
Journal of the Chinese Institute of Engineers

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/tcie20

# Optimizing the performance of MLP and SVR predictors based on logical oring and experimental ranking equation

Nadia Al-Rousan & Hazem Al-Najjar

**To cite this article:** Nadia Al-Rousan & Hazem Al-Najjar (2021) Optimizing the performance of MLP and SVR predictors based on logical oring and experimental ranking equation, Journal of the Chinese Institute of Engineers, 44:2, 149-157, DOI: 10.1080/02533839.2020.1856726

**To link to this article:** https://doi.org/10.1080/02533839.2020.1856726

Published online: 11 Jan 2021.

JOURNAL OF THE CHINESE INSTITUTE OF ENGINEERS
2021, VOL. 44, NO. 2, 149–157
https://doi.org/10.1080/02533839.2020.1856726



# Optimizing the performance of MLP and SVR predictors based on logical oring and experimental ranking equation

Nadia Al-Rousan and Hazem Al-Najjar

Department of Computer Engineering Faculty of Engineering and Architecture, Istanbul Gelisim University, Istanbul, Turkey

**ABSTRACT**
Improving conventional prediction systems is widely used to optimize the learning process, achieve higher performance, and avoid overfitting. This paper's purpose is to propose a new predictor for solar tracking systems applications based on oring operator and ranking equation with a conventional predictor including Multi-Layer Perceptron (MLP) and Support Vector Machine Regression (SVR). The point of using oring and ranking equation is to create a new variable that stores the information of combined attributes. This process aims to increase the accuracy of predictors and increase the efficiency of intelligent solar tracking systems. The experiments used 6 different datasets for solar tracking systems. The results revealed that the proposed predictors performed better than conventional predictors. Using the proposed predictors has improved both Root Mean Square Error (*RMSE*) and Coefficient of Determination ($R^2$). The developed MLP models showed lower *RMSE* and higher $R^2$ compared to conventional MLP models. The improvement ranges for using MLP are from 1.0013 to 1.4614 degrees for *RMSE*, and from 1.0019 to 1.4984 times for $R^2$, while the improvement ranges using SVM are from 1.001 to 1.988 degrees for *RMSE* and from 1.000 to 2.385 times for $R^2$.

**ARTICLE HISTORY**
Received 21 October 2019
Accepted 16 November 2020

**KEYWORDS**
Optimization model; multilayer perceptron; support vector machine regression; single axis tracker; prediction model

## 1. Introduction

Prediction models have been studied and analyzed in different fields globally (i.e. chemistry, biology, environmental engineering, computers, medicine, and energy systems).

Recently, prediction models have been used to predict future behavior based on current events which could help decision makers to take a decision (AL-Rousan, Isa, and Desa 2018; Mosavi, Ozturk, and Chau 2018). Several prediction models have been used globally (i.e., multi-layer perceptron, support vector machine, fuzzy system, decision tree, and so on) (Zhang, Patras, and Haddadi 2019; Al-Najjar and Hassan 2016). On the other hand, conventional prediction models suffer from several problems that make them not suited for complex problems. Low performance is the main important problem. Optimizing and developing the current prediction models based on integrating conventional models with other techniques is a promising solution (AL-Rousan, Isa, and Desa 2018; Mosavi, Ozturk, and Chau 2018; Zhang, Patras, and Haddadi 2019; Al-Najjar and Hassan 2016; Osowski, Siwek, and Markiewicz 2004; AL-Rousan et al. 2012). Several studies have proposed new predictor models based on this solution. Kordos and Rusiecki (2013) have improved the MLP predictor model by eliminating the influence of outliers. The idea of this prediction model is to improve the accuracy of a heart disease predictor by using feature selection techniques. In addition, different prediction models have been tested individually to search for the optimum model to achieve that targeted goal (i.e., Decision Tree, Logistic regression, Logistic regression SVM, Naïve Bayes, and Random forest). The results have proven that the accuracy of the Logistic Regression and Naïve Bayes predictors are higher than the accuracy of conventional models. Mata-Moya et al. (2015) have improved the conventional MLP using Constant False Alarm Rate techniques to improve coherent radar detection. To check and validate the proposed model, complex target trajectory scenarios were used. The results found that the proposed model is efficient compared to conventional models. Stowe et al. (2018) have improved a deep Learning model to predict human behavior during hurricanes using social media data collected from Twitter. The results found that deep learning is better than a feature-based model in predicting human behavior. Talreja et al. (2018) have proposed a fast, low-cost storage system using hashing-based image retrieval to improve the retrieval process for face images. The results showed that the proposed model achieved higher accuracy and performed faster compared to most retrieval algorithms.

Al-Najjar, Alhady, and Saleh (2019) have proposed a new prediction method based on linear regression and fitting models to predict the run time of a job in a distributed system. The results showed that the proposed model is efficient with low error and high prediction rates. Htwe and Kham (2019) have proposed a prediction model based on MLP and a genetic algorithm to detect malicious activities in networks. Dai et al. (2019) have proposed a deep neural network model based on Entity Linking Algorithm to predict words in newscasts. The results revealed that the proposed algorithm achieved higher performance compared to conventional algorithms.

Moreover, several methods, models, techniques, and technologies have been proposed in different fields to maximize

**CONTACT** Hazem AL-Najjar hazem_najjar@yahoo.com Department of Computer Engineering, Faculty of Engineering and Architecture, Istanbul Gelisim University, Istanbul 34310, Turkey

the prediction rate, minimize the error ratio, speed up the processing time, or simplify the prediction techniques (Panigrahi and Behera 2020; Jaouedi, Boujnah, and Bouhlel 2019; Tan 2020; Abdualrhman and Padma 2019; Cerri, Barros, and de Carvalho 2014; Shao et al. 2019; Chen 2019; Qi et al. 2019).

Several prediction models have been used in solar tracking systems globally (AL-Rousan, Isa, and Desa 2020). Solar tracking systems can be defined as systems that aim to drive solar photovoltaic modules toward the optimum angles relative to the trajectory of the sun across the sky. Tracking the optimum angles of solar exposure can produce efficient sun incidence, produce optimum-gained power, and increase the efficiency of solar systems as well (Srikumar and Saibabu 2020). Thus, solar tracking systems are preferable compared to stationary solar photovoltaics. Two main types of solar tracking systems exist, namely, single-axis and dual-axis. The difference between them is mainly the number of angles (directions) that are considered during tracking of the trajectory of the sun across the sky.

Using intelligent predictors to predict the optimum angles of solar tracking systems is a hot topic nowadays (AL-Rousan, Isa, and Desa 2018). Several intelligent solar tracking controllers have been proposed and implemented to increase the performance of solar systems. Intelligent techniques that are used in predicting optimum angles for solar tracking systems vary from one system to another. The majority of these proposed systems are based on existing techniques and metrological data. Similar to all intelligent models, these predictors suffer from long processing time and slow convergence.

This paper proposes a new improvement in prediction systems to increase the accuracy. Multi-Layer Perceptron (MLP) and Support Vector Machine Regression (SVR) are used with oring operator and experimental ranking equation. The main objectives of this article are to improve prediction models by simplifying and speeding up processes adopted to implement these models. In addition, these processes can be used to enhance the performance of the conventional techniques without applying any change to internal design and architecture. To the best of the authors' knowledge, no prior published research examines and evaluates using predicting the optimum angles of solar tracking systems by optimizing Multi-layer perceptron or support vector regression based on logical oring and Experimental Ranking Equation. Using such an approach would decrease the processing time and speed the convergence as well.

Section 2 explains the preliminaries and definitions used in the proposed model. The proposed methodology is discussed in Section 3. The results, discussion, and analysis of the proposed predictors are explained in Section 4. Finally, the conclusions and future trends are drawn in Section 5.

## 2. Preliminaries and definitions

This section is to explain the main mechanisms and preliminaries used to improve the performance of multi-layer perceptron predictor. This section is divided into three parts, namely single-axis solar tracking system, support vector machine, and multi-layer perceptron neural network. The first part focuses on explaining the definition of solar tracking systems and single-axis solar tracking systems as well. The second part presents the support vector machine model, how it works, and its main kernels. The third part discusses the multi-layer perceptron neural network predictor.

### 2.1. *Single-axis solar tracking system*

A photovoltaic system is a solar power system that consists of several components to absorb and convert solar power into electricity. Solar tracking system is the most popular solar power system globally (Deb and Roy 2012). It has been designed and developed to track the trajectory of the sun across the sky (AL-Rousan et al. 2012), and to keep the solar panel at the optimum angle that can produce the best power output (Desa et al. 2016). Using solar tracking systems can increase the input of solar radiation, therefore, the output of electrical energy can also be increased (Randall 2016). Single-axis solar tracking system is the most popular type of solar tracking system. Single-axis solar tracking system is a unidirectional system that can move horizontally or vertically (Juswanto and Ali 2016). Tilt ($\theta^\circ$) and orientation ($\phi^\circ$) angles are used to track the position of the sun sufficiently. The tilt angle can be defined as the angle between the solar tracking system and the horizontal axis while the orientation angle is the angle that can be used to move the solar tracking system horizontally to ensure that the sun is perpendicular to the solar tracking system surface as shown in Figure 1. The main objective of solar tracking systems is to choose the best tilt and orientation angles that allow the systems to gain more power through solar radiation.

Several variables have been used to build solar tracking systems including orientation and tilt angles, the gained photovoltaic power, the power radiated from the sun, and the current and voltage flow through the photovoltaic system.

### 2.2. *Support Vector Machine (SVM)*

Support Vector Machine (SVM) is a popular machine learning model that is used for classification and regression. Support Vector Machine Regression (SVR) is used for continuous values, and it is considered as a nonparametric technique because it relies on kernel functions (Drucker et al. 1997). A kernel function or mapping function is used to perform non-linear mapping between the input space and the feature space (Wu, Tzeng, and Lin 2009). Kernel function can transform the training data so that a non-linear decision surface is transformed to a linear equation in a higher number of dimensions (Üstün, Melssen, and Buydens 2006).

The main function of SVR is to minimize the generalization error bound, thus, to achieve generalized performance. SVR is based on calculating a linear regression function in a high dimensional feature space where the input data are mapped via a nonlinear function such as the following:

Suppose the training data $(x_1, y_1), \ldots, (x_i, y_i)$ where, the target value $y_i \in R$, $i$ is the number of features. Thus, SVR is used to find a function $f(x_i) \approx y_i$ that deviates from $y_i$ by a value not greater than the value of $\varepsilon$ for each training data point $x_i$ and it minimizes the $\varepsilon$-insensitive loss-function $R^{\varepsilon}_{Emp}$ such that:

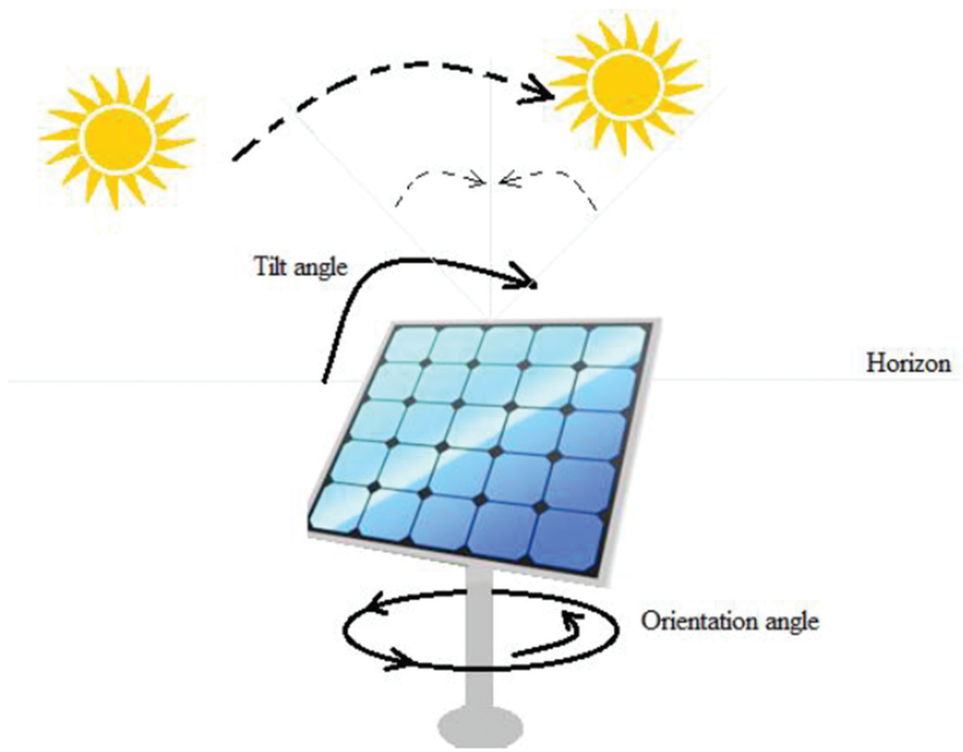


Figure 1. Tilt and orientation angles.

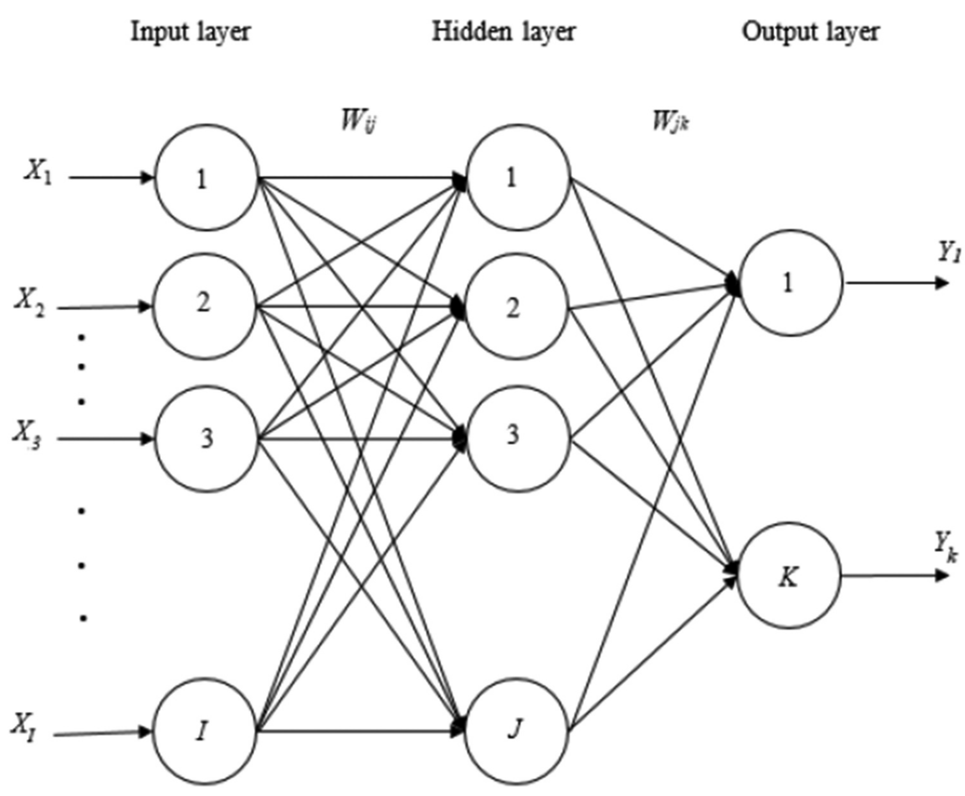


Figure 2. The architecture of MLP network for one hidden layer.

$$R^{\varepsilon}_{Emp} = |f(x) - y|_{\varepsilon} = \max(0, |f(x) - y| - \varepsilon). \quad (1)$$

Two main types of support vector machine regression exist, namely linear support vector machine, and non-linear support vector machine. The main difference between the two types is the loss function that is used to include a distance measure. The loss function of linear SVR depends on ignoring the error values that are within the distance. However, the kernel function varies from one type to another. The loss function of linear SVR depends on ignoring the error values that are within the distance $\varepsilon$ of the observed value $y$by supposing them equal to zero as in the following:

$$L_{\varepsilon} = \left\{ \begin{array}{c} 0, \text{if}|y - f(x)| \leq \varepsilon \\ |y - f(x)| - \varepsilon, \text{otherwise} \end{array} \right\}. \quad (2)$$

On the other hand, a nonlinear kernel function is used to transform maps x to a high-dimensional space in non-linear SVR. In addition, several types of non-linear SVR exist including Quadratic SVR, Cubic SVR, Fine Gaussian SVR, Median Gaussian SVR, Coarse Gaussian SVR, etc. The difference between these types is mainly the kernel scale, therefore the kernel function.

### 2.3. *Multi-layer perceptron (MLP)*

Multi-layer perceptron is an artificial neural network model wherein connections between its units are sequential. The architecture of MLP contains three types of layers namely, input layer, hidden layer(s), and output layer as shown in Figure 2 (Haykin 2009). Each layer consists of independent processing units called neurons. Neurons can process information between layers that are highly interconnected using weights (Landwehr, Hall, and Frank 2005).

As shown in Figure 2, the function of input layer is to accept the input data, while hidden layer is used to accept the outputs from input layer, weight them, and propagate them to output layer. The targeted results are produced by the output layer (Zhang and Gupta 2000).

The prediction output of $k_{\text{th}}$ node in the output layer is calculated as follows:

$$y_k = f\left(\sum_{j=1}^{2} O_j \, w_{jk} + b_k\right), \quad (3)$$

where $y_k$ is the output of $k_{\text{th}}$ node in the output layer, $K$ is the number of nodes in the output layer, $f(.)$ is the transfer function, $w_{jk}$ is the weight from the $j_{\text{th}}$ node in the hidden layer to the $k_{\text{th}}$ hidden node in the output layer, and $b_k$ is the bias for the $k_{\text{th}}$ output node.

The performance of multi-layer perceptron neural network mainly depends on tuning several parameters namely, bias, weight, number of hidden layers, number of hidden nodes, and the type of transfer function. However, the bias is a pseudo input that passes to each neuron in the hidden and the output layers to overcome the problems when the values of input pattern are zero. Weight values are associated with each node in the network to constrain how input data are related to output data (Koutsoukas et al. 2017). In addition, the activation function is a nonlinear function that is used to produce the actual output for the neuron by using the weighted sum of inputs.

The time complexity can be defined as time taken to define and set the new weights to calculate the outputs of multi-layer perceptron. The time complexity for single layer network can be expressed using the big $O$ notation as shown in Equation (2) (Mizutani and Dreyfus 2001).

$$Complexity_{MLP} = O(d \times h) + O(h \times c), \quad (4)$$

where $d$ is the number of input variables, $h$ denotes the number of hidden nodes in the hidden layer, and $c$ denotes the number of output variables.

## 3. Proposed methodology

This section is to discuss the proposed methodology used to design the prediction model. The methodology is divided into three main phases, namely dataset description, prediction model based on oring operator and ranking equation, and performance evaluation.

### 3.1. *Dataset description*

Six datasets are adopted to evaluate the proposed model (AL-Rousan et al. 2012). The datasets were collected from a real mechanical single axis solar tracking system. Several readings were measured and collected (i.e., month, day, time, short circuit current ($I_{sc}$), open circuit voltage ($V_{oc}$), and power radiation). The target of the used solar tracking system is to find the orientation angle of a solar photovoltaic panel. Month, date and time variables are month of year, date of month and the time to take measurements. Short circuit current is the current measured through the photovoltaic module when the voltage across the module is equal to zero, while open circuit voltage is measured when no external load is connected to the photovoltaic module terminals, therefore, no external electric current flows between terminals. In addition, the power radiation ($P_r$) is measured using pyranometer device to find the amount of the power radiated from the sun on the sea surface.

Table 1 shows the number of variables adopted to use the selected datasets for conventional models and for the proposed models (after applying the proposed model).

### 3.2. *Prediction model based on oring operator and ranking equation*

To improve the conventional MLP and SVR, the proposed model is divided into two main steps. In the first step, the discrete variables and continuous variables are forwarded to logical OR and ranking equation, respectively. This process aims

**Table 1.** Number of variables in each dataset.

| Dataset | Conventional predictor | Proposed predictor (after processing) |
|---|---|---|
| 1 | 6 | 5 |
| 2 | 3 | 4 |
| 3 | 4 | 4 |
| 4 | 2 | 3 |
| 5 | 3 | 4 |
| 6 | 3 | 4 |

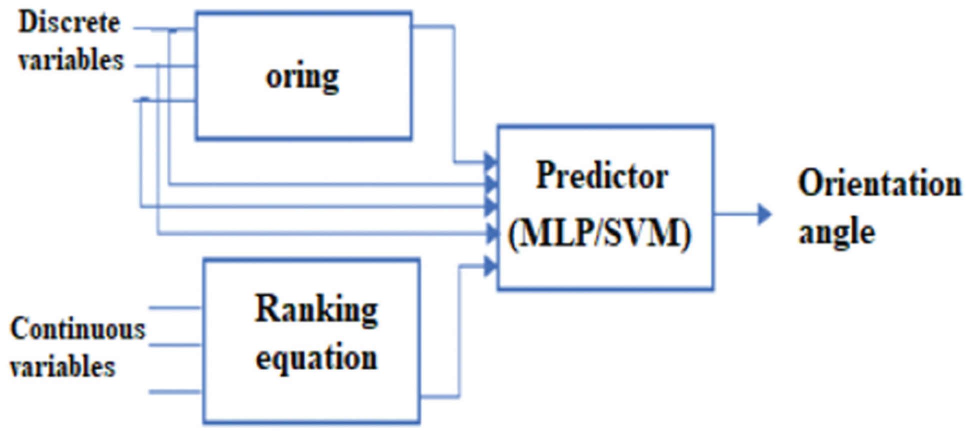


**Figure 3.** Proposed prediction model.

to create new variables that store the behavior of forwarded variables. These new variables will be used to enhance the prediction model. The ranking equation is calculated based on the experimental analysis of the datasets, so each continuous variable is given a unique number.

$$Ranking\ Equation = 100 * I_{sc} + 200 * V_{oc} + 300 * P_r. \quad (5)$$

While discrete variables can be combined using the bit oring as follow:

$$BitOR = Var_1\ OR\ Var_2, \quad (6)$$

where $Var_1$ and $Var_2$ are the forwarding variables, and OR is a logical bit OR operator. Figure 3 shows the prediction model and the interconnections between the components of the model.

As shown in Figure 3, the prediction model will forward both discrete variables and continuous variables to logical OR operator and ranking equation, respectively. The discrete variables will be forwarded with the output of oring and ranking equation to a conventional predictor (i.e. MLP and SVR).

Re-forwarding discrete variables as input is based on experimental analysis of using all data sets with conventional MLP and SVR to predict the orientation angle as targeted output. In addition, the correlation analysis between the generated variables and the orientation angle has supported the step of re-forwarding the discrete variables as inputs for conventional predictors. MLP and SVR are selected based on their performance in different fields and applications (Nielsen 2015; Wu et al. 2010). To validate the proposed MLP-based model, three MLP scenarios are considered as follow:

(i) MLP with one hidden layer.
(ii) MLP with two hidden layers.
(iii) MLP with three hidden layers.

Scenarios are adopted to achieve the best performance of MLP. In addition, several parameters should be tuned to use MLP predictor efficiently. Based on literature review, the optimum parameters to implement MLP predictor are shown in Table 2. On the other hand, the number of neurons in input layer vary depending on the number of variables that are adopted for each dataset separately. In addition, the number of neurons in the hidden layers is calculated based on trial and error. To achieve this target, the largest dataset with 3 hidden layers is used to find the optimum number of neurons as discussed in Section 5.

**Table 2.** Tuned parameters for MLP.

| Parameter | Value |
|---|---|
| Number of epochs | 1000 |
| Mu | 1 x $10^{10}$ |
| Gradient descent value | 1 x $10^{-7}$ |
| Performance | 1 x $10^{-15}$ |
| Total number of parameters | Based on dataset |
| Training ratio | 70% |
| Validating ratio | 15% |
| Testing ratio | 15% |
| Number of neurons (hidden layer/s) | 10 |
| Hidden transfer function | Tansig |
| Output transfer function | Purlin |
| Optimization algorithm | Levenberg-Marquart (LM) |

To create a support vector machine regression (SVR) predictor different kernels are used as follows:

(i) Linear SVR.
(ii) Quadratic SVR.
(iii) Cubic SVR.
(iv) Fine Gaussian SVR.
(v) Medium Gaussian SVR.
(vi) Coarse Gaussian SVR.

Table 3 shows the kernel scale for each type of SVR models.

**Table 3.** Kernel scale for SVR models.

| Type of non-linear SVR | Kernel Scale |
|---|---|
| Quadratic SVR | $K = c$ |
| Cubic SVR | $K = c$ |
| Fine Gaussian SVR | $K = \sqrt{i}/4$ |
| Median Gaussian SVR | $K = \sqrt{i}$ |
| Coarse Gaussian SVR | $K = 4 \times \sqrt{i}$ |

where $K$ is the kernel scale, $i$ is the number of features.

### 3.3. *Performance evaluation*

This section presents the performance criteria that are used to evaluate the research methodology. Two performance criteria, which are used in this study are discussed (i.e., *RMSE*, and $R^2$). Both *RMSE* and $R^2$ are adopted as performance criteria to evaluate the developed MLP and SVR. It is recommended to use RMSE performance metric when the outcomes are numerical variables (Kuhn and Johnson 2019). RMSE should be minimized to optimize the developed models. In contract, $R^2$ should be maximized to perform better. $R^2$ and *RMSE* can be calculated by using Equations (7) and (8) respectively.

$$R^2 = 1 - \frac{\sum_{i=1}^{N} (y_i - \widehat{y}_i)^2}{\sum_{i=1}^{N} (y_i - \bar{y})^2}, \tag{7}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=0}^{N} (y_i - \widehat{y}_i)^2}, \tag{8}$$

where, $y_i$, $\widehat{y}_i$, and $\bar{y}$ are the target of the prediction model, the predicted value, and the mean of the target data, respectively. In addition, $N$ and $i$ are the number of samples in the dataset and the index of the data in the dataset, respectively.

## 4. Experimental results and discussion

To verify the proposed prediction model compared to conventional SVR and MLP predictors, this section presents experimental results based on *RMSE* and $R^2$ values.

### 4.1. *Proposed predictor-based support vector machine regression (SVR)*

To implement the proposed predictor-based SVR model, different kernels are used. These changes on kernels are done to investigate the optimum SVR types that can be used to find the targeted results while maximizing both *RMSE* and $R^2$ performance metrics. The results of conventional Linear, Quadratic, Cubic, Fine, Median, and Coarse Gaussian SVR are shown in Table 4, while the results of the proposed models based SVR are shown in Table 5.

As shown from Table 4, the best $R^2$ and *RMSE* results are found using quadratic kernel where the *RMSE* is 0.151 using dataset 6 and $R^2$ is 0.69 using both datasets 1 and 6. The results of the proposed model show an improvement on conventional SVR results with combination of median Gaussian kernel and dataset 1. *RMSE* and $R^2$ are 0.148 and 0.71, respectively. The results indicated that the proposed model improved 90% of overall cases with improvement ranges from 1.001 to 1.988 and 1.000 and 2.385 for both *RMSE* and $R^2$, respectively.

The results revealed that using the proposed technique with dataset 6 is the most efficient model compared to other models since the number of forwarded inputs is lower than other models. Besides that, it was found that the computational processing is faster. The results proved that using oring operator and ranking equation could enhance the performance of conventional SVR using different kernels.

### 4.2. *Proposed predictor based Multi-Layer Perceptron (MLP)*

To build MLP predictor using different datasets, the predefined parameters are set as shown in Section 4. After assigning the parameters, numbers of neurons are determined experimentally using different datasets to minimize the *RMSE* and maximize $R^2$. To tune the optimum parameters, all the datasets are analyzed and tested using three hidden layers, then the parameters of the highest performance dataset were adopted to implement the proposed predictor. It is found that the best dataset based on *RMSE* and $R^2$ analysis is dataset 6. While the optimum number of neurons are selected by testing dataset 6 with variation in the number of neurons from 1 to 14. Figure 4 shows the results of *RMSE* and $R^2$ based on using different numbers of neurons. The results proved that using 10 neurons performed better for both *RMSE* and $R^2$.

The overall results including training, validation and testing processes for conventional MLP and proposed MLP-based system are shown in Table 6. As shown in Table 6, the optimal $R^2$

**Table 4.** Conventional SVR results.

| | Linear | | Quadratic | | Cubic | | Fine | | Median | | Coarse | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Dataset | $R^2$ | *RMSE* | $R^2$ | *RMSE* | $R^2$ | *RMSE* | $R^2$ | *RMSE* | $R^2$ | *RMSE* | $R^2$ | *RMSE* |
| 1 | 0.62 | 0.169 | 0.69 | 0.154 | 0.65 | 0.163 | 0.26 | 0.239 | 0.68 | 0.156 | 0.64 | 0.171 |
| 2 | 0.62 | 0.169 | 0.63 | 0.167 | 0.59 | 0.177 | 0.57 | 0.179 | 0.62 | 0.171 | 0.63 | 0.170 |
| 3 | 0.63 | 0.167 | 0.55 | 0.185 | 0.43 | 0.208 | 0.37 | 0.218 | 0.6 | 0.174 | 0.62 | 0.170 |
| 4 | 0.60 | 0.175 | 0.67 | 0.159 | 0.63 | 0.166 | 0.63 | 0.168 | 0.63 | 0.168 | 0.61 | 0.172 |
| 5 | 0.59 | 0.175 | 0.54 | 0.185 | 0.48 | 0.199 | 0.41 | 0.210 | 0.57 | 0.179 | 0.58 | 0.178 |
| 6 | 0.60 | 0.174 | 0.69 | 0.151 | 0.67 | 0.157 | 0.67 | 0.158 | 0.67 | 0.157 | 0.62 | 0.170 |

**Table 5.** Proposed model based SVR results.

| | Linear | | Quadratic | | Cubic | | Fine | | Median | | Coarse | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Dataset | $R^2$ | *RMSE* | $R^2$ | *RMSE* | $R^2$ | *RMSE* | $R^2$ | *RMSE* | $R^2$ | *RMSE* | $R^2$ | *RMSE* |
| 1 | 0.70 | 0.152 | 0.70 | 0.150 | 0.63 | 0.166 | 0.62 | 0.169 | 0.71 | 0.148 | 0.63 | 0.168 |
| 2 | 0.65 | 0.165 | 0.68 | 0.157 | 0.64 | 0.166 | 0.48 | 0.200 | 0.65 | 0.163 | 0.64 | 0.169 |
| 3 | 0.64 | 0.165 | 0.68 | 0.156 | 0.65 | 0.163 | 0.48 | 0.198 | 0.63 | 0.167 | 0.63 | 0.169 |
| 4 | 0.60 | 0.173 | 0.67 | 0.158 | 0.65 | 0.163 | 0.58 | 0.180 | 0.64 | 0.166 | 0.62 | 0.168 |
| 5 | 0.60 | 0.175 | 0.59 | 0.178 | 0.58 | 0.177 | 0.44 | 0.206 | 0.61 | 0.1736 | 0.59 | 0.175 |
| 6 | 0.61 | 0.171 | 0.71 | 0.149 | 0.67 | 0.157 | 0.68 | 0.154 | 0.70 | 0.149 | 0.64 | 0.165 |

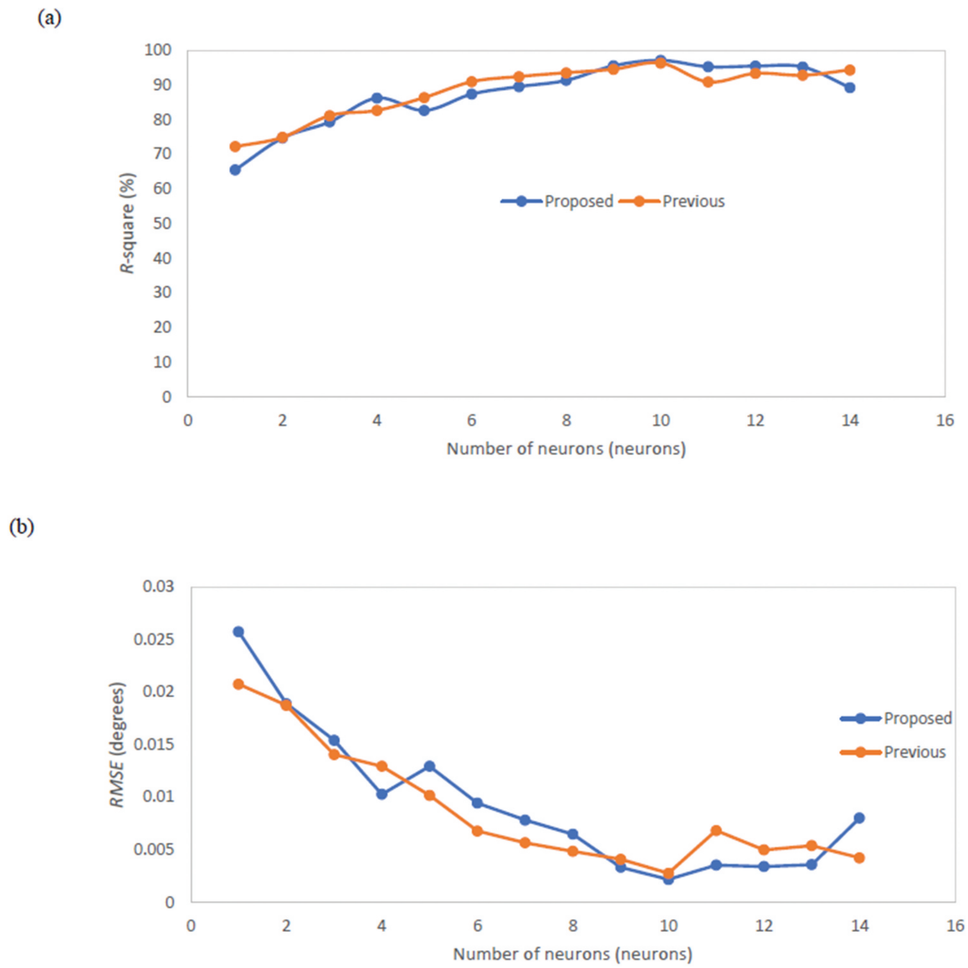


**Figure 4.** Optimum number of neurons in old and proposed model, using (a) $R^2$; (b) RMSE.

**Table 6.** MLP and proposed model based MLP results.

| | MLP | | Modified MLP | | MLP | | Modified MLP | | MLP | | Modified MLP | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | One hidden | | One hidden | | Two hidden | | Two hidden | | Three hidden | | Three hidden | |
| Dataset | *RMSE* | $R^2$ | *RMSE* | $R^2$ | *RMSE* | $R^2$ | *RMSE* | $R^2$ | *RMSE* | $R^2$ | *RMSE* | $R^2$ |
| 1 | 0.085 | 0.9017 | 0.079 | 0.9158 | 0.069 | 0.9356 | 0.061 | 0.9507 | 0.053 | 0.9629 | 0.047 | 0.9709 |
| 2 | 0.121 | 0.7913 | 0.120 | 0.7928 | 0.120 | 0.8080 | 0.118 | 0.8119 | 0.110 | 0.8382 | 0.115 | 0.8226 |
| 3 | 0.125 | 0.7910 | 0.122 | 0.7992 | 0.120 | 0.8044 | 0.121 | 0.8055 | 0.122 | 0.8008 | 0.121 | 0.8042 |
| 4 | 0.124 | 0.7940 | 0.123 | 0.7966 | 0.126 | 0.7876 | 0.125 | 0.7907 | 0.123 | 0.7973 | 0.122 | 0.7989 |
| 5 | 0.132 | 0.7688 | 0.126 | 0.7873 | 0.126 | 0.7885 | 0.123 | 0.7959 | 0.125 | 0.7892 | 0.123 | 0.7971 |
| 6 | 0.077 | 0.9208 | 0.063 | 0.9458 | 0.055 | 0.9594 | 0.048 | 0.9686 | 0.052 | 0.9633 | 0.042 | 0.9755 |

and *RMSE* are achieved using three hidden layers with dataset 6. *RMSE* and $R^2$ values of the proposed model are 0.042 degrees and 97.55%, respectively, while *RMSE* and $R^2$ of the conventional model are 0.052 degrees and 96.33%, respectively. The results revealed that the proposed model could improve 93% of the overall cases compared to conventional model. In addition, the improvement ranges are varying from 1.0013 to 1.4613 degrees and from 1.0019 to 1.4984 times for *RMSE* and $R^2$ respectively. The results revealed that using the proposed model with dataset 6 and three hidden layers is more efficient compared to other models. This would prove that using oring operator and ranking equation can enhance the performance of MLP using different hidden layers.

### 4.3. *Analysis and capability of the proposed categorical linear predictor*

To create a solar tracking system, researchers have used different continuous and categorical variables to estimate the best angles that could collect the highest power radiation using photovoltaics cells. In this study, combined variables using logical OR (for categorical) and ranking equation (for continuous variables) are used with one of the prediction models including MLP and SVR The created models have the same time complexity as the convolutional prediction models with extra memory and time in processing the ranking and bit oring equations. To eliminate the effect of ranking and bit oring equations, a preprocessing method is suggested. Preprocessing step is considered to combine the inputs together to save time and memory before using conventional methods. The results revealed that hybrid combination between bit oring and conventional model achieved better performance in term of prediction and error, while the process time is a little longer than conventional model. To sum up all the results, this study concludes that using an additional variable will improve the prediction rate and error function of the proposed models.

## 5. Conclusion

This article wishes to improve the conventional SVR and MLP predictors based on oring operator and ranking equation. This integration aims to increase the accuracy of the learning system, minimize the *RMSE*, maximize the $R^2$, and increase the capability and the robustness of the conventional models. The results proved that the proposed predictors performed better compared to conventional predictors. The average improvements of proposed based MLP predictor are 1.047 degrees for *RMSE*, and 1.083 times for $R^2$, while using SVR predictor averagely improved 1.103 degrees for *RMSE* and 1.094 times for $R^2$. The results revealed that integrating the input variables in one variable can improve the performance of learning system with less overhead in real environment.

## Nomenclature

| | |
|---|---|
| $I_{sc}$ | Short Circuit Current |
| MLP | Multi-Layer Perceptron |
| $P_r$ | Power Radiation |
| $R^2$ | Coefficient of Determination |
| *RMSE* | Root Mean Square Error |
| SVR | Support Vector Machine Regression |
| $V_{oc}$ | Open Crcuit Voltage |

## Disclosure statement

No potential conflict of interest was reported by the authors.

## ORCID

Nadia Al-Rousan http://orcid.org/0000-0001-8451-898X
Hazem Al-Najjar http://orcid.org/0000-0002-6143-2734

## References

Abdualrhman, M. A. A., and M. C. Padma. 2019. “CS-IBC: Cuckoo Search Based Incremental Binary Classifier for Data Streams.” *Journal of King Saud University-Computer and Information Sciences* 31 (3): 367–377. doi:10.1016/j.jksuci.2017.05.008.

Al-Najjar, H., S. S. N. Alhady, and J. M. Saleh. 2019. “Improving a Run Time Job Prediction Model for Distributed Computing Based on Two Level Predictions.” In *10th International Conference on Robotics, Vision, Signal Processing and Power Applications*, Penang, Malaysia, 14-15 August 2018: 35–41. Singapore: Springer.

Al-Najjar, H. M., and S. S. N. A. S. Hassan. 2016. “A Survey of Job Scheduling Algorithms in Distributed Environment.” In *2016 6th IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, Batu Ferringhi, Malaysia, 25-27 November 2016: 39–44. New York: IEEE.

AL-Rousan, N., M. Al-Rousan, A. Shareiah, and H. AL-Najjar. 2012. “Choosing the Efficient Tracking Method for Real Time Tracking System in Jordan and It’s Neighbors to Get Maximum Gained Power Based on Experimental Data.” In *2012 International Conference on Renewable Energy Research and Applications (ICRERA)*, Nagasaki, Japan, 11-14 November 2012: 1–6. New York: IEEE.

AL-Rousan, N., N. A. M. Isa, and M. K. M. Desa. 2018. “Advances in Solar Photovoltaic Tracking Systems: A Review.” *Renewable and Sustainable Energy Reviews* 82 (3): 2548–2569. doi:10.1016/j.rser.2017.09.077.

AL-Rousan, N., N. A. M. Isa, and M. K. M. Desa. 2020. “Efficient Single and Dual Axis Solar Tracking System Controllers Based on Adaptive Neural Fuzzy Inference System.” *Journal of King Saud University - Engineering Sciences* 32 (7): 459–469. doi:10.1016/j.jksues.2020.04.004.

Cerri, R., R. C. Barros, and A. C. P. L. F. de Carvalho. 2014. “Hierarchical Multi-Label Classification Using Local Neural Networks.” *Journal of Computer and System Sciences* 80 (1): 39–56. doi:10.1016/j.jcss.2013.03.007.

Chen, M. 2019. “Improved Artificial Bee Colony Algorithm Based on Escaped Foraging Strategy.” *Journal of the Chinese Institute of Engineers* 42 (6): 516–524. doi:10.1080/02533839.2019.1611479.

Dai, H., D. Du, X. Li, and Y. Song. 2019. “Improving Fine-Grained Entity Typing with Entity Linking.” In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, Hong Kong, China, 3–7 November 2019: 6210–6215. China: Association for Computational Linguistics.

Deb, G., and A. B. Roy. 2012. “Use of Solar Tracking System for Extracting Solar Energy.” *International Journal of Computer and Electrical Engineering* 4 (1): 42–46. doi:10.7763/IJCEE.2012.V4.449.

Desa, M. K. M., S. Sapeai, A. W. Azhari, K. Sopian, M. Y. Sulaiman, N. Amin, and S. H. Zaidi. 2016. “Silicon Back Contact Solar Cell Configuration: A Pathway Towards Higher Efficiency.” *Renewable and Sustainable Energy Reviews* 60: 1516–1532. doi:10.1016/j.rser.2016.03.004.

Drucker, H., C. J. C. Burges, L. Kaufman, A. J. Smola, and V. Vapnik. 1997. “Support Vector Regression Machines.” In *Advances in Neural Information Processing Systems* 9: 155–161.

Haykin, S. 2009. *Neural Networks and Learning Machine*. 3rd ed. USA: Pearson.

Htwe, T. T., and N. S. M. Kham. 2019. “Improving Accuracy of IDS Using Genetic Algorithm and Multilayer Perceptron Network.” In *International Conference on Innovative Computing and Communications*, New Delhi, India, 5-6 May 2018: 313–321. Singapore: Springer.

Jaouedi, N., N. Boujnah, and M. S. Bouhlel. 2019. “A New Hybrid Deep Learning Model for Human Action Recognition.” *Journal of King Saud University-Computer and Information Sciences* 32 (4): 447–453. doi:10.1016/j.jksuci.2019.09.004.

Juswanto, W., and Z. Ali. 2016. “Renewable Energy and Sustainable Development in Pacific Island Countries.” *Asian Development Bank Institute* 2016-5: 1–6.

Kordos, M., and A. Rusiecki. 2013. “Improving MLP Neural Network Performance by Noise Reduction.” *Theory and Practice of Natural Computing* 8273: 133–144. doi:10.1007/978-3-642-45008-2_11.

Koutsoukas, A., K. J. Monaghan, X. Li, and J. Huan. 2017. “Deep-Learning: Investigating Deep Neural Networks Hyper-Parameters and Comparison of Performance to Shallow Methods for Modeling Bioactivity Data.” *Journal of Cheminformatics* 9 (1): 42–56. doi:10.1186/s13321-017-0226-y.

Kuhn, M., and K. Johnson. 2019. *Feature Engineering and Selection: A Practical Approach for Predictive Models*. USA: CRC Press.

Landwehr, N., M. Hall, and E. Frank. 2005. “Logistic Model Trees.” *Machine Learning* 59 (1–2): 161–205. doi:10.1007/s10994-005-0466-3.

Mata-Moya, D., N. del-Rey-Maestre, V. M. Peláez-Sánchez, M.-P. Jarabo-Amores, and J. Martín-de-Nicolás. 2015. “MLP-CFAR for Icoherent Radar Detectors Robustness in Variable Scenarios.” *Expert Systems with Applications* 42 (11): 4878–4891. doi:10.1016/j.eswa.2014.12.055.

Mizutani, E., and S. E. Dreyfus. 2001. “On Complexity Analysis of Supervised MLP-Learning for Algorithmic Comparisons.” *International Joint Conference on Neural Networks*, Washington, DC, USA, 15-19 July 2001: 347–352. New York: IEEE.

Mosavi, A., P. Ozturk, and K.-W. Chau. 2018. “Flood Prediction Using Machine Learning Models: Literature Review.” *Water* 10 (11): 1536-1-1536-40. doi:10.3390/w10111536.

Nielsen, M. 2015. *Neural Networks and Deep Learning*. USA: Determination Press.

Osowski, S., K. Siwek, and T. Markiewicz. 2004. "MLP and SVM Networks - a Comparative Study." In *Proceedings of the 6th Nordic Signal Processing Symposium*, Espoo, Finland, 11 June 2004: 37–40. New York: IEEE.

Panigrahi, S., and H. S. Behera. 2020. "A Study on Leading Machine Learning Techniques for High Order Fuzzy Time Series Forecasting." *Engineering Applications of Artificial Intelligence* 87: 103245-1-103245-10. doi:10.1016/j.engappai.2019.103245.

Qi, W., X. Luo, X. Liu, Y. Liu, and Y. Yu. 2019. "Product Optimization with the Improved Marginal Moment Model." *Journal of the Chinese Institute of Engineers* 42 (5): 449–459. doi:10.1080/02533839.2019.1598283.

Randall, T. 2016. "Wind and Solar are Crushing Fossil Fuels." Bloomberg. Accessed April 6. https://www.bloomberg.com/news/articles/2016-04-06/wind-and-solar-are-crushing-fossil-fuels.

Shao, Y., H. Ou, P. Guo, and H. Yang. 2019. "Shape Optimization of Preform Tools in Forging of Aerofoil Using a Metamodel-Assisted Multi-Island Genetic Algorithm." *Journal of the Chinese Institute of Engineers* 42 (4): 297–308. doi:10.1080/02533839.2019.1584734.

Srikumar, K., and C. Saibabu. 2020. "A System and Novel Methodology to Track Maximum Power from Photo Voltaic System: A Comparative and Experimental Analysis." *Journal of King Saud University-Engineering Sciences* 32 (7): 442–458. doi:10.1016/j.jksues.2018.02.006.

Stowe, K., J. Anderson, M. Palmer, L. Palen, and K. Anderson. 2018. "Improving Classification of Twitter Behavior During Hurricane Events." In *Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media*, Melbourne, Australia, 20 July 2018: 67–75. Australia: Association for Computational Linguistics.

Talreja, V., F. Taherkhani, M. C. Valenti, and N. M. Nasrabadi. 2018. "Using Deep Cross Modal Hashing and Error Correcting Codes for Improving the Efficiency of Attribute Guided Facial Image Retrieval." In *2018 IEEE Global Conference on Signal and Information Processing*, Anaheim, CA, USA, 26-29 November 2018: 564–568. New York: IEEE.

Tan, J. 2020. "Complex Object Detection Using Deep Proposal Mechanism." *Engineering Applications of Artificial Intelligence* 87: 103234-1-103234-6. doi:10.1016/j.engappai.2019.09.003.

Üstün, B., W. J. Melssen, and L. M. C. Buydens. 2006. "Facilitating the Application of Support Vector Regression by Using a Universal Pearson VII Function Based Kernel." *Chemometrics and Intelligent Laboratory Systems* 81 (1): 29–40. doi:10.1016/j.chemolab.2005.09.003.

Wu, C., X. Lv, X. Cao, Y. Mo, and C. Chen. 2010. "Application of Support Vector Regression to Predict Metallogenic Favourability Degree." *International Journal of Physical Sciences* 5 (16): 2523–2527. doi:10.5897/IJPS.9000080.

Wu, C. H., G. H. Tzeng, and R. H. Lin. 2009. "A Novel Hybrid Genetic Algorithm for Kernel Function and Parameter Optimization in Support Vector Regression." *Expert Systems with Applications* 36 (3): 4725–4735. doi:10.1016/j.eswa.2008.06.046.

Zhang, C., P. Patras, and H. Haddadi. 2019. "Deep Learning in Mobile and Wireless Networking: A Survey." *IEEE Communications Surveys and Tutorials* 21 (3): 2224–2287. doi:10.1109/COMST.2019.2904897.

Zhang, Q. J., and K. C. Gupta. 2000. *Neural Networks for RF and Microwave Design*. USA: Artech House.