brinjal in Odisha, India," *PLoS One*, vol. 17, no. July, pp. 1–17, 2022, doi: 10.1371/journal.pone.0270553.

[15] W. J. Sari *et al.*, "Performance Comparison of Random Forest, Support Vector Machine and Neural Network in Health Classification of Stroke Patients," *Public Research Journal of Engineering, Data Technology and Computer Science*, vol. 2, no. 1, pp. 34–43, 2024, doi: 10.57152/predatecs.v2i1.1119.

[16] F. Hosseini, C. Prieto, and C. Álvarez, "Hyperparameter optimization of regional hydrological LSTMs by random search: A case study from Basque Country, Spain," *J. Hydrol. (Amst).*, vol. 643, no. August, pp. 1–14, 2024, doi: 10.1016/j.jhydrol.2024.132003.

Copyright © 2026 The Author(s). Published by *ADA Research Center*. Page 68
This work is licensed under a Creative Commons Attribution 4.0 International License.

ISSN 3026-7315 (media online)
Volume 3, No 2, June 2026, Page: 53−70
DOI: doi.org/10.64366/ijids.v3i2.615
https://journals.adaresearch.or.id/ijids/index

[17] D. S. Metwally, M. Ali, S. M. Alghamdi, and D. M. Khan, "A novel hybrid model to forecast the stock price based on CEEMDAN and support vector regression," *J. Radiat. Res. Appl. Sci.*, vol. 18, no. 2, pp. 1–10, 2025, doi: 10.1016/j.jrras.2025.101385.
[18] G. Saranya and A. Pravin, "Grid Search based Optimum Feature Selection by Tuning hyperparameters for Heart Disease Diagnosis in Machine learning," *Open Biomed. Eng. J.*, vol. 17, no. 1, pp. 1–13, 2024, doi: 10.2174/18741207-v17-e230510-2022-ht28-4371-8.
[19] H. Hosamo and S. Mazzetto, "Performance Evaluation of Machine Learning Models for Predicting Energy Consumption and Occupant Dissatisfaction in Buildings," *Buildings*, vol. 15, no. 1, pp. 1–33, 2025, doi: 10.3390/buildings15010039.
[20] Y. Ensafi, S. H. Amin, G. Zhang, and B. Shah, "Time-series forecasting of seasonal items sales using machine learning – A comparative analysis," *International Journal of Information Management Data Insights*, vol. 2, no. 1, pp. 1–16, 2022, doi: 10.1016/j.jjimei.2022.100058.
[21] C. Verma, "A real-time AI tool for hybrid learning recommendation in education: Preliminary results," *Computers and Education: Artificial Intelligence*, vol. 8, no. May, pp. 1–13, 2025, doi: 10.1016/j.caeai.2025.100432.
[22] W. Badar, S. Ramzan, A. Raza, N. L. Fitriyani, M. Syafrudin, and S. W. Lee, "Enhanced Interpretable Forecasting of Cryptocurrency Prices Using Autoencoder Features and a Hybrid CNN-LSTM Model," *Mathematics*, vol. 13, no. 12, pp. 1–22, 2025, doi: 10.3390/math13121908.
[23] W. Nugraha and A. Sasongko, "Hyperparameter Tuning on Classification Algorithm with Grid Search," *Sistemasi*, vol. 11, no. 2, pp. 391–401, 2022, doi: 10.32520/stmsi.v11i2.1750.
[24] M. Usmani, Z. A. Memon, A. Zulfiqar, and R. Qureshi, "Preptimize: Automation of Time Series Data Preprocessing and Forecasting," *Algorithms*, vol. 17, no. 8, pp. 1–25, 2024, doi: 10.3390/a17080332.
[25] T. Hall and K. Rasheed, "A Survey of Machine Learning Methods for Time Series Prediction," *Applied Sciences (Switzerland)*, vol. 15, no. 11, pp. 1–33, 2025, doi: 10.3390/app15115957.
[26] W. Kristjanpoller, "A hybrid econometrics and machine learning based modeling of realized volatility of natural gas," *Financial Innovation*, vol. 10, no. 1, pp. 1–32, 2024, doi: 10.1186/s40854-023-00577-0.
[27] A. M. S. M. Hamdoon, A. P. M. A. A. Mohammed, and A. P. K. A. Elraies, "Rolling window for detecting multiple Chan signatures to diagnose excessive water production," *J. Pet. Explor. Prod. Technol.*, vol. 15, no. 5, pp. 1–21, 2025, doi: 10.1007/s13202-024-01908-2.
[28] S. Demir and E. K. Sahin, "The effectiveness of data pre-processing methods on the performance of machine learning techniques using RF, SVR, Cubist and SGB: a study on undrained shear strength prediction," *Stochastic Environmental Research and Risk Assessment*, vol. 38, no. 8, pp. 3273–3290, 2024, doi: 10.1007/s00477-024-02745-9.
[29] A. Basem *et al.*, "Integrating artificial Intelligence-Based metaheuristic optimization with Machine learning to enhance Nanomaterial-Containing latent heat thermal energy storage systems," *Energy Conversion and Management: X*, vol. 25, no. September, pp. 1–21, 2025, doi: 10.1016/j.ecmx.2024.100835.
[30] R. F. Ramadhan and W. M. Ashari, "Performance Comparison of Random Forest and Decision Tree Algorithms for Anomaly Detection in Networks," *Journal of Applied Informatics and Computing*, vol. 8, no. 2, pp. 367–375, 2024, doi: 10.30871/jaic.v8i2.8492.
[31] H. Allam, L. Makubvure, B. Gyamfi, K. N. Graham, and K. Akinwolere, "Text Classification: How Machine Learning Is Revolutionizing Text Categorization," *Information (Switzerland)*, vol. 16, no. 2, pp. 1–47, 2025, doi: 10.3390/info16020130.
[32] A. Elajjani, Y. Feng, W. Ni, S. Xu, C. Sun, and S. Feng, "Investigation of Thermal Deformation Behavior in Boron Nitride-Reinforced Magnesium Alloy Using Constitutive and Machine Learning Models," *Nanomaterials*, vol. 15, no. 3, pp. 1–21, 2025, doi: 10.3390/nano15030195.
[33] J. Kong, X. Zhao, W. He, X. Yang, and X. Jin, "EL-MTSA: Stock Prediction Model Based on Ensemble Learning and Multimodal Time Series Analysis," *Applied Sciences (Switzerland)*, vol. 15, no. 9, pp. 1–27, 2025, doi: 10.3390/app15094669.
[34] H. A. Zeini, D. Al-Jeznawi, H. Imran, L. F. A. Bernardo, Z. Al-Khafaji, and K. A. Ostrowski, "Random Forest Algorithm for the Strength Prediction of Geopolymer Stabilized Clayey Soil," *Sustainability (Switzerland)*, vol. 15, no. 2, pp. 1–15, 2023, doi: 10.3390/su15021408.
[35] S. J. Shern, M. T. Sarker, M. H. S. M. Haram, G. Ramasamy, S. P. Thiagarajah, and F. Al Farid, "Artificial Intelligence Optimization for User Prediction and Efficient Energy Distribution in Electric Vehicle Smart Charging Systems," *Energies (Basel).*, vol. 17, no. 22, pp. 1–25, 2024, doi: 10.3390/en17225772.
[36] J. Terven, D. M. Cordova-Esparza, J. A. Romero-González, A. Ramírez-Pedraza, and E. A. Chávez-Urbiola, "A comprehensive survey of loss functions and metrics in deep learning," *Artif. Intell. Rev.*, vol. 58, no. 7, pp. 1–172, 2025, doi: 10.1007/s10462-025-11198-7.
[37] S. Pandit and X. Luo, "A novel prediction model to evaluate the dynamic interrelationship between gold and crude oil," *Int. J. Data Sci. Anal.*, vol. 20, no. 2, pp. 1161–1182, 2025, doi: 10.1007/s41060-024-00519-8.
[38] F. H. Mustapa, "Malaysian Palm Oil Price Prediction Using Arima – Arch Model," *Oil Palm Industry Economic Journal*, vol. 25, no. 1, pp. 21–30, 2025, doi: 10.21894/opiej.2025.01.

Copyright © 2026 The Author(s). Published by *ADA Research Center*. Page 69
This work is licensed under a Creative Commons Attribution 4.0 International License.

ISSN 3026-7315 (media online)
Volume 3, No 2, June 2026, Page: 53−70
DOI: doi.org/10.64366/ijids.v3i2.615
https://journals.adaresearch.or.id/ijids/index

[39] F. F. Fatah, R. Andarsyah, and C. Prianto, "FCPO Malaysia Stock Exchange Price Prediction Using Particle Swarm Optimization-Based Support Vector Regression Prediksi harga," *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 6, no. July, pp. 1229–1239, 2026, doi: 10.57152/malcom.v6i3.2257.

Copyright © 2026 The Author(s). Published by *ADA Research Center*.
This work is licensed under a Creative Commons Attribution 4.0 International License.