**Asian Academy *of* Management Journal *of* Accounting *and* Finance**

*AAMJAF Vol. 20, No. 1, 305–333, 2024*

# CRUDE PALM OIL PRICE PREDICTION USING SIMULATED ANNEALING-BASED SUPPORT VECTOR REGRESSION (SA-SVR)

**Chai Wen Goh, Jack Chai, Amirah Rahman* and Wen Eng Ong**

*School of Mathematical Sciences, Universiti Sains Malaysia, 11800 USM Pulau Pinang, Malaysia*

*Corresponding author: amirahr@usm.my

## ABSTRACT

*Palm oil is one of the major export products of Malaysia. Predicting the price of crude palm oil futures (FCPO) traded on BURSA Malaysia Derivative is essential as agricultural markets have an inherent tendency towards instability, and thus are more vulnerable to price shocks than other industrial sectors. Hence, if the price of the futures contract on crude palm oil can be forecasted accurately, many parties such as farmers, refiners and distributors can manage the risk of price fluctuations through FCPO. This study proposes the metaheuristic and machine learning hybridised model of simulated annealing-based support vector regression (SA-SVR). The SVR in this model produces close price predictions of the FCPO with minimum deviation from the actual value with the help of SA, which first determines the best hyperparameter set to be utilised in the SVR. Although the proposed Radial Basis Function (RBF) kernelised SA-SVR model inputs only 10% of training data due to memory overload issues, it has produced a satisfying prediction result with an average execution time of 2 minutes and 34 seconds. The model performance was analysed further by using different ratios in data splitting, varying temperature combinations for the SA algorithm and initiating the parameter search based on the previous best hyperparameter set. Results show that keeping the test size constant and extracting more historical data on FCPO price for model training is better than varying train-test split ratios. The temperature schedule strategy showed that different initial and minimum SA temperature combinations affects the overall optimisation results.*

Received: 3 May 2023; Accepted: 20 November 2023; Published: 7 June 2024

To cite this article: Goh, C. W., Chai, J., Rahman, A., & Ong, W. E. (2024). Crude palm oil price prediction using simulated annealing-based support vector regression (SA-SVR). *Asian Academy of Management Journal of Accounting and Finance, 20*(1), 305–333. https://doi.org/10.21315/aamjaf2024.20.1.10

To link to this article: https://doi.org/10.21315/aamjaf2024.20.1.10

*The best combination was the initial temperature of 100 and minimum of 40. In addition, the number of temperature reductions and average execution time to reach the best state decreases when the starting point of the parameter search space is close to the best values.*

**Keywords:** Simulated annealing, Support vector regression, Crude palm oil price prediction, Hyperparameter tuning

## INTRODUCTION

Palm oil is a popular agricultural product in the Malaysian plantation industry. Due to the favourable climate, palm oil can grow easily in Malaysia and has brought high economic value to both farmers and the country. Based on Malaysia Palm Oil Council (2020), palm oil is extracted from the mesocarp of the fruit of an oil palm. Crude palm oil is normally processed by a physical refining process in which the oil is turned into a golden yellow refined oil for further end use applications. Palm oil is mainly used for cooking, whereas crude palm oil is mainly used for manufacturing of pre-packaged food, cosmetics, and cleaning products. According to Malaysian Palm Oil Industry, Malaysia accounted for 25.8% and 34.3% of world's palm oil production and exports in 2020. Malaysia is the second largest palm oil producer in the world, with around 17.3 million tonnes metric in palm oil export (Shahbandeh, 2021). Thus, palm oil is an important resource in Malaysia.

Agricultural markets have an inherent tendency towards instability. This is mainly because the supply and demand market fundamentals of the agricultural sector are characterised by rigidity. Producers of agricultural commodities are therefore much more vulnerable to price shocks than other industrial sectors (Madre & Devuyst, 2016). There is futures contract on crude palm oil offered by BURSA Malaysia. The futures contract is actively used by edible oils and fats industry players as a risk management solution, as well as by fund managers and financial institutions for managing price fluctuations in the market (BURSA Malaysia, 2021). Hence, if the price of the futures contract on crude palm oil can be forecasted accurately, many parties, such as farmers, refiners and distributors, will benefit as they can plan by purchasing the futures contract offered by BURSA Malaysia to manage the risk of price fluctuation.

There has recently been an increase in the usage of machine learning approaches in price prediction due its ability in processing large amounts of information efficiently as reported by Obthong et al. (2020). One popular approach is support vector regression (SVR), which is a machine learning algorithm that aims to increase prediction accuracy by intaking more data and works well in high dimensionality space (Drucker et al., 1996). However, machine learning

approaches involve setting many parameter values, and these values may affect the quality of the price prediction. The usage of metaheuristics in determining hyperparameters for machine learning approaches has been increasing recently, and has been found to enhance the accuracy of the approaches (Oliva et al., 2021). Simulated annealing (SA), a metaheuristic algorithm consisting of exploration and exploitation, provides optimised hyperparameters that can improve the SVR. Hence, this study proposes a hybrid simulated annealing-based support vector regression (SA-SVR) approach for the crude palm oil futures (FCPO) price prediction problem.

## LITERATURE REVIEW

Time-series forecasting models are capable of price prediction based on the historical data of market prices. Tan et al. (2021) reported that the two main approaches for time series forecasting are classical statistical and artificial intelligence models. Classical statistical models such as Autoregressive Integrated Moving Average (ARIMA) (Dong et al., 2017; Alwadi et al., 2018; Hyndman & Athanasopoulos, 2018) and Exponential Smoothing (ES) (Shahid & Rahaman, 2020; Funde & Damani, 2023) methods are frequently used to provide benchmark forecasts, serving as one basis for comparing performance. A recent article by Hussin et al. (2023) utilised Bayesian Network design for CPO price prediction. Selected forecasting models using machine learning approaches such as Artificial Neural Networks (ANN) and SVR can be found in González-Mancha et al. (2018), Amal et al. (2021), and Jaquart et al. (2021). Yee and Samsudin (2021) compared ANN with ARIMA for CPO price prediction and found that ANN provided more accurate predictions. Ofuoku and Ngniatedema (2022) compared three ANN techniques with a SARIMA model for CPO price prediction and determined that the ANN technique of Long Short-Term Memory (LTSM) outperformed the rest.

The high magnitude of fluctuation and chaotic and non-linear behaviours of the CPO price series reflects significant challenges in forecasting approaches. The review by Kasturi et al. (2017) found that ML techniques outperformed the classical statistical methods in recent financial forecasting studies as classical models are inherent to certain assumptions, such as linearity, which might affect the prediction accuracy. For instance, ARIMA is insufficient to capture the non-linearity in CPO prices as it is a linear model. A pioneering work in forecasting crude palm oil price in Malaysia by Arshad and Ghaffar (1986) showed that ARIMA model with Box-Jenkins technique is limited to short-term forecasting. SVR reveals promising capabilities for CPO price forecasting compared to popular neural network models. Xie et al. (2006), Kasturi et al. (2017) and Shabri and

Hamid (2019) found that SVR can achieve excellent generalisation ability since it implements structural risk minimisation (SRM) principle that aims to minimise an upper bound of the generalisation error rather than minimising the training error. SRM based SVR has the strength of producing more general solutions (global optimum), which means that the tendency of getting stuck in local minima is less likely to occur in SVR. Also, it is resistant to the overfitting issue as it uses SRM. Kasturi et al. (2020) presented a comparative study of HWES, ANN and LSTM for CPO price forecasting. The results showed that LSTM demonstrates significant improvement with the least RMSE, indicating it is better in determining the movement or behaviour of CPO price. Mohd Nain et al. (2022) reviewed studies that used an artificial intelligence framework in CPO price prediction stated that SVR was the among the most promising machine learning methods for CPO price prediction.

Apart from comparing the forecasting accuracy of different models, some studies focus on parameter selection. Saadah et al. (2021) applied SVR to predict CPO price in Indonesia. The study compared the three kernel functions (linear, polynomial, and radial basis function [RBF]) and tested different parameter values (regularisation parameter and bias value). The results showed the use of the RBF kernel with the best parameter values provided high accuracy (about 98%) in CPO price forecasting. The study by Aini and Haviluddin (2019) on parameter selection for BPNN also resulted in good prediction accuracy.

Limited studies have applied metaheuristics in the palm oil industry. A genetic algorithm neural network (GANN) was applied to predict the international price of CPO and Soybean Oil (SBO) (Silalahi, 2013). The study concluded that GANN could perform precise prediction to most agricultural commodities. The results also suggested that this method could be an important tool for forecasting as it showed prediction value with the smallest error. Salman et al. (2018) proposed a model of BPNN with PSO optimisation (PSO-BP), which showed better results than the independent BPNN model for accurate prediction and error convergence by providing better RMSE values.

The challenge of achieving reliability and predictive accuracy of an ML model resides in the selection of hyperparameters. Manually setting the values of hyperparameters in a trial-and-error manner is cumbersome and time-consuming. Hence, automation using a metaheuristic algorithm can quickly obtain the appropriate solution set. According to Siddique and Adeli (2016), SA is a popular metaheuristic algorithm with great success in engineering and industrial applications. Seminal work in applying support vector machine (SVM) with SA include Pai and Hong (2005) who forecasted electricity load, and Lin et al. (2008) who achieved higher accuracy in classification problems on UCI datasets. The

advantages of computational efficiency, simplicity, ease of implementation and coding, and the ability to handle quite complex cost functions and constraints make SA an attractive choice for parameter determination in the SVR.

Although the literature regarding adoption of the SA algorithm to train SVR for CPO price forecasting is non-existent (to the knowledge of the authors), the forecasting capability for different types of time series of this hybrid model was discovered. Li et al. (2010) and Zeng et al. (2012) proposed the hybrid SA-SVR model in time series cooling load forecasting and sintered temperature forecasting, respectively. The experimental results of both studies showed that SA-SVR achieved better accuracy and generalisation than SVR-PSO. It revealed the feasibility of applying the proposed model to complex forecasting problems. Even though other algorithms may either be on par or outperform SA, there is always room for further exploration and improvement as no single best algorithm fits all problems. Therefore, this study integrates SA with SVR for CPO price forecasting.

## METHODOLOGY AND DATA

### Non-linear SVR Model Formulation

Support vector regression (SVR) is an extension of the support vector machine (SVM) applied to regression problems. Consider a given training data of $N$ elements, $\{(x_i, y_i), i = 1, 2, \ldots, N\}$, where $x_i$ denotes the $i$-th element in $n$-dimensional input space; that is, $x_i = x_{1i}, \ldots, x_{ni} \in \mathbb{R}^n$ and $y_i \in \mathbb{R}$ is the output value corresponding to $x_i$. The regression problem approximates the value of $y$ for any $x$ by the function of $y = f(x)$ Suppose the decision function of a linear regression problem takes the form of $f(x) = \langle w, x \rangle + b$, where $\langle .,. \rangle$ denotes the inner product, $w$ is the weight vector, and $b$ is the bias term. The objective of SVR is to fit the data by finding $f(x)$ that has the most deviation from the actual target value $y_i$ for every training data and ensure that $w$ is small. In other words, the goal is to minimise the Euclidean norm of $w$, $\|w\|$, subject to the prediction error of each training data should be at most equal to the defined precision $\varepsilon$. The optimisation problem can be formulated as follows:

$$\text{Minimise } \frac{1}{2}\|w\|^2,$$

$$\text{subject to } \begin{cases} y_i - \langle w, x_i \rangle - b \leq \varepsilon, \\ \langle w, x_i \rangle + b - y_i \leq \varepsilon. \end{cases}$$

For cases when the data points lie outside the range, the $\varepsilon$-insensitive loss function (Ojemakinde, 2006) is considered to penalise the errors. The optimisation problem can be written as:

$$\text{Minimise } \frac{1}{2}\| w \|^2 + C\sum_{i=0}^{N} L_\varepsilon \left( y_i, f(x_i) \right),$$

Where $C$ is a predefined regularising parameter or called the "coefficient of penalty", and

$$\mathrm{L}_\varepsilon(y, f(x)) = \begin{cases} 0, \text{ if } \left| y - f(x) \right| \leq \varepsilon, \\ \left| y - f(x) \right| \leq \varepsilon, \text{ if } \left| y - f(x) \right| > \varepsilon. \end{cases}$$

Note that the absolute value in the loss function makes the objective function non-differentiable. Slack variables $\xi^*$ and $\xi$ are incorporated to create room for infeasible constraints. These slack variables measure the distance from actual values to the corresponding boundary values of the $\varepsilon$-tube, as illustrated in Figure 1. Each data point above the tube has $\xi^* > 0$ and $\xi = 0$, while each data point below the tube has $\xi^* = 0$ and $\xi > 0$.

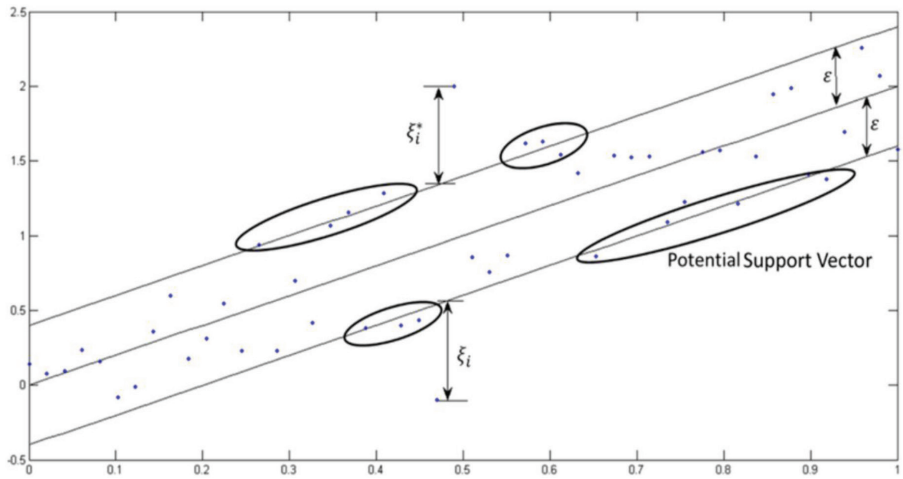



*Figure 1*. One-dimensional linear SVR (Source: Awad & Khanna, 2015)

The optimisation problem is modified accordingly and expressed in this form:

$$\text{Minimise } \frac{1}{2}\| w \|^2 + C\sum_{i=1}^{N} (\xi_i^* + \xi_i)$$

$$\text{subject to} \begin{cases} y_i - \langle w, x_i \rangle - b \leq \varepsilon + \xi_i^*. \\ \langle w, x_i \rangle + b - y_i \leq \varepsilon + \xi_i. \\ \xi_i^*, \xi_i \geq 0, \end{cases}$$

where $\xi^*$ and $\xi$ are the slack variables. The usual solution procedure of the above inequality constrained convex problem is through solving its dual form by applying the Lagrange multipliers method and Karush-Kuhn Tucker (KKT) conditions.

Introducing Lagrange multipliers $\lambda^*$, $\lambda$, $\alpha^*$ and $\alpha$ forms the corresponding Lagrangian as:

$$L = L(w, b, \xi^*, \xi, \lambda^*, \lambda, \alpha^*, \alpha) \frac{1}{2} \| w \|^2 + C \sum_{i=1}^{N} (\xi_i^* + \xi_i) - \sum_{i=1}^{N} (\lambda_i^* \xi_i^* + \lambda_i \xi_i)$$
$$- \sum_{i=1}^{N} \alpha_i^* (\varepsilon + \xi_i^* + y_i - \langle w, x_i \rangle - b)$$
$$- \sum_{i=1}^{N} \alpha_i^* (\varepsilon + \xi_i^* - y_i + \langle w, x_i \rangle + b)$$

where $\lambda_i^*, \lambda, \alpha_i^*, \alpha_i \geq 0$. Then, the dual optimisation problem is obtained based on KKT conditions. According to Awad and Khanna (2015), the KKT conditions include setting the partial derivatives of $L$ with respect to each primal variable ($w$, $b$, $\xi^*$, $\xi$) equal to zero, the partial derivatives of $L$ with respect to each Lagrange multiplier ($\lambda^*$,$\lambda$,$\alpha^*$, $\alpha$) less than or equal to zero, and the product of each Lagrange multiplier ($\lambda^*$,$\lambda$,$\alpha^*$, $\alpha$) and the corresponding constraint equal to zero.

Finally, it yields the following dual optimisation problem.

$$\underset{\alpha, \alpha*}{\text{Maximise}} - \frac{1}{2} \sum_{i,j=1}^{N} (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*) \langle x_i, x_j \rangle$$
$$- \varepsilon \sum_{i,j=1}^{N} (\alpha_i + \alpha_i^*) + y_i \sum_{i,j=1}^{N} (\alpha_i - \alpha_i^*)$$
$$\text{subject to} \begin{cases} \sum_{i,j=1}^{N} (\alpha_i - \alpha_i^*) = 0, \\ \alpha_i, \alpha_i^* \in [0, C]. \end{cases}$$

More details of the formulation can be referred to Awad and Khanna (2015).

The dual formulation of the linear SVR model presented can be extended to the non-linear setting through feature mapping. The kernel trick is applied to map the original input data space into a higher dimensional feature space through a transform kernel function $\varphi$, then constructs the linear equation in the transformed space. See Smola and Schölkopf (2004) and Ranjan (2019) for detailed explanations of the fundamentals behind explicit and implicit mapping and how the kernel trick avoids the curse of dimensionality. One of the commonly used kernel functions is Gaussian Radial Basis Function (RBF), which can be defined as:

$$K(x_i, x_j) = exp\left(-\frac{(x_i - x_j)^2}{2\sigma^2}\right),$$

where $x_i$ and $x_j$ are two sample points, σ is the kernel width with $\sigma > 0$. It can be expressed in simple form as follows:

$$K(x_i, x_j) = exp\,(-\gamma(x_i - x_j)^2),$$

where $\gamma > 0$. Accordingly, $\gamma$ is the only one extra tuneable hyperparameter of the RBF kernel.

According to Ojemakinde (2006), without prior knowledge about the data, the RBF kernel is the preferable choice justified by valid reasons. First, it requires less tuneable hyperparameters than the polynomial kernel. RBF also has fewer numerical difficulties since the kernel value ($\gamma$) ranges from 0 to 1, whereas the range of these values of the polynomial kernel can fall between 0 and ∞. Moreover, although the sigmoid kernel is successfully applicable, it does not always fulfil the requirement for an SVR kernel, called Mercer's condition. The sigmoid kernel is also similar to the RBF kernel when the kernel width is small. In addition, findings from Alahmari (2020) and Saadah et al. (2021) revealed that SVR with RBF kernel demonstrated outstanding prediction performance in price prediction problems. For any kernel type, the SVR model complexity can be affected by the values of $C$ and $\varepsilon$. In this paper, the RBF kernel is applied for CPO price forecasting. Therefore, the tuneable hyperparameters of the RBF-SVR model are $C$, $\varepsilon$ and $\gamma$. Simulated annealing (SA) is used to find near optimal solutions for these hyperparameters to improve the prediction performance.

**SA**

Consider a minimisation problem with the set of all possible feasible solutions $x_1, \ldots, x_n \in X$ where the objective function to be minimised is $f: X \rightarrow \mathbb{R}$. The optimal solution $x^* \in X$ is defined as $f(x^*) \leq f(x_i)$ for all $x_i \in X$, $i \in \{1, \ldots, n\}$.

SA is an iterative algorithm that discovers a series of solutions $x_1, \ldots, x_n \in X$. At each iteration, SA considers a move from the current feasible solution, $x_i$ to another potential feasible solution, $x_{new}$. Assume that the difference between new and current objective function is $\Delta f = f(x_{new}) - f(x_i)$. Then, $x_{new}$ is considered to be worse than $x_i$ if the objective function worsens when moving from $x_i$ to $x_{new}$ ($\Delta f$ is positive).

One significant characteristic of SA is that worsening moves are not neglected but accepted with an acceptance probability $P$. The acceptance probability $P$ depends on the amount of worsening $\Delta f$ and the parameter $T$ such that $T \in \mathbb{R}^+$. According to Metropolis et al. (1953), the acceptance probability is computed through the Metropolis' formula:

$$p = \begin{cases} e^{\frac{-\Delta f}{T}}, \Delta f > 0, \\ 1, \Delta f \leq 0. \end{cases}$$

Note that the probability of accepting a worsening move is high when $\Delta f > 0$ is small and $T$ is large. Also, the probability is always 1 when $\Delta f \leq 0$, which means the improving moves are always accepted by the algorithm. Also, a random number of between 0 and 1 is computed through a uniform distribution. The new move is accepted only if the acceptance probability is greater than the random number. $T$ is initialised and reduced recursively throughout the execution of the algorithm so as to make the worsening moves less likely to happen in the last iteration. According to Fischetti and Stringher (2019), $T$ can be updated using a simple formula $T = \alpha \times T$, with cooling factor $\alpha \in (0,1)$ such that $\alpha \in (0.7, 0.8)$ when cooling is applied after several SA iterations with a constant $T$.

There are four SA parameters used during hyperparameter tuning. They are cooling factor ($\alpha$), number of iterations, initial temperature ($T_0$) and minimum temperature ($T_{min}$). In this work, it would be ideal to choose $\alpha \in (0.7, 0.8)$ as cooling is applied after several iterations at a constant temperature. Hence, an average of the two boundaries, 0.7 and 0.8 is taken, which is 0.75. Secondly, SA is an iteration-intensive algorithm where the number of iterations at any given temperature will affect the duration and the quality of the obtained solution. According to Szykman et al. (1997), the number of iterations needed to achieve global optima depends on the size of the problem, as the number of iterations might be as large as millions. However, Martinez-Rios and Frausto-Solis (2012) used SA with only 100 iterations in solving a nondeterministic polynomial-time complete (NP-complete) problem, which is the "Boolean Satisfiability problem". Thus, considering using 100 iterations in hyperparameter tuning is worthy as this process might involve the search space of a million numbers.

Finally, temperature can have great impact on the overall tuning result, as it affects the acceptance probability. The initial temperature should be large enough to make the acceptance probability close to 1, and the minimum temperature should be much lower such that it decreases the acceptance probability gradually throughout the annealing process. Fischetti and Stringher (2019) chose 1 as their initial temperature and considered temperature reduction of 3 to 5 times, which decreased the acceptance probability low enough. Therefore, the initial temperature of 100 and minimum temperature of 30 are chosen (which is achieved after five temperature reductions) in this study.

**Model Performance Evaluation Metrics**

To evaluate the proposed model's prediction performance, four metrics are used: Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), and R-squared ($R^2$). The formula for each metric:

$$MAE = \frac{1}{N}\sum_{i=1}^{n}|y_i - \hat{y}_i|,$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=0}^{n}(y_i - \hat{y}_i)^2},$$

$$MAPE = \frac{1}{N}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{max(\varepsilon, |y_i|)}\right|,$$

$$R^2 = 1 - \frac{SSE}{SST} = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2},$$

where $n$ is the number of data points, $y_i$ is the actual target value and $\hat{y}_i$ is the predicted value. SST is the sum of squares, SSE is the sum of squared residuals error, and $\bar{y}$ indicates the mean value.

**Hyperparameter Tuning**

The objective of this work is to minimise the evaluation metric obtained by using different sets of hyperparameters of the RBF kernelised SVR model. These hyperparameters ($C$, ε, γ) are the inputs of the SA algorithm. The procedure of hyperparameter tuning is as follows:

1. Randomly choose values for all hyperparameters, assuming it as current state and evaluating model performance with the selected evaluation metric.

2. Obtain new current state by randomly updating the value of one hyperparameter by randomly selecting a value in the neighbourhood to get neighbouring state.
3. If a hyperparameter combination is repeated, repeat Step 2 until a new combination is generated.
4. Evaluate model performance on the neighbouring state.
5. Compare the model performance of neighbouring state to the current state and decide whether to accept the neighbouring state as current state or reject it based on the evaluation metric.
6. According to the result of Step 5, repeat Steps 2 through 5.

For further analysis, the algorithm can intake a set of previously found hyperparameters and then continue the search of new current state around the neighbourhood of previously found hyperparameters.

**Data**

The historical data of FCPO used was downloaded in comma-separated value (CSV) format, by searching for "Palm Oil Futures" on the website (http://www.investing.com). The range of dates for data obtained is from 2 January 2018 to 30 June 2022. The data consists of 7 columns (features) and 1,099 rows. The description of each FCPO data feature is listed in Table 1. Since there are 61 NaN values detected in the "Volume" column, the respective rows are dropped to enable computation purposes, resulting in 1038 rows of data.

Table 1

*FCPO data features*

| Features | Description |
|---|---|
| Date | Trading date when the data is recorded. |
| Open | Price when the market opens. |
| Close | Price when the market closes. |
| High | The highest price reached of a trading day. |
| Low | The lowest price reached of a trading day. |
| Volume | The total amount of trading volume of a trading day. |
| Percentage change | The percentage difference between the close price of current and previous trading day. |

Figure 2 shows the candlestick chart of CPO prices from 2 January 2018 to 30 June 2022. Green candlesticks indicate upward movement while red ones indicate downward movement. From Figure 2, we observe the increasing trend of CPO prices from the middle of May 2020 to the first quarter of 2022, surging to a peak around March 2022. The maximum closing price at MYR8,163.00 per tonne falls on 1 March 2022, while for open, high and low prices are on the next trading day, which is at super-high MYR8,200.00, MYR8,707.00, and MYR7,780.00 per tonne, respectively. After the sharp increase, we notice the CPO price declines until the end of 2022, although the prices in June 2022 are still considered high compared to previous years. In addition, the line plot in Figure 3 shows the direction of the CPO closing price. This simple plot enables us quickly to observe the price trend.

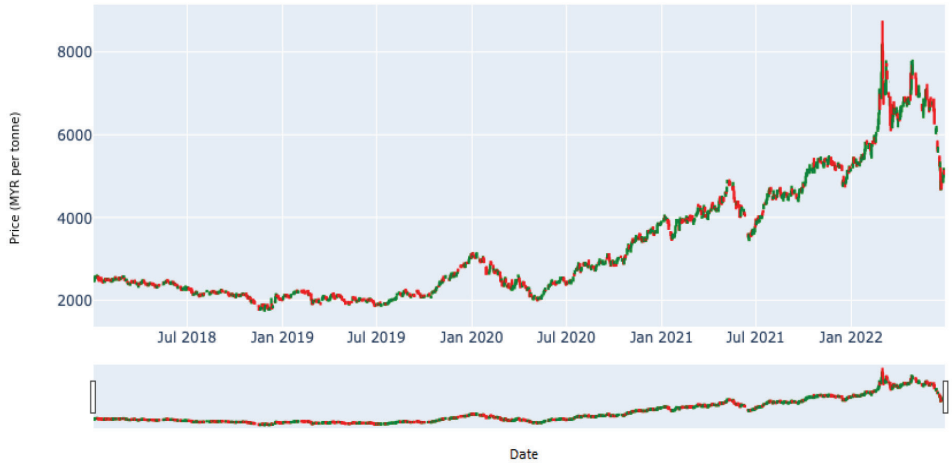



*Figure 2.* CPO prices candlestick chart

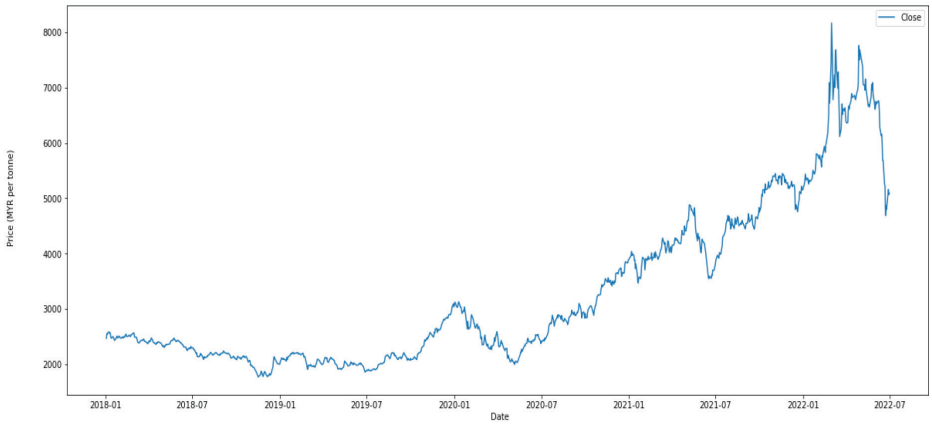



*Figure 3.* Line plot of historical CPO daily close price

## RESULTS AND DISCUSSION

The proposed SA-SVR is implemented in Python using Google Colab with RAM of 1.26 GB and no accelerator.

Before the machine learning model is applied to make predictions, data splitting is performed to partition the dataset into two or more subsets, typically the training and test sets. The training set is used to train and build up the model, while the test set is used for testing the model after completing training. It is common practice for the proportion of train-test splitting to be 6:4, 7:3 and 8:2. In trials, the SA-SVR with RBF kernel model works well in a single run when fitting 70% training data to the model. However, heavy computation and overload issues arise when the algorithm is executed 30 times due to insufficient of computer memory. Almassian et al. (2009) provide evidence that SVM with RBF kernel can have an impressive prediction performance using only 10% data as a training set. The prediction accuracy in their binary classification study reached up to 99.88%. Therefore, we decided to use 1:9 for the train-test data splitting to resolve the preceding issues.

### Comparison of SVR with and without Hyperparameter Tuning

The default hyperparameter setting in SVR module function with RBF kernel are $C = 1$, gamma = 'scale' and epsilon = 0.1. For ease of interpretation, we transform the actual and predicted values into their original scale (MYR) and evaluate the performance metrics (see Table 2). The scale-dependent MAE and RMSE values now have the same scale as the original close price. The MAE value states the mean absolute difference between the actual and predicted closing prices is MYR1,249.68, while RMSE represents the standard deviation of prediction errors (residuals) is MYR1,802.08. The $R^2$ value is also negative, an indicator of poor prediction. Accordingly, the overall prediction error is high and accounts for the conclusion that the model has low predictive power.

As for hyperparameter tuning, after 30 runs with an average execution time as stated in Figure 4, the best combination of hyperparameter values found is $C = 741709$, gamma = $1 \times 10^{-6}$ and epsilon = $1 \times 10^{-7}$ (see Figure 5). After two times of temperature reduction, reaching the temperature of 56.25, SA obtains this best set of hyperparameters as the combination achieves the best metric in prediction.

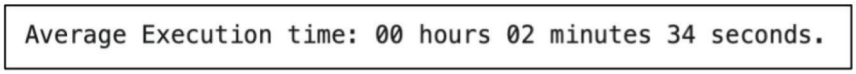



*Figure 4.* Average execution time in 30 runs

| Number of Temperature Reductions | Current Temperature | C | gamma | epsilon |
|---|---|---|---|---|
| 2 | 56.25 | 741709 | 1e-06 | 1e-07 |

*Figure 5*. Result of hyperparameter searching using SA

With these hyperparameters set in the RBF-SVR model, the prediction performance evaluation metrics (in original scale) are shown in Table 2. MAE and RMSE values indicate that the mean absolute difference between the actual and predicted values by SA-SVR is MYR98.48, and the standard deviation of prediction error by SA-SVR is MYR205.62. The prediction error is considered low enough. The predictions made by the fine-tuned SA-SVR model achieve high accuracy as $R^2$ is high. It is undoubted that SA-SVR is a high-performing model with accurate and trustworthy predictions compared to the SVR model with a default hyperparameter setting. Table 2 compares the performance evaluation metrics of both models.

Table 2
*Evaluation metrics of SVR and SA-SVR in original scale*

| Model | MAE | MAPE | RMSE | $R^2$ |
|---|---|---|---|---|
| SVR | 1249.68 | 0.2882 | 1802.08 | –0.4357 |
| SA-SVR | 98.49 | 0.0210 | 205.62 | 0.9813 |

*Notes*: SVR refers to support vector regression, SA-SVR refers to simulated annealing-based support vector regression, MAE refers to mean absolute error, MAPE refers to mean absolute percentage error, RMSE refers to root mean squared error, and $R^2$ refers to R-squared error.

As expected, the error measures MAE, MAPE and RMSE of SA-SVR are all lower than SVR, and $R^2$ of SA-SVR is far greater than SVR. Figure 6 is the time series plot comparing the predictions made by both single and hybrid SVR models. Undeniably, it demonstrates the outstanding predictive ability of the SA-SVR model in close price prediction, showing that hyperparameter tuning contributes enormously to performance improvement. Essentially, the model building uses only 10% of the data as the training set due to the memory-intensive issues. With a small training dataset, the fine-tuned SA-SVR model performs impressively, showing its potential to be a powerful "super-forecaster". Yet, lack of training data may be one of the reasons impacting the performance of the without-tuned SVR model. When the training data size is small, the model may have inadequate

data to capture and learn the pattern from them, and thus results in low predictive ability. This logic leads to the first analysis in the next section.


*Figure 6.* Comparison between SVR and SA-SVR

## Analysis 1: The Effect of Training Data Size to SVR Model

To study the effect of training data size on the model's performance, we conduct a simple experiment by splitting the dataset into the train-test proportions of 1:9, 6:4, 7:3, 8:2, 9:1 and 9.5:0.5. Table 3 depicts the number of training and test data points, and the date range of each. Each training set is fitted to the SVR model with the default hyperparameter setting in the module function. This built model is then used to predict the test set data.

Table 3
*Amount of data in different training and testing sets*

| Train-test ratio | Amount of training data | Date range | Amount of test data | Date range | Total |
|---|---|---|---|---|---|
| 1:9 | 103 | 3 January 2018 – 18 June 2018 | 934 | 19 June 2018 – 30 June 2022 | 1,037 |
| 6:4 | 622 | 3 January 2018 – 07 Oct 2020 | 415 | 08 October 2020 – 30 June 2022 | 1,037 |
| 7:3 | 725 | 3 January 2018 – 12 Mar 2022 | 312 | 16 March 2021 – 30 June 2022 | 1,037 |
| 8:2 | 829 | 3 January 2018 – 20 Aug 2021 | 208 | 23 August 2021 – 30 June 2022 | 1,037 |
| 9:1 | 933 | 3 January 2018 – 21 Jan 2022 | 104 | 24 January 2022 – 30 June 2022 | 1,037 |
| 9.5:0.5 | 985 | 3 January 2018 – 07 April 2022 | 52 | 08 April 2022 – 30 June 2022 | 1,037 |

*Note*: This table shows the train and test data amounts in each data set along with their date ranges.

Table 4 summarises the SVR evaluation metrics according to different train-test splitting ratios. The metric values are obtained using the original-scaled actual and predicted data to avoid misinterpretation. The grey-shaded row indicates the result of the single SVR model fitted with only 10% training data as discussed in the preceding section. The blue-shaded cells indicate the best error metrics.

Table 4
*Evaluation metrics of SVR with different training data size – Original scale*

| Train-test ratio | MAE | MAPE | RMSE | $R^2$ |
|---|---|---|---|---|
| 1:9 | 1249.68 | 0.2882 | 1802.08 | –0.4357 |
| 6:4 | 2196.20 | 0.4139 | 2561.50 | –3.9823 |
| 7:3 | 1529.18 | 0.2649 | 1884.07 | –2.2979 |
| 8:2 | 1547.00 | 0.2500 | 1859.81 | –3.2816 |
| 9:1 | 1724.04 | 0.2565 | 1979.66 | –6.4755 |
| 9.5:0.5 | 695.57 | 0.1054 | 845.25 | –0.1947 |

*Notes*: MAE refers to mean absolute error, MAPE refers to mean absolute percentage error, RMSE refers to root mean squared error, and $R^2$ refers to R-squared error.

According to Table 4, $R^2$ is not an appropriate metric to explain the performance power in this case. Since $R^2$ is defined as the proportion of the variance for the response variable, its value is usually affected by the inherent variability of data. $R^2$ values remain negative in different training data sizes indicating that the predictive ability of this simple SVR model is low.

The SVR model fitted with 95% training data obtains the lowest MAE, RMSE and MAPE, while the highest resulted from 60% training data, indicating its poor prediction performance. However, it is noticeable that the error indicators do not decrease proportionally to the training data size. Figure 7 visualises the performance of SVR in different training data sizes. From the plot, we can see that the predictions improve gradually with the training data size. It seems contradictory to the evaluation metrics displayed in Table 4. This may be due to the different sizes in the test set, and the original scale price is in thousands.

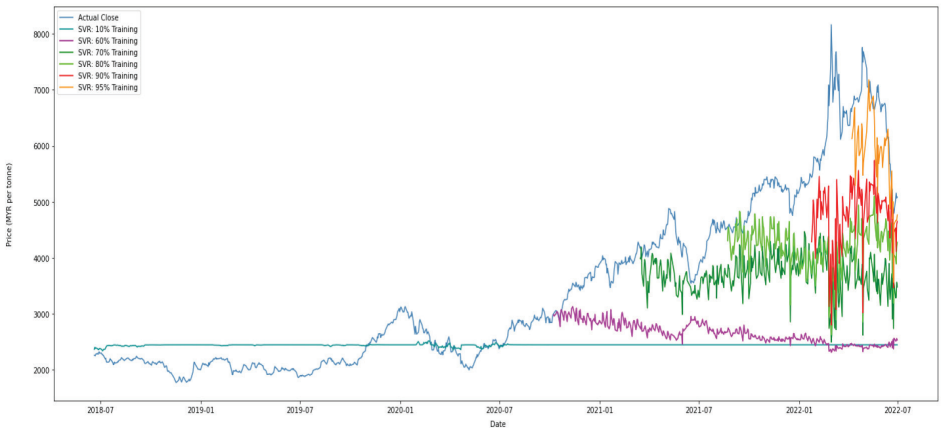



*Figure 7.* Performance of SVR in difference training data size

Figure 8 shows the absolute difference between every actual and predicted price. The orange and brown lines in this plot represent the ratios of 6:4 and 9.5:0.5, respectively. For 9.5:0.5, the 5% test set size consists of only the last 52 trading dates (8 April 2022 to 30 June 2022), and almost all 52 points have the lowest absolute error. For 6:4, the absolute errors of all points in the test set are at a high value, thus resulting in the highest error metric values. However, the plot demonstrates that the absolute errors of 1:9 during the same date range in the 6:4 test set (8 October 2020 to 30 June 2022) are higher. Hence, the large test size () may be the reason for the low RMSE and MAE.

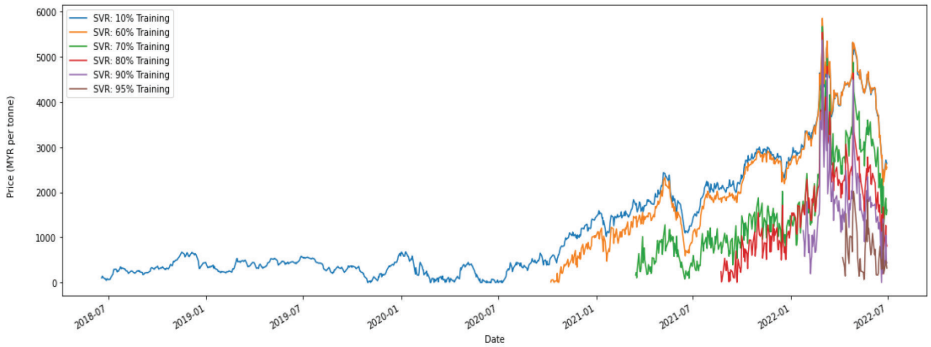



*Figure 8.* Comparison of the absolute difference between each actual and predicted price

The error measure metrics of the 70%, 80% and 90% training sizes are similar. According to Table 4, 70% training size has the most accurate result, followed by 80% and 90%. However, Figure 8 gives a different point of view in the date ranging from 24 January 2022 to 30 June 2022. Their absolute errors are

represented by the green, red and purple lines, respectively. Although there is some overlapping, the green line significantly has the highest absolute error among the three, followed by red and purple. In essence, changing the data splitting ratio does not necessarily improve model performance. From the analysis above, fixing the test set size and using older data to increase the training data size may be a better approach.

**Analysis 2: Selection Temperature Parameters of SA**

The main control parameter in SA is the temperature. It affects the decision on the transition to neighbouring states. Theoretically, an initial temperature that is too high may cause too many changes at the early stages during exploration, while one that is too low may result in an unnecessarily constrained search in a specific area. An experiment by choosing a different combination of initial and minimum temperature is carried out to determine the best tuning result. The cooling factor remains constant. The train-test data size is the same as the initially proposed model, splitting in a 1:9 ratio. Let the initial temperature be $T_0$, and the minimum temperature be $T_{min}$.

Table 5 shows the fine-tuned hyperparameter sets of each temperature combination after 30 runs with the average execution time (in seconds) and their respective RMSE. The combinations of ($T_0$, $T_{min}$) = (150, 40) and ($T_0$, $T_{min}$) = (100, 20 obtain different hyperparameter sets that achieve the same best metric. We consider the mean of them. Relate the values in the RMSE (MYR) column for the original-scaled (1,775.00, 8,316.00).

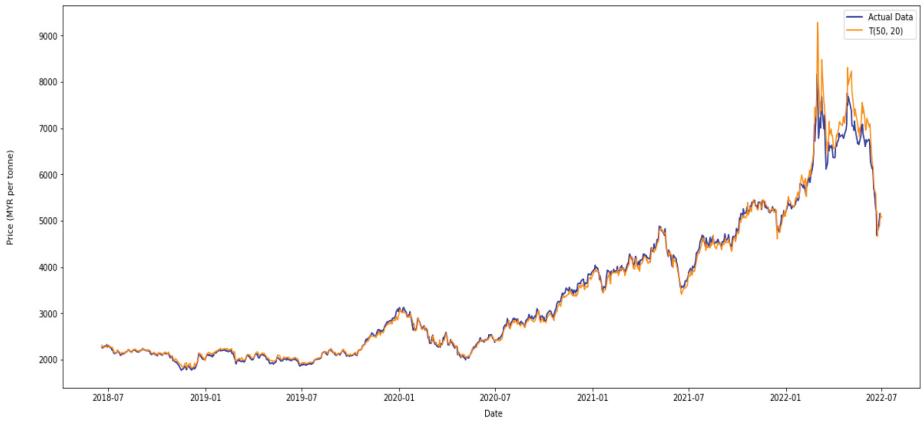
Table 5
*Best hyperparameter set obtained using different temperature combination*

| $T_0$, $T_{min}$ | $C$ | $\gamma$ | $\varepsilon$ | RMSE (MYR) | Average execution time (seconds) |
|---|---|---|---|---|---|
| 50, 20 | 674,541 | $1 \times 10^{-6}$ | $1 \times 10^{-10}$ | 155.12 | 105 |
| 50, 30 | 741,362 | $1 \times 10^{-6}$ | $1 \times 10^{-18}$ | 205.65 | 57 |
| 50, 40 | 1,000,000 | $1 \times 10^{-6}$ | $1 \times 10^{-12}$ | 200.20 | 28 |
| 100, 20 | 900,000 | $1 \times 10^{-6}$ | $1 \times 10^{-12}$ | 251.23 | 165 |
| 100, 30 | 741,709 | $1 \times 10^{-6}$ | $1 \times 10^{-7}$ | 205.62 | 154 |
| 100, 40 | 670,126 | $1 \times 10^{-6}$ | $1 \times 10^{-16}$ | 153.23 | 106 |
| 150, 20 | 741,727 | $1 \times 10^{-6}$ | $1 \times 10^{-8}$ | 203.21 | 212 |
| 150, 30 | 741,301 | $1 \times 10^{-6}$ | $1 \times 10^{-6}$ | 205.53 | 176 |
| 150, 40 | 1,000,000 | $1 \times 10^{-6}$ | $1 \times 10^{-12}$ | 200.20 | 135 |

*Notes*: $T_0$, $T_{min}$ are initial and minimum temperatures, respectively, $C$ is the coefficient of penalty, $\gamma$ is the kernel value of the radial basis function, $\varepsilon$ is the defined precision, and RMSE is the root mean squared error. The blue-shaded cell indicates the best hyperparameter set resulting in the lowest RMSE.

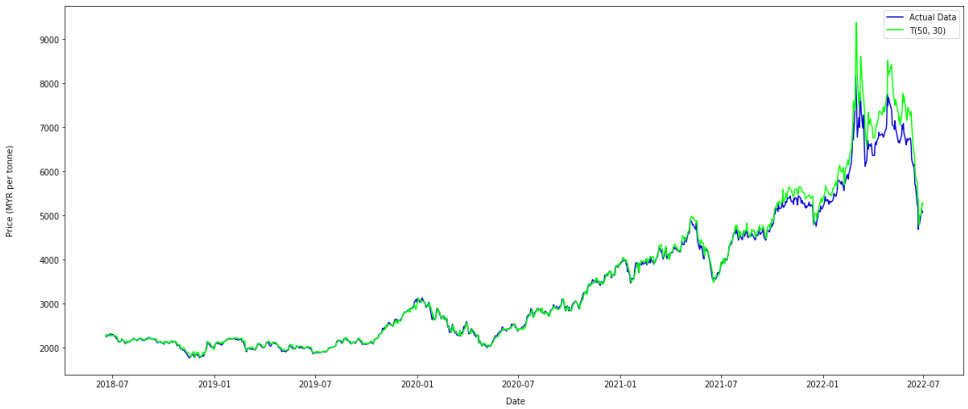
According to the hyperparameter values in Table 5, the best  value obtained by all models is $1 \times 10^{-6}$. The tuning results of (50, 40) and (150, 40) are the same, but the average execution time of (50, 40) is shorter due to the small temperature gap. The combination (100, 20) obtains the same  and  as these combinations, but its *C* value is smaller with higher RMSE value, indicating more prediction error. Recall that *C* is the penalty of the error term, corresponding to a tighter margin. With a high *C* value, the algorithm seeks hardly to reduce the error in the training phase. High *C* value may lead to the overfitting problem, and thus the model does not generalise well. However, these findings show higher *C* obtained by (50, 40) and (150, 40) has lower RMSE implying the prediction of unseen data (test set) exhibits fewer errors compared to the lower *C* obtained by (100, 20). It shows high prediction accuracy, indicating the model is not overfitting. The combinations of (50, 30) and (150, 30) have an almost identical RMSE value to the initially proposed model with (100, 30) temperature settings, which are 205.65, 205.53 and 205.62, respectively. Overall, the result of most combinations has no vast differences. The best combination among all is (100, 40), resulting in the lowest RMSE, which is 153.23. It indicates the standard deviation of the price prediction error is only MYR153.23. The average execution time of this lowest-RMSE combination is relatively short as the temperature gap is 60.

The time series plots for each model with different $T_0$ and $T_{\min}$ are shown in Figures 9 to 11. All models are visualised in Figure 12 and the overlapping lines indicate the predictions of all models have no significant differences and follow the actual price trend.

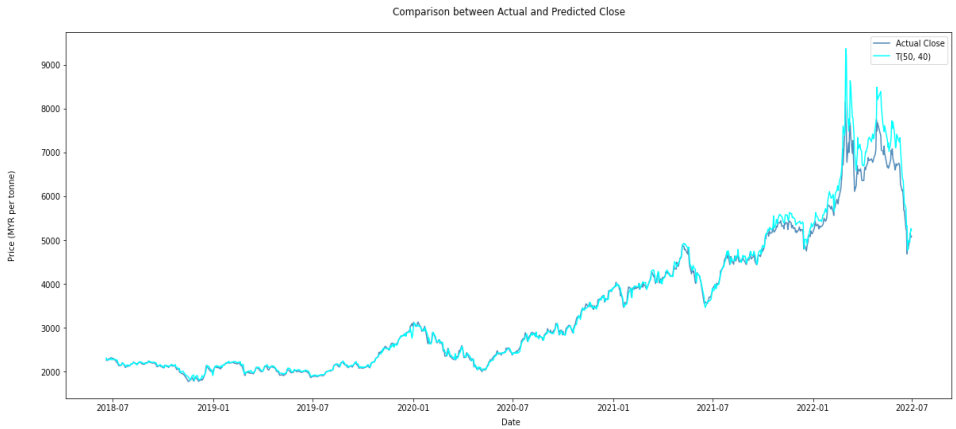



(a) $T_0 = 50$ and $T_{min} = 20$

*Figure 9.* Time series plots of actual and predicted values for $T_0 = 50$ and various $T_{min}$. (a) $T_0 = 50$ and $T_{min} = 20$; (b) $T_0 = 50$ and $T_{min} = 30$; and (c) $T_0 = 50$ and $T_{min} = 40$

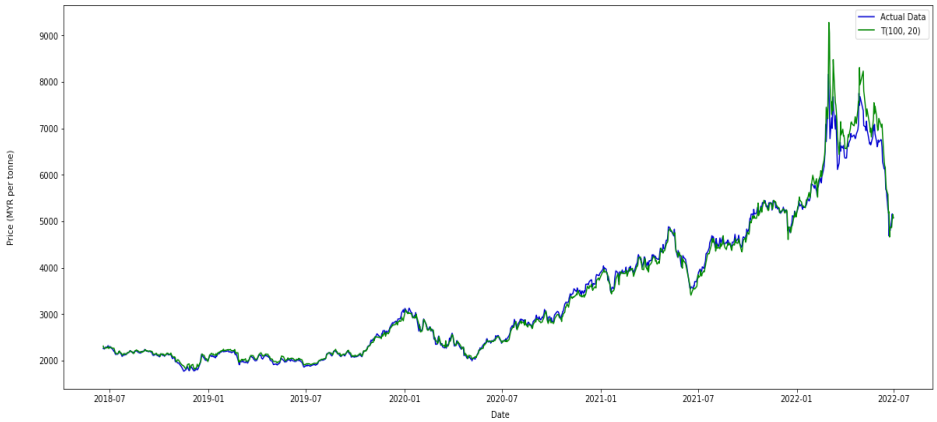



(b) $T_0 = 50$ and $T_{min} = 30$

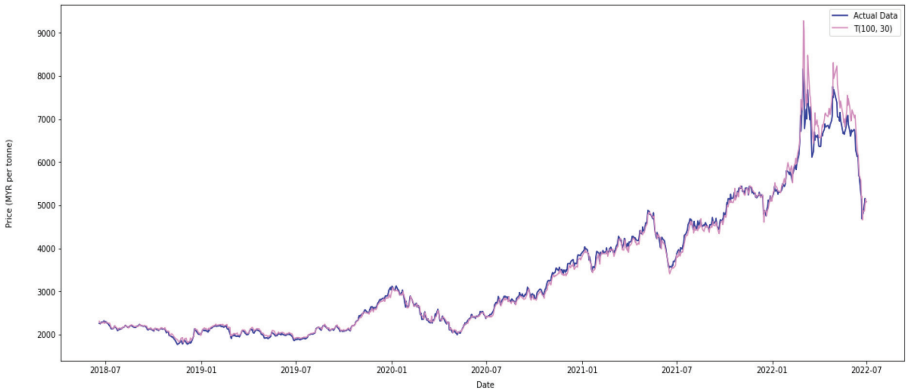



(c) $T_0 = 50$ and $T_{min} = 40$

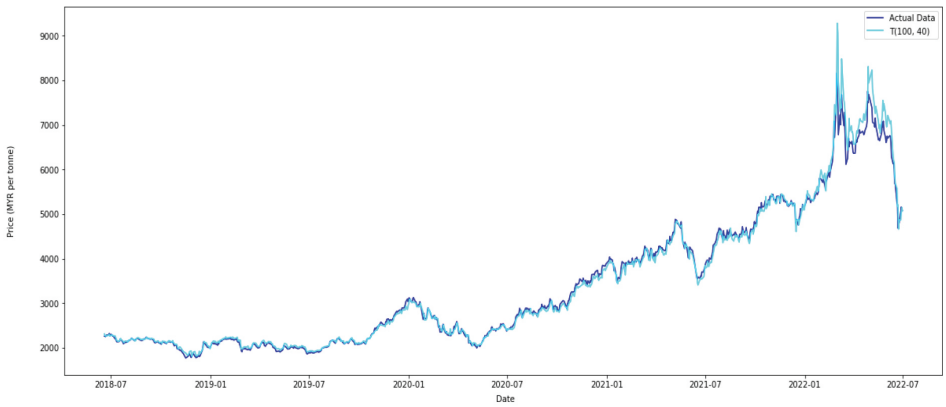
*Figure 9.* Continued




(a) $T_0 = 100$ and $T_{min} = 20$

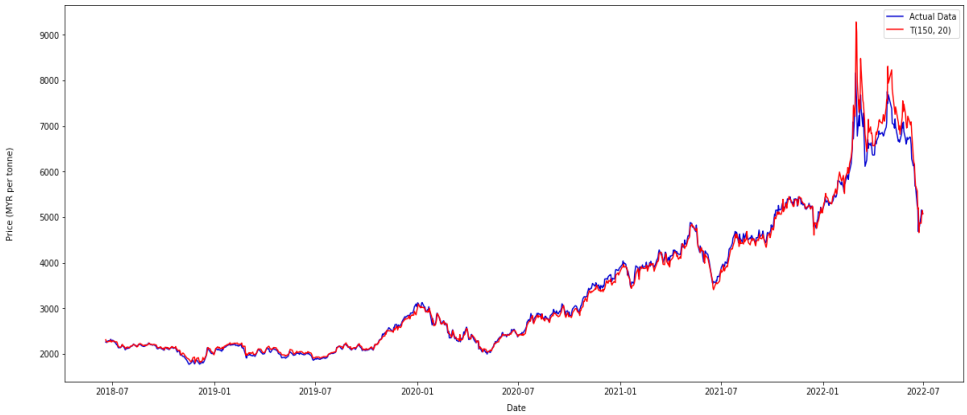



(b) $T_0 = 100$ and $T_{min} = 30$

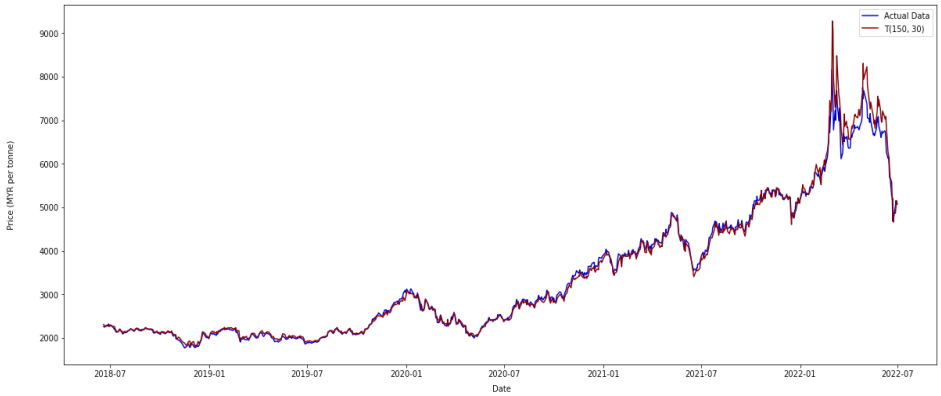



(c) $T_0 = 100$ and $T_{min} = 40$

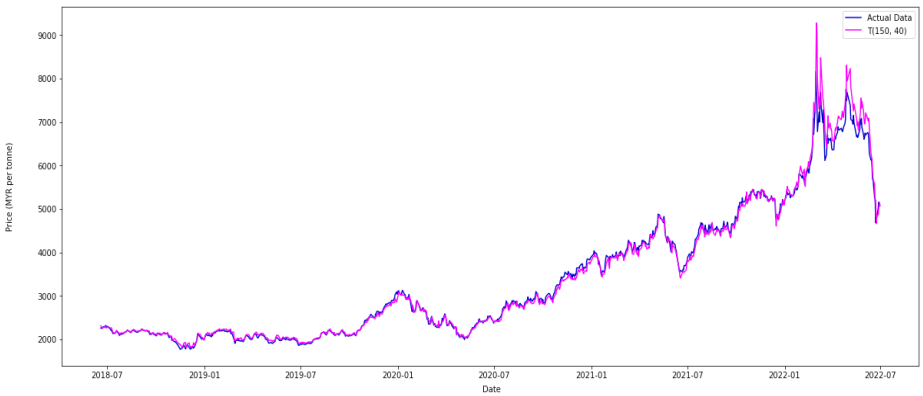
*Figure 10.* Time series plots of actual and predicted values for $T_0 = 100$ and various $T_{min}$. (a) $T_0 = 100$ and $T_{min} = 20$; (b) $T_0 = 100$ and $T_{min} = 30$; and (c) $T_0 = 100$ and $T_{min} = 40$




(a) $T_0 = 150$ and $T_{min} = 20$




(b) $T_0 = 150$ and $T_{min} = 30$




(c) $T_0 = 150$ and $T_{min} = 40$

*Figure 11.* Time series plots of actual and predicted values for $T_0 = 150$ and various $T_{min}$. (a) $T_0 = 150$ and $T_{min} = 20$; (b) $T_0 = 150$ and $T_{min} = 30$; and (c) $T_0 = 150$ and $T_{min} = 40$

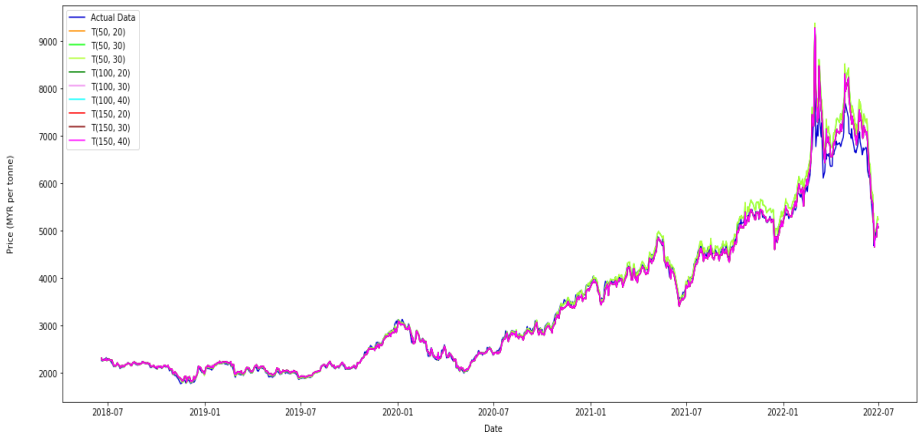



*Figure 12*. Comparison of prediction performances in different temperature combinations

Table 6 summarises the number of temperature reductions and the current temperature where the best hyperparameter set is highlighted. From observation, the initial and minimum temperatures do affect the number of temperature reductions. However, there is no obvious pattern that can be identified. This implies that there is no universally ideal temperature value for all problems, and that initial and minimum temperature combination is problem-dependent.

Table 6
*Number of temperature reduction and current temperature*

| $T_0$, $T_{min}$ | Number of temperature reductions | Current temperature |
|---|---|---|
| 50, 20 | 0 | 50.00 |
| 50, 30 | 1 | 37.50 |
| 50, 40 | 0 | 50.00 |
| 100, 20 | 4 | 31.64 |
| 100, 30 | 2 | 56.25 |
| 100, 40 | 3 | 42.19 |
| 150, 20 | 1 | 112.50 |
| 150, 30 | 4 | 47.46 |
| 150, 40 | 2 | 84.38 |

*Notes:* $T_0$, $T_{min}$ are initial and minimum temperatures, respectively. The blue-shaded cells indicate the number of temperature reduction and current temperature of the best hyperparameter set.

In the next analysis, we repeat the parameter searching from the previous hyperparameter values of the best set, and the temperature setting remains as ($T_0$, $T_{min}$) = (100, 40). The result is discussed in these two aspects.

### Analysis 3: Search of Hyperparameters from New Initial Point

This experiment is conducted by starting the parameter search from the previous hyperparameter values of the best set in Analysis 2. The temperature setting is $T_0 = 100$, and $T_{min} = 40$. After 30 runs with an average execution time stated in Figure 13, the best combination of hyperparameter values found is $C = 670126$, $\gamma = 1e - 06 = 1 \times 10^{-6}$ and $\varepsilon = 1e - 07 = 1 \times 10^{-7}$, which is the same as the best set in Analysis 2.

```
Average Execution time: 00 hours 01 minutes 20 seconds.
```

*Figure 13*. Average execution time after 30 runs

Table 7 includes the temperature reduction counts and the current temperature where the best set is obtained for both models. Previously, the searching process required three times of temperature reduction, from $T_0 = 100$ drops to 42.19. The model with new defined searching point is able to determine the best hyperparameter set in the initial temperature without any temperature reductions.

Table 7
*Number of temperature reductions and current temperature for different initial hyperparameter search point*

| Model | Number of temperature reductions | Current temperature (best hyperparameters) |
|---|---|---|
| Analysis 2 | 3 | 42.19 |
| Analysis 3 | 0 | 100 |

*Note*: This table shows information for Analysis 2 and Analysis 3 models regarding the number of temperature reductions and current temperature.

This experiment shows the importance of having a good starting parameter search point. The results show that having a starting point of the hyperparameter search space that is closer to the best values reduces both the number of temperature reductions and the average execution time to reach the best state. This may lead to significant improvement in the performance of model.

## CONCLUSION

In this work, a hybrid metaheuristic and machine learning model – SA-SVR for crude palm oil price prediction is proposed. This combination of SA with SVR has to date not been applied to FCPO price prediction thus far (to the authors' knowledge). One crucial task is to improve the prediction accuracy of the SVR

model by tuning the hyperparameters using SA. Instead of manually selecting different hyperparameter combinations, SA is incorporated to efficiently search the defined parameter space and find the best set.

In the first experiment, the algorithm is executed 30 times, and the average execution time is recorded. Due to memory overload issues, the RBF kernelised SVR model intakes only 10% of training data. Essentially, with this small training data set, the SA-SVR model can still predict the highly fluctuated price series, where it achieved low error measure metrics MAE, MAPE, RMSE and higher $R^2$. The SA-SVR model performs far better than the SVR model with default (non-tuned) hyperparameter setting in the module function.

Several analyses via numerical experiments were conducted. These include studying the effect of training data size on the model performance by using different train-test split ratios, determining the temperature parameters in SA, and starting the parameter search from previously obtained hyperparameter values of the best set. In the first analysis, it is good practice to fix the test size and extract more historical price data for model training instead of varying the train-test split ratio. Crucially, from our temperature schedule strategy in the second analysis, it is notable that different combinations of temperatures affect the overall optimisation results. We believe that there is no universally ideal temperature value for all problems. From the third analysis, if the starting point of the hyperparameter search space is close to the best values, the number of temperature reductions and average execution time to reach the best state decreases.

In conclusion, this study showed that using SA with SVR improves the accuracy of FCPO price prediction compared to using SVR on its own. Further experimentation showed that having more training data leads to higher prediction accuracy, SA temperature parameter value is problem-dependent, and the starting point of the search space affects execution time. Thus, an appropriate selection of hyperparameter values may further improve the model prediction performance as shown by the excellent prediction ability of the proposed SA-SVR model in CPO price.

## LIMITATIONS AND FUTURE WORK

In this project, a very specific set of parameters are chosen to be tuned. One can also choose a different set of parameters to be tuned by the SA algorithm for the SVR hyperparameters. In addition, the current search size of the neighbourhood solution is 25% around the current solution, and one can choose to use different search sizes and evaluate the performance of the SVR.

This study only considers SA in optimising the hyperparameters for the SVR. One can consider using other metaheuristic methods such as particle swam optimisation (PSO), genetic algorithm (GA), and ant colony optimisation (ACO). In addition, one can hybridise the metaheuristic methods. For example, SA-GA, PSO-GA and ACO-SA are some popular hybridised metaheuristics methods. One can also choose to optimise the hyperparameters of other machine learning models like artificial neural network and random forest regression using SA or hybridised methods. Then, one can compare the performance among these models with our proposed SA-SVR model.

Another suggestion for future work is to apply the proposed method (SA-SVR) to predict other continuous data. Potential applications include predicting stock price, bond price and foreign exchange rate. In addition, including more relevant information (if available) for a specific dataset may improve the prediction as there are more features for the machine learning model to learn.

## REFERENCES

Aini, H., & Haviluddin, H. (2019). Crude palm oil prediction based on backpropagation neural network approach. *Knowledge Engineering and Data Science*, *2*, 1–9. https://doi.org/10.17977/um018v2i12019p1-9

Alahmari, S. A. (2020). Predicting the price of cryptocurrency using support vector regression methods. *Journal of Mechanics of Continua and Mathematical Sciences, 15*(4), 313–322. https://doi.org/10.26782/jmcms.2020.04.00023

Almassian, N., Azmi, R., & Berenji, S. (2009). AIDSLK: An anomaly based intrusion detection system in Linux Kernel. *Communications in Computer and Information Science*, *31,* 232–243. https://doi.org/10.1007/978-3-642-00405-6_26

Alwadi, S., Almasarweh, M., & Alsaraireh, A. (2018). Predicting closed price time series data using ARIMA model. *Modern Applied Science*, *12*, 181. https://doi.org/10.5539/mas.v12n11p181

Amal, I., Tarno, & Suparti. (2021). Crude palm oil price prediction using multilayer perceptron and long short-term memory. *Journal of Mathematics and Computer Science, 11*(6), 8034–8045.

Arshad, F. M., & Ghaffar, R. A. (1986). Crude palm oil price forecasting: Box-Jenkins approach. *Pertanika Journal of Tropical Agricultural Science, 9*(3), 359–367.

Awad, M., & Khanna, R. (2015). Support vector regression. In M. Awad, & R. Khanna (Eds.), *Efficient learning machines: theories, concepts, and applications for engineers and system designers* (pp. 67–80). Springer Nature.

BURSA Malaysia. (2021). *Crude palm oil futures (FCPO)*. https://www.bursamalaysia.com/trade/our_products_services/derivatives/commodity_derivatives/crude_palm_oil_futures

Dong, Y., Li, S., & Gong, X. (2017, January). Time series analysis: An application of ARIMA model in stock price forecasting. In *Proceedings of the 2017 International Conference on Innovations in Economic Management and Social Science (IEMSS 2017)* (pp. 2352–5428). Atlantis Press. https://doi.org/10.2991/iemss-17.2017.140

Drucker, H., Burges, C. J., Kaufman, L., Smola, A., & Vapnik, V. (1996). Support vector regression machines. In M. C. Mozer, M. Jordan, & T. Petsche (Eds.), *Advances in neural information processing systems* (vol. 9, pp. 155–161). MIT Press.

Fischetti, M., & Stringher, M. (2019). *Embedded hyper-parameter tuning by simulated annealing.* ArXiv. https://doi.org/10.48550/arXiv.1906.01504

Funde, Y., & Damani, A. (2023). Comparison of ARIMA and exponential smoothing models in prediction of stock prices. *The Journal of Prediction Markets, 17*(1), 21–38.

González-Mancha, J. J., Frausto-Solís, J., Castilla Valdez, G., Terán-Villanueva, J. D., & González Barbosa, J. J. (2018). Financial time series forecasting using Simulated Annealing and Support Vector Regression. *International Journal of Combinatorial Optimization Problems and Informatics*, *8*(2), 10–18. https://ijcopi.org/ojs/article/view/9

Hussin, M., Ismail, Z., & Ilias, I. S. C. (2023). Bayesian Network Design for Crude Palm Oil (CPO) price prediction driven by fluctuation patterns and trends. *Journal of Advanced Research in Applied Sciences and Engineering Technology*, *31*(2), 117–129. https://doi.org/10.37934/araset.31.2.117129

Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: Principles and practice* (2nd ed.). OTexts.

Jaquart, P., Dann, D., & Weinhardt, C. (2021). Short-term bitcoin market prediction via machine learning. *The Journal of Finance and Data Science*, *7,* 45–66. https://doi.org/10.1016/j.jfds.2021.03.001

Kasturi, K., Salim, N., Sukprasert, A., Krishnan, R., & Hashim, U. (2017). Multivariate time series forecasting of crude palm oil price using machine learning techniques. *IOP Conference Series: Materials Science and Engineering*, *226*, 12117. https://doi.org/10.1088/1757-899X/226/1/012117

Kasturi, K., Salim, N., Sukprasert, A., Krishnan, R., & Hashim, U. R. (2020). A comparative study on univariate time series based crude palm oil price prediction model using machine learning algorithms. *International Journal of Advanced Trends in Computer Science and Engineering*, *9*, 5802–5806. https://doi.org/10.30534/ijatcse/2020/238942020

Li, X. M., Li Xing, D., Jin Hu, L., & Li, I. L. (2010). Building cooling load forecasting based on support vector machines with simulated annealing. *Advanced Materials Research*, *108–111*, 1003–1008. https://doi.org/10.4028/www.scientific.net/AMR.108-111.1003

Lin, S. W., Lee, Z. J., Chen, S. C., & Tseng, T. Y. (2008). Parameter determination of support vector machine and feature selection using simulated annealing approach. *Applied Soft Computing, 8*(4), 1505–1512. https://doi.org/10.1016/j.asoc.2007.10.012

Madre, Y., & Devuyst, P. (2016, 29 April). *Are futures the future for farmers?* https://www.farm-europe.eu/travaux/are-futures-the-future-for-farmers-2/

Malaysia Palm Oil Council. (2020). *About palm oil*. https://mpoc.org.my/

Martinez-Rios, F., & Frausto-Solis, J. (2012). A simulated annealing algorithm for the satisfiability problem using dynamic Markov chains with linear regression equilibrium. In M. d. S. G. Tsuzuki (Ed.), *Simulated annealing: Advances, applications and hybridizations* (pp. 1–302). InTechOpen. https://doi.org/10.5772/46175

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics, 21*(6), 1087–1092. https://doi.org/10.1063/1.1699114

Mohd Nain, F. N., Ahamed Hassain Malim, N. H., Abdullah, R., Abdul Rahim, M. F., Ahmad Mokhtar, M. A., & Mohamad Fauzi, N. S. (2022). A review of an artificial intelligence framework for identifying the most effective palm oil prediction. *Algorithms, 15*(6), 218. https://doi.org/10.3390/a15060218

Obthong, M., Tantisantiwong, N., Jeamwatthanachai, W., & Wills, G. (2020). *A survey on machine learning for stock price prediction: Algorithms and techniques* [Paper presentation]. The 2nd International Conference on Finance, Economics, Management and IT Business. Vienna House Diplomat Prague, Prague, Czech Republic. 5–6 May, pp. 63–71. https://doi.org/10.5220/0009340700630071

Ofuoku, M., & Ngniatedema, T. (2022). Predicting the price of crude palm oil: A deep learning approach. *International Journal of Strategic Decision Sciences, 13*(1), 1–15. https://doi.org/10.4018/IJSDS.305830

Ojemakinde, B. (2006). Support vector regression for non-stationary time series. [Master's thesis, University of Tennessee]. https://trace.tennessee.edu/utk_gradthes/1756

Oliva, D., Houssein, E. H., & Hinojosa, S. (2021). *Metaheuristics in machine learning: Theory and applications.* Berlin: Springer. https://doi.org/10.1007/978-3-030-70542-8

Pai, P. F., & Hong, W. C. (2005). Support vector machines with simulated annealing algorithms in electricity load forecasting. *Energy Conversion and Management, 46*(17), 2669–2688. https://doi.org/10.1016/j.enconman.2005.02.004

Ranjan, C. (2019, 9 May). *Understanding the Kernel Trick with fundamentals*. Towards Data Science. https://towardsdatascience.com/truly-understanding-the-kernel-trick-1aeb11560769

Saadah, S., Fakhira Zahra, Z., & Hasna Haifa, Z. (2021). Support Vector Regression (SVR) dalam memprediksi harga minyak kelapa sawit di Indonesia dan nilai tukar mata uang EUR/USD [Support Vector Machine (SVM) to predict crude oil palm in indonesia and exchange rate of EUR/USD]. *Journal of Computer Science and Informatics Engineering (J-Cosine)*, *5*, 85–92. https://doi.org/10.29303/jcosine.v5i1.403

Salman, N., Lawi, A., & Syarif, S. (2018). Artificial Neural Network backpropagation with particle swarm optimization for crude palm oil price prediction. *Journal of Physics: Conference Series*, *1114*, 12088. https://doi.org/10.1088/1742-6596/1114/1/012088

Shabri, A., & Hamid, M. F. A. (2019). Wavelet-support vector machine for forecasting palm oil price. *Malaysian Journal of Fundamental and Applied Sciences, 15*, 398–406. https://doi.org/10.11113/mjfas.v15n3.1149

Shahbandeh, M. (2021, 16 July). Export volume of palm oil worldwide in 2020/21. *Statista*. https://www.statista.com/statistics/620219/palm-oil-export-volume-worldwide-by-country/

Shahid, S., & Rahaman, A. (2020). Exponential smoothing methods for detection of the movement of stock prices. *International Journal of Recent Technology and Engineering*, *8*(5), 1420–1422. https://doi.org/10.35940/ijrte.E6409.018520

Siddique, N., & Adeli, H. (2016). Simulated annealing, its variants and engineering applications. *International Journal on Artificial Intelligence Tools*, *25*, 1630001. https://doi.org/10.1142/S0218213016300015

Silalahi, D. (2013). *Application of Neural Network Model with genetic algorithm to predict the international price of Crude Palm Oil (CPO) and Soybean Oil (SBO)* [Paper presentation]. 12th National Convention on Statistics (NCS). EDSA Shangri-La Hotel, Mandaluyong City, Philippines, 1–2 October.

Smola, A. J., & Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and Computing, 14*(3), 199–222. https://doi.org/10.1023/B:STCO.0000035301.49549.88

Szykman, S., Schmidt, L. C., & Shetty, H. (1997). *Improving the efficiency of simulated annealing optimization through detection of productive search* [Paper presentation]. Proceedings of the ASME Design Engineering Technical Conference, Sacramento, California, 14–17 September. https://doi.org/10.1115/DETC97/DAC-3980

Tan, Y. F., Ong, L. Y., Chew, L., & Goh, Y. X. (2021). Exploring time-series forecasting models for dynamic pricing in digital signage advertising. *Future Internet*, *13*, 241. https://doi.org/10.3390/fi13100241

Xie, W., Yu, L., Xu, S., & Wang, S. (2006). *A new method for crude oil price forecasting based on support vector machines*. In V. N. Alexandrov, G. D. van Albada, P. M. A. Sloot, J. Dongarra (Eds.), *Computational Science – ICCS 2006. ICCS 2006. Lecture Notes in Computer Science* (vol. 3994). Berlin, Heidelberg: Springer. https://doi.org/10.1007/11758549_63

Yee, K. W., & Samsudin, H. B. (2021). Comparison between artificial neural network and ARIMA model in forecasting palm oil price in Malaysia. *International Journal of Scientific Engineering and Science, 5*(11), 12–15.

Zeng, D., Liu, Y., Jiang, L., Li, L., & Xu, G. (2012). Wick sintered temperature forecasting based on support vector machines with simulated annealing. *Physics Procedia*, *25*, 427–434. https://doi.org/10.1016/j.phpro.2012.03.107