ESKİŞEHİR TECHNICAL UNIVERSITY JOURNAL OF SCIENCE AND TECHNOLOGY
A- APPLIED SCIENCES AND ENGINEERING

2020, 21(2), pp. 294 - 313, DOI: 10.18038/estubtda.650497

HOURLY GLOBAL SOLAR RADIATION ESTIMATION BASED ON MACHINE LEARNING METHODS IN ESKİŞEHİR

Massa ALSAFADI *, Ümmühan BAŞARAN FİLİK

Electrical & Electronics Engineering Department, Engineering Faculty, Eskişehir Technical University, Eskişehir, Turkey

ABSTRACT

Due to the increasing importance of knowing the amount of global solar radiation (GSR) that is incident on solar panels; short term data, such as hourly global solar radiation (HGSR), is essentially required to obtain more accurate and reliable power generation prediction. Nowadays, Machine Learning (ML) methods are becoming a huge trend for data forecasting. Therefore, in this paper, a comparison between Collares-Pereira & Rabl empirical model modified by Gueymard (CPRG) and ML methods for HGSR estimation in Eskişehir city in Turkey is conducted. Artificial Neural Network (ANN), Regression Tree (RT), and Support Vector Regression (SVR) are ML methods that are used to predict HGSR. Besides, hourly metrological and geographical parameters for the year 2014 are taken as inputs in the training models. The inputs are solar time, solar hour angle, Julian day number, daily GSR, longitude, latitude, hourly average humidity, hourly temperature, and hourly pressure. To demonstrate these techniques, a comparison is implemented using MATLAB software with the help of existing toolboxes. Finally, this study proves that ML methods outperform the CPRG model, not to mention they have far more accurate results. Although almost all ML models gave similar results, SVR was the best among them with a correlation coefficient of 0.979532 for the training set and 0.978244 for the testing set. In a nutshell, ML are very efficient methods in that should be taken into consideration to perfectly estimate HGSR.

Keywords: Hourly global solar radiation, Machine learning, Artificial neural network, Regression tree, Support vector regression

1. INTRODUCTION

Traditional energy sources, like fossil fuels and gas, have been recently taking a major part in producing electrical energy for many years. However, they are considered to be the main source of the earth's pollution and global warming, they are also going to extinct one day soon [1]. In addition, the increase in the world's population with their demand for electricity can cause limitations in traditional energy production. Therefore, modern and more efficient methods had to be found to produce electricity instead of the, previously mentioned, conventional sources [2]. Thereby, renewable energy has started to take the lead in this field in the last decades. Among several renewable energy resources, solar energy has gained a huge attraction as a potential substitute for electrical energy because of its availability, cost efficiency, and environmental friendliness [3]. Nowadays, electrical engineers are trying to achieve an integration between conventional and renewable energy into largely connected grids or only by using renewable energy recourses in isolated regions in the next upcoming years. In that case, a balance between energy production and consumption must be created to maintain accurate and controllable demand usage of power. Furthermore, The intermittent nature of solar power can cause some problems that lead to a lack of electrical management. Thus, efficient methods have to be followed to keep that balance by finding ways to precisely estimate the most fundamental feature in solar energy, global solar radiation (GSR), to design solar panels that can produce electrical energy to reach specific goals [4]. For GSR prediction, physical empirical methods are basically used for a while until now, not only by introducing regression models but also by using some metrological and geographical parameters like cloud index, solar angle, solar hour, latitude longitude, and…etc. Since technology is getting developed and improved day by day, artificial intelligence (AI) is now taking a huge part in most electronic devices. Machine learning (ML) is part of AI used to predict and classify different features in engineering,

*Corresponding Author: massasafadi@hotmail.com
Received: 25.11.2019 Published: 15.06.2020

medicine, agriculture, and other fields. To summarize, empirical models and ML methods are found by researches to be effective methodologies for estimating solar radiation for the last 15 years [5].

ML has recently been gaining wide popularity because of its goal to make machines and electronic applications learn similar to the human brains and biological neural systems. Therefore, in the field of solar radiation prediction, optimum solar radiation estimation can be achieved by using this heuristic approach. Basically, GSR is spread all around the earth's surface, but due to the difficulty of establishing metrological stations in every location on earth, other solutions must be found to replace expensive measurement devices. Thereby, ML methodologies are taken into consideration due to their accurate outputs compared to classical methods. Moreover, most of the studies that are done by using ML methods, which are proposed in scientific journals, are for monthly and daily GSR [2]. However, for more precise GSR prediction in the short time interval that leads to better performance in transferring solar energy to electrical one and for more efficient solar systems design, hourly global solar radiation (HGSR) can be forecasted instead of daily and monthly. Thus, this paper investigates HGSR estimation using popular ML methods which are Artificial Neural Network (ANN), Regression Tree (RT), and Support Vector Regression (SVR).

ANN's are the most common, used lately, ML method in many fields. They are becoming widely used to predict solar radiation in several studies, giving accurate values results compared to classical methods with minimum errors. Optimum solar radiation estimation can be achieved by using this heuristic approach, and it's proved that that 79% of ML methods used in weather forecasting problems are based on ANN, especially with the absence of metrological stations in many locations of the world. Multi-layer Perceptron Neural Networks (MLPNN) is one of the most widespread methods for feature learning using the backpropagation technique in the ANN field which is conducted in this study. Most of the studies, that involved solar radiation prediction, are done by using MLPNN to predict daily or monthly GSR [4]. Potential for solar energy was predicted by Sözen and Arcaklioğlu in Turkey using ANN. As for, geographical parameters, mean sunshine duration, mean temperature and month number are taken as inputs to the network which gave good results [6]. Elminir also used MLPNN for global solar radiation estimation Helwan, Egypt control stations. To gain the estimation output, the input data were cloudiness, ambient temperature, wind speed, wind direction, relative humidity, and water vapour [7]. In addition, Karoro and other authors proposed in their study an ANN to predict daily global solar radiation in a matter of monthly average values on a horizontal surface in Kampala, Uganda depending on a single feature which is the sunshine duration. Applying 65 neurons with tansig transfer function gave the best results in this study [8]. When it comes to hourly solar radiation, it's rare to find HGSR prediction studies. However, a study was conducted in Amman, Jordan using Feedforward, Elman, and Nonlinear Autoregressive Exogenous (NARX) ANN's on 10 years data while the last gave the best performance [9].

Another popular ML method, that is known for its high efficiency and is used in different engineering, medical and environmental problems; is Regression Tree (RT). Unfortunately, it is rarely used in GSR prediction [10]. However, a feature selection problem for GSR estimation in Tokyo, Japan by using RT; is an example of a paper written by Mori and Takahashi. The research showed that GSR, sunshine duration, and sun altitude are important features that ensure the authors' selection to forecast GSR [11]. On the other hand, RT has been proved to work well in other fields in renewable energy. Troncoso and Salcedo-Sanz proved that hourly short time wind speed prediction in Spain using RT worked well by measuring it in several wind towers. Compared to support vector machines, multilayer perceptron, CART, multilinear regression, and other ML methods, RT outperformed them all [10].

Other than ANN and RT, Support Vector Regression (SVR) has recently been a great attraction in forecasting problems because of its feasibility. Furthermore, it was indicated in several studies that SVR has better accuracy than ANN and regression statistical models in different prediction problems [12]. Despite the fact that there are several engineering applications to predict variables using the SVR

method, just a few studies have estimated GSR by using it [13]. An example for a research paper related to GSR estimation using SVR was introduced by Ramedani and Omid who made a comparison between fuzzy linear regression method and SVR to predict daily GSR in Tehran, Iran by taking daylight hours, Julian day, minimum, maximum and the actual duration of sunshine, clear-sky solar radiation and extraterrestrial radiation as inputs. The study also clarifies that SVR has better performance than fuzzy linear regression [14]. Another research study was written by Mohammadi and Shamshirband using daily and monthly long-term measured GSR for 11 years in the city of Isfahan in Iran was achieved by only using sunshine hours as input [13]. Moreover, Chen and Li made a comparison between Angstrom-Prescott empirical models and SVR to estimate daily GSR in several cities in China by taking inputs of sunshine duration combinations for using SVR. There is no doubt that the study proved that the SVR method gave more descent results than the empirical models [12]. Lately, enhanced and developed models for SVR were found to predict HGSR more accurately by using penalized SVR and forward regression on a quadratic kernel SVM that were introduced by Jiang and Dong in [15].

This current study proposes HGSR estimation in the city of Eskişehir, Turkey by comparing an existing well-chosen empirical model known as Collares-Pereira & Rabl modified by Gueymard referred to as (CPRG) with decent ML methods like ANN, RT, and SVR. The paper clarifies a literature explanation of empirical models and a detailed demonstration of the CPRG model. Moreover, the previously mentioned ML methods are presented with their methodologies. Finally, a MATLAB program is applied to output the estimation results and compare them with each other using these methods with the discussion of these results.

2. METHODOLOGY

Before starting to estimate HGSR using ML methods, a background about empirical regression models and a popular model which is CPRG is introduced in the following section in order to evaluate the accuracy and compare between the classical and modern intelligent estimation methods. In addition, a literature review of three different types of ML methods is also presented.

2.1. Empirical Models

As solar radiation is an important element for many solar applications like generating power and thermal uses; and since many locations around the world don't have any pyranometer or pyrheliometers or any solar radiation measuring devices; alternative ways of prediction had to be found. Therefore, empirical models are essential for estimating HGSR; which are relationships between physical, astronomical, geographical, and meteorological parameters that are taken from a location and correlated to give HGSR from daily solar radiation [16]. However, since most of the measurements around the world are daily, decomposition models from daily to hourly data are necessary for the long-term forecasting, because measurement devices are short time predicted [17].

Nevertheless, there are several empirical models for HGSR estimation mentioned in a number of research articles and works of literature that are divided into three groups depending on their inputs. The first group has inputs that are related to time such as solar hour angle, day length, and solar time, and not related to atmospheric changes [18]. Examples of these models are Whillier model [17], Liu & Jordan model [19], Collares-Pereira & Rabl (CPR) model [20], CPRG model which is the CPR modified by Gueymard [21] and Garg & Garg model [22]. All these previous models are modified and extracted from each other. The second group depends on atmospheric changes and climate variation and has hourly values distributed in a normal Gaussian function [18]. Such as Jain model 1 [23], Jain model 2 [24] , and Shazly model [25]. The third group of HGSR estimation models neither depends on time change nor the randomness of climate change like Newell's model [26].

2.1.1. CPRG model

Since this study is basically depending on ML methods, only an existing empirical model will be chosen to be evaluated and compared with the other ML techniques. CPRG model is considered to be the most common model in HGSR estimation. As described in other researches, by using decomposition algorithms, it's proved that CPRG gave the best accurate results compared to the rest of the empirical models [18]. In addition, a study that was previously done in Eskişehir proved that CPRG worked the best; since it had the least number of errors among other models as mentioned in [27]. The model was modified from several models which were also modified by the first HGSR model that was found by Whillier in 1956 [17].

The CPRG model is defined by the following equation

$$\frac{I}{H} = \frac{(a + b\cos W)r_0}{f_c} \tag{1}$$

While $\frac{I}{H}$ is the hourly to daily GSR ratio and the variables a and b are regression coefficients

$$a = 0.409 + 0.5016\sin(W_s - 60^{\circ}) \tag{2}$$

$$b = 0.6609 - 0.4767\sin(W_s - 60^{\circ}) \tag{3}$$

While r_0 and f_c are calculated as

$$f_c = a + 0.5\,b\,\frac{\frac{\pi W_s}{180} - \sin W_s\,\cos W_s}{\sin W_s - \frac{\pi W_s}{180}\cos W_s} \tag{4}$$

$$r_0 = \frac{\pi}{24}.\frac{\cos W - \cos W_s}{\sin W_s - \frac{\pi W_s}{180}\cos W_s} \tag{5}$$

While W_s and W are the sunset hour angle and solar hour angle respectively taken in degrees defined as

$$W_s = \cos^{-1}(-\tan\varphi\tan\delta) \tag{6}$$

$$W = \frac{360(t_s - 12)}{24} \tag{7}$$

Where δ, φ and t_s are the declination angle, the latitude in degrees, and the solar time in hours that can be calculated from this equation [28].

$$t_s = LT + \frac{ET}{60} + \frac{4}{60}(L_s - L_l) \tag{8}$$

LT is the local standard time taken from the clock, L_s is the standard meridian for a local zone, L_l is the longitude, ET is the equation of time given as

$$ET = 9.87 \sin 2B - 7.53 \cos B - 1.5 \cos B \tag{9}$$

$$B = \frac{360(d-81)}{365} \tag{10}$$

In addition, the declination angle is known as

$$\begin{aligned}\delta = (\frac{180}{\pi})(&0.006918 - 0.399912 \cos \Gamma \\ &+ 0.070257 \sin \Gamma - 0.006758 \cos 2\Gamma \\ &+ 0.000907 \sin 2\Gamma - 0.002697 \cos 3\Gamma \\ &+ 0.00148 \sin 3\Gamma)\end{aligned} \tag{11}$$

$$\Gamma = \frac{2\pi(d-1)}{365} \tag{12}$$

Where d is the day number starting 1 for 1st of January Γ is the day angle in radians.

2.2. Machine Learning (ML)

After explaining HGSR estimation using CPRG empirical model that is considered to be a classical method for estimating solar radiation, a smarter way has been found to estimate monthly, daily and hourly global solar radiation in the last three decades. ML methods have been proved in several scientific papers that they can give good results and accuracy in estimating HGSR better than conventional methods [29]. ML is an essential part of AI that is similar to human learning ability but with a computer instead of the human brain. It also can solve unsolvable problems by using computational and statistical algorithms and teaches computers to learn from the received, huge amount of data. Moreover, it's a method that can represent difficult problems that can't use normal algorithms and with no clear relations between its variables. However, inputs and targets are required for these methods to be trained in a specific map; which depends on the model to give a predicted output, then compared with the true target and minimize the loss function between the target and the predicted output; until the best model is gained. ML is used in several real-life cases such as classifications, pattern recognition, spam filtering, time series prediction, and forecasting. Moreover, there are several types of ML such as supervised learning, unsupervised learning, reinforcement learning, and evolutionary learning. Figure 1 shows the most popular two types that are commonly used nowadays which are supervised and unsupervised learning [4].

2.2.1. Artificial neural network (ANN)

ANN is part of artificial intelligence that consists of an algorithm that allows machines to mimic human beings' neural system and brain by supervised or unsupervised learning. Generally, a neural network consists of neurons that are elements connected between each other by weights and biases. An ANN has an input layer that receives inputs, hidden layers, and an output layer that gives predicted values after mapping the inputs in nonlinear transfer functions. There are several types and methods to perform training in ANN such as Generalized Regression Neural Networks, Cascade-Forward Backpropagation Networks, Elman Back Propagation Networks, recurrent neural networks, and Time Series Neural Networks [30, 31].

However, the most common one used in solar radiation forecasting problems is the multi-layer perceptron neural network (MLPNN) [2]. MLPNN, as seen in Figure 2, is a kind of neural network method that is used for supervised learning and uses the feedforward and backpropagation methods to train the network. It consists of an input layer, several hidden layers, and an output layer. It's generally used for big datasets and real models to satisfy the purpose of using it in real life. MLPNN works by giving the network a dataset of input and output, calculating the predicted outputs from the output layer, and then back-propagating the calculated error, which is the difference between the desired and predicted output, using gradient descent to update the weights. After that, the perfect weights would be found and finally tested on an unseen test set to evaluate the final model. ANN can learn any function that applies to input data, by doing a generalization of the datasets which are trained with, then testing this model on other datasets to see whether it fits well or not by comparing the errors [32].

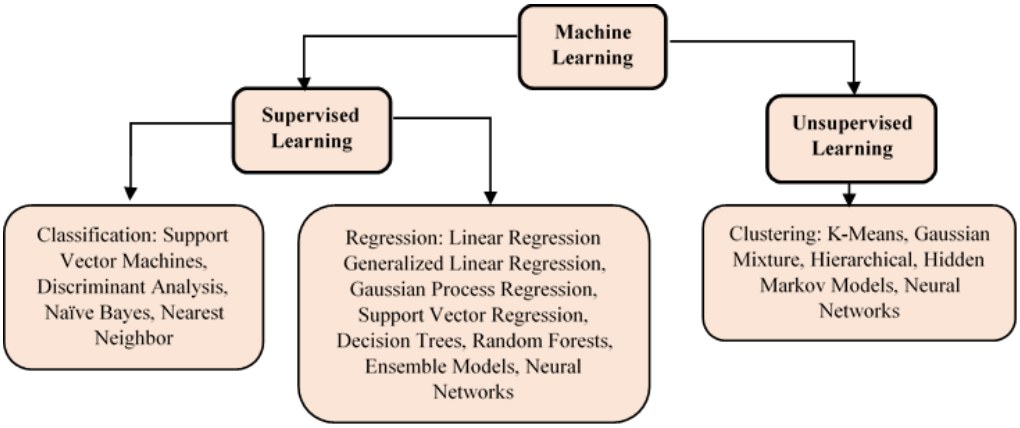


Figure 1. Machine learning techniques

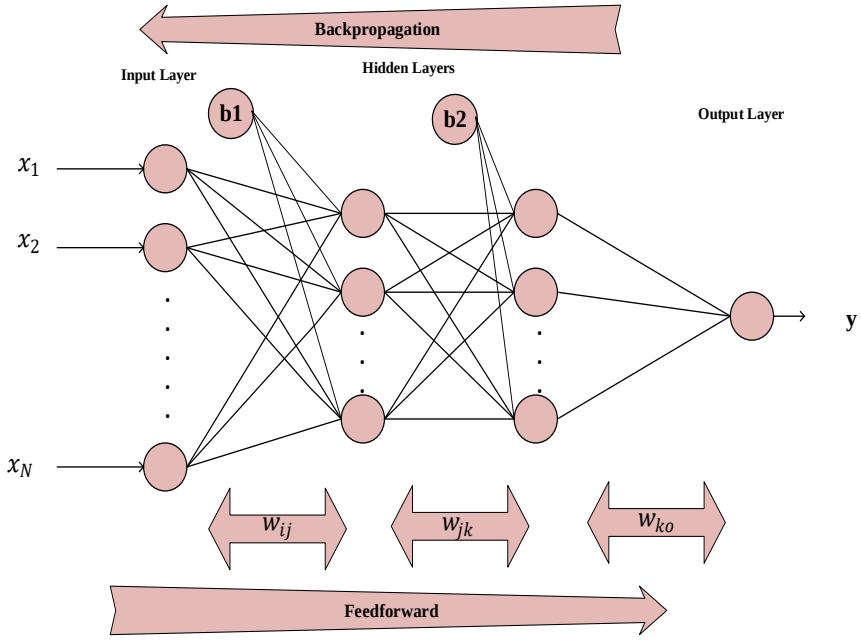


Figure 2. MLPNN diagram

Assuming to have one hidden layer MLPNN, the main equation for the output can be defined by the following function

$$f(x) = G\left(b_2 + w^{(2)}\left(S\left(b_1 + w^{(1)}x\right)\right)\right) \tag{13}$$

While G and S are transfer functions like tansig, pureline, or logsig, the b's are the biases vectors, x is the input matrix and the W are the weight matrices. Examples for most used transfer functions are shown below

$$Logsig(x) = \frac{1}{1 - e^{-x}} \tag{14}$$

$$Tansig(x) = \frac{2}{1 + e^{-2x}} - 1 \tag{15}$$

$$Pureline(x) = x \tag{16}$$

The main steps for the backpropagation algorithm start with randomly initializing the weights between the layers with small values. Then calculating the hypothesis by multiplying the weights with the inputs and adding the bias for the hidden layer as shown in equation (17). This hypothesis for each neuron in the hidden layer is set into a transfer function, after that, the output of the hidden layer is also set into another transfer function to finally acquire the output y in equation (18) as shown below

$$g_j(x) = S\left(\sum_{i=0}^{N} w_{ij}x_i + b_1\right) \tag{17}$$

$$y = f(x) = G\left(\sum_{j=0}^{M} w_j g_j(x) + b_2\right) \tag{18}$$

While N is the number of input nodes, M is the number of neurons in the hidden layer; Forward propagating from layer to layer until the predicted values are obtained. The cost function calculates the error between the desired and predicted outputs

$$E(w) = \frac{1}{2}\sum_{i=0}^{N}(d_i - f_i(x))^2 \tag{19}$$

The error is propagated backwards from output to input while updating the weights as follows

$$w_{current} = w_{current} - \alpha\nabla E(w_{current}) \tag{20}$$

Where α is the learning rate, usually a small decimal number, and $w_{current}$ is the current weight that will be updated. This procedure is repeated until a minimum error is obtained. In order to evaluate the model and select the best one, validation and test sets are tested to the model to set the generalization [32].

2.2.2. Regression tree (RT)

Most of the modern heuristic approaches, which are used in modern ML methods, are inspired by nature life like neural networks, genetic algorithms, random forests …etc. Decision trees are from easy and well-known supervised ML methods that are used in classification and regression problems. Classification and Regression Trees (CART) are one of the most popular algorithms in decision trees. A decision tree model is similar to an upside-down real-life tree which starts with a root node, representing the most effective feature to predict an output, continuing in a greedy way to decision nodes that give binary if and else decisions for a specific feature's condition, ending up with leaf nodes that are the predicted outputs. Figure 3 below shows the upper node ($x_1<t_1$) as the root node, the following nodes are the decision ones and R_1, R_2, R_3, R_4 are the leaf nodes. The difference between classification and regression trees is that the first one predicts categorical attributes, whereas the second predicts continuous values, therefore in this study RT method is selected to predict HGSR. Regarding their models, RT is fast, easy to understand and imagine rather than other ML methods that aren't understandable, because most of their processes are executed behind the scenes. RT is also efficient in dealing with missing data, as a decision node can be skipped to continue without negatively affecting the process [4, 32].

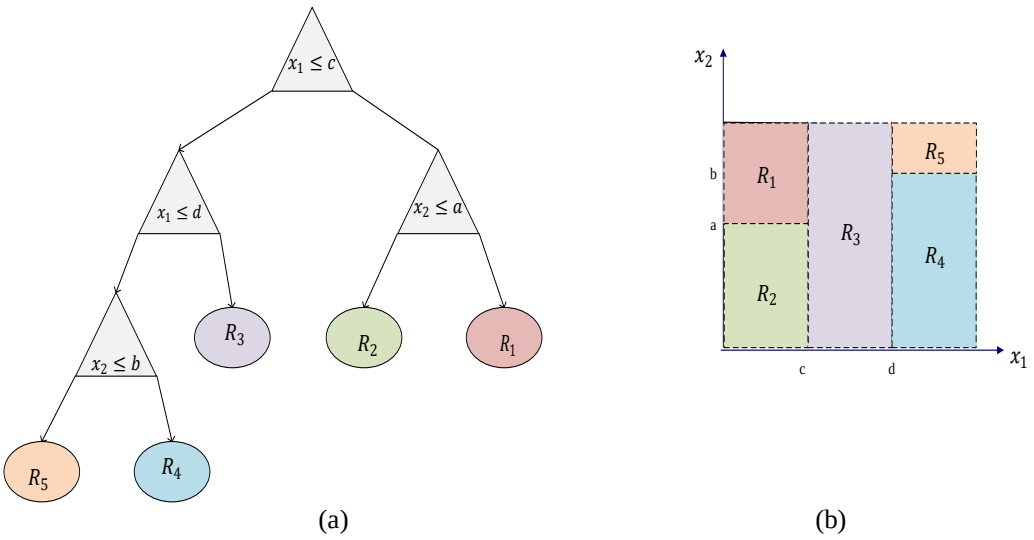


Figure 3. A regression tree: (a) diagram; (b) with its corresponding regression regions.

The working mechanism of RT starts by taking the training dataset and splitting it into smaller subsets to make simple models for each one. Each feature in this subset is checked whether it has the most informative contribution or not and depending on the result it will be chosen to be the root node. Then, a condition is asked, if the answer is yes or no the tree will be split into one of the branches to construct a subtree with decision nodes. By asking binary questions about a single feature, a branch is chosen greedily by not looking forward and just taking the best split in the following branch. In the end, a leaf node, which is the final predicted output, is reached. These procedures are repeated for each subset by using the recursive partition. In other words, instead of applying a linear model to the whole dataset, nonlinear simple models, with more than one attribute for smaller sets, are applied to make it much easier to deal with. Furthermore, each leaf node contains a region of several data points that has an average value m_c and by limiting the sum of squared error into a specific small value, stopping criteria for building a RT can be obtained as shown below

$$m_c = \frac{1}{c}\sum_{i=1}^{c} y_i \tag{21}$$

$$S = \sum_{c \in leaves(T)} \sum_{i \in c} (y_i - m_c)^2 \tag{22}$$

While y is the true output response, c is the number of data points included in a leaf node and T is the number of final leaf nodes. To prevent overfitting on unseen data as in any ML regression problem, tree pruning can be done by minimizing the following complexity cost

$$minimize\ \{S + \alpha|T|\} \tag{23}$$

While α is the complexity parameter that can be chosen by using cross-validation. Thereby, a good model is built that can be tested on a similar regression problem [33, 34].

2.2.3. Support vector regression (SVR)

Support Vector Machine (SVM), is a popular nonparametric ML method used to solve linear and nonlinear classification and regression problems [32]. It is essentially known for classification and pattern recognition problems for supervised learning. By finding the best line or hyperplane separator between two classes or more with the widest margins around that separator, it can accurately classify the data as far as possible as shown in Figure 4. The data points that are located closest to the margins are called support vectors because by using them we can find the maximum width for the separator. The SVM is also used for regression and forecasting problems with continuous data that is called Support Vector Regression (SVR). It has the same working principle of the SVM but instead of classifying it's used for data fitting. SVR finds the best linear regression model and its margins that fits the data. Similar to the ANN, an error should be minimized after several iterations by applying training dataset, whereas ANN finds a local minimum error, SVM deals with a convex that has a unique solution [35]. Moreover, any errors related to the estimated data between the margins are ignored. Therefore, SVR is considered to be a practical method to extend possible solutions. In the case of the most common nonlinear problems, flexible functions are used to transform the nonlinear data into linear space by mapping it, in which they are called kernel functions. Kernel functions can be an alternative to transfer functions in ANN, so they can make the learning easier and cheaper to achieve [13, 36].

The main algorithm for the SVR is having a linear equation fitting the input training dataset with its corresponding output $\{(x_1, y_1), \ldots . (x_N, y_N)\}$ attached to a vector of weights w and bias of b as in the initial ANN equation

$$f(x) = \langle w.x \rangle + b \tag{24}$$

Considering that (.) is the dot product between the x and w that produces a specific space in the model. And to make the function as flat as possible to avoid overfitting, which is considering the noise of the data, the width should be maximized between the margins to gain the widest solution as shown below

$$m = \frac{2}{\|w\|} \tag{25}$$

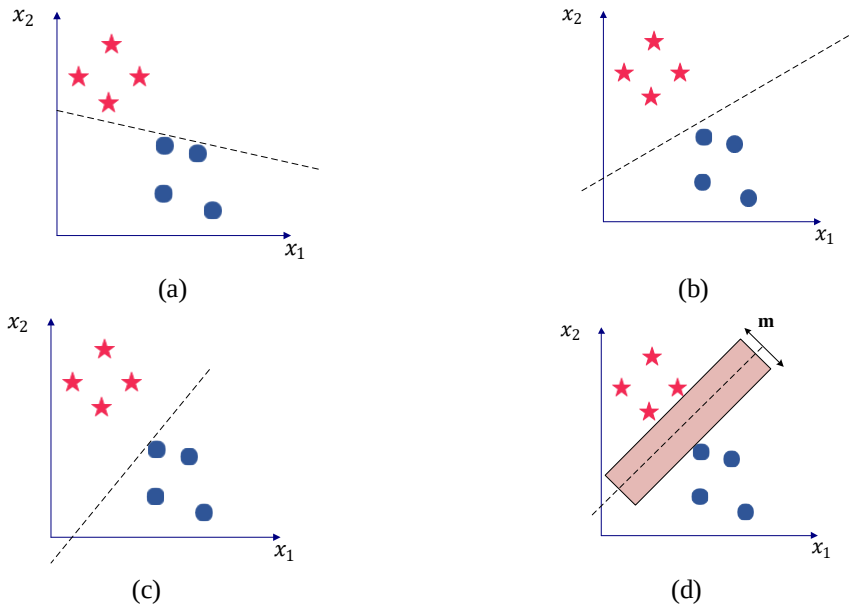


Figure 4. Different classification models: (a), (b) and (c) inaccurate classification; (d) SVM classification

While m is the width between the margins. This leads to minimizing w so the number of features is reduced to have the perfect linear regression that fits the data well. For a more convenient solution, a convex function is used to minimize the term below

$$\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}(s_i + s_i^*) \tag{26}$$

The second term is the generalization term. s_i, s_i^* are slack variables that are the errors represented by the closest data to the margins that are located outside them. C is the box constraint that helps to generalize the model to avoid the overfitting problem, it also tolerates the errors that are larger than ε, which is the width between the margin and the mainline as shown in Figure 5. Thus, it defines the flatness of the function and the tolerated large errors at the same time. Of course, there are conditions while minimizing the above equation as long as there are margins around that linear function and they are defined as [37]

$$\begin{gathered} y_i - \langle w.x_i\rangle - b \le \varepsilon + s_i \\ \langle w.x_i\rangle + b - y_i \le \varepsilon + s_i^* \\ s_i, s_i^* \ge 0 \end{gathered} \tag{27}$$

These inequalities represent both lines above and under the mainline. All the errors related to the data that are inside these margins are ignored as shown in the so-called ε-insensitive loss function which is mentioned below

$$|s|_\varepsilon := \begin{cases} 0 & if\ |s| \le \varepsilon \\ |s| - \varepsilon & otherwise \end{cases} \tag{28}$$

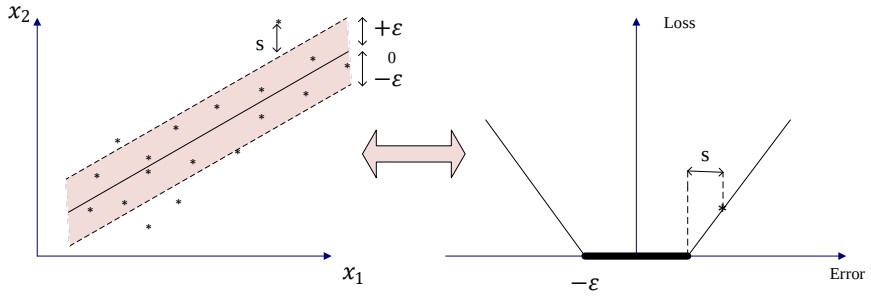


Figure 5. Linear SVR and its loss settings

If there is any function that needs to be optimized subject to other functions with constraints, whether it is minimization or maximization, a Lagrangian function is a method to solve such kinds of problems. So by using them, the main final model will be a dual linear problem. This model will be used to predict new values after learning is completed as shown below

$$f(x) = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*)\langle x_i . x\rangle + b \tag{29}$$

Most of the world's problems can't be solved linearly, instead, it has nonlinear data which it's impossible to be solved by the previous method. Therefore, kernels are used to transform small dimensional features of the data into higher dimensions, by mapping the data into nonlinear functions; so it can be predicted linearly. The kernel function is presented as

$$f(x, y) = (\varphi(x).\varphi(y)) \tag{30}$$

$\varphi(x)$ and $\varphi(y)$are nonlinear mappings for x and y that correspond to input features with the dot product between them. Thereby, for nonlinear problems in the model function instead of the $\langle x_i . x\rangle$ a kernel function is substituted as the following [38]

$$f(x) = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) k\langle x_i . x\rangle + b \tag{31}$$

Examples for common kernels used in SVM classification or regression models are linear, Gaussian and polynomial as respectively shown below

$$k\langle x_i . x_j\rangle = x_i' . x_j \tag{32}$$

$$k\langle x_i . x_j\rangle = e^{\left(-\|x_i - x_j\|^2\right)} \tag{33}$$

$$k\langle x_i . x_j\rangle = \left(1 + x_i' . x_j\right)^p \quad p \in \{2,3 \ldots .\} \tag{34}$$

3. APPLICATION AND RESULTS

Data was collected from a measuring station in Eskişehir city to compare it with the previous methodologies' implementation outputs. Therefore, in the next section, the type of data, how it was taken and prepared, software outputs, and discussion of the results are presented. Moreover, statistical errors are calculated and shown in tables and figures to clarify the comparison.

3.1. Data Preparation

This study is conducted in Eskişehir city which is located in the Anatolian region in Turkey that has an altitude of 788 meters, the latitude of 39.766193'' and the longitude of 30.526714''. For this study, the measured HGSR and daily GSR data are taken from the Turkish State Meteorological Service in Eskişehir from January 2014 to December 2014 that is 8760 samples of data. Some data were missing due to specific problems that occurred with the devices in the meteorological center, therefore the linear interpolation method is used to fill these missing data. Eliminating the measured HGSR at night hours that are equal to 0, improves the quality of the forecasting especially in ML methods because it should only be fed with meaningful data [31]. Therefore, only 3794 sunlight hours were taken to evaluate the proceeding models. For HGSR prediction using ML methods, the measured meteorological and geographical input features are used to be trained to gain the output. Relevant measured input features that are chosen for this study are solar time, solar hour angle, Julian day number, daily GSR, longitude, latitude, hourly average humidity, hourly temperature, and hourly pressure. After estimating the HGSR, some values have been negative, therefore they are taken as zeroes because of the physical meaning of solar radiation [18].

The calculated and the measured values are averaged over their corresponding months to ease the plotting and to compare them statistically. Statistical errors equations such as root mean square error (RMSE), mean absolute bias error (MABE), mean bias error (MBE), and correlation coefficient (R) are commonly used to test the accuracy level for solar radiation estimations [27]. The unit of the RMSE, MABE, and MBE are taken in w/m^2, whereas R is unitless.

$$RMSE = \sum_{i=1}^{n} \sqrt{\frac{1}{n}(m_i - p_i)^2} \tag{35}$$

$$MABE = \frac{1}{n}\sum_{i=1}^{n}(|m_i - p_i|) \tag{36}$$

$$MBE = \frac{1}{n}\sum_{i=1}^{n}(m_i - p_i) \tag{37}$$

$$R = \frac{\sum_{i=1}^{n}(p_i - p_a).(m_i - m_a)}{\sqrt{[\sum_{i=1}^{n}(p_i - p_a)^2][\sum_{i=1}^{n}(m_i - m_a)^2]}} \tag{38}$$

Where p_i, m_i p_a and m_a are the i^{th} predicted, measured HGSR and the averaged predicted, the measured HGSR respectively, and n is the number of data samples. A MATLAB program is used with its different toolboxes to compare the empirical model with ML methods.

3.2. Results

Among several existing empirical models such as Whillier, CPR, CPRG, Jain, and Newell models, CPRG is proved to work the best in different studies [18, 27]. These models were also applied in this study, but not mentioned in details so that the average RMSE for monthly average HGSR for Whillier, CPR, Jain and Newell models are 51 w/m^2, 40.6 w/m^2, 48.9 w/m^2 and 60.1 w/m^2 respectively. Whereas CPRG gave the minimum error with the value of 39.7 w/m^2.

Moving to the ML methods, ANN is designed with the help of MATLAB Deep Learning toolbox, and by using a fitting neural network tool to predict HGSR [39]. For the network architecture, as shown in Table 1, one hidden layer was first applied then two hidden layers with a different number of neurons as shown in Figure 6. As long as the transfer function tansig is used for the hidden and purelin for the

output layer, the data was normalized between [-1,1] by using the mapinmax function to make all the features in the same numerical range before the training part starts. The best training backpropagation algorithm mostly used is the Levenberg –Marquardt algorithm, which is applied in the learning session. In the model, 70%, 15%, 15% of the data are chosen for the training set, validation set, and test set respectively.

Table 1. ANN architecture specifications

	Number of neurons	Transfer function
Input layer	9	-
1st hidden layer	19	Tansig
2nd hidden layer	15	Tansig
Output layer	1	Pureline

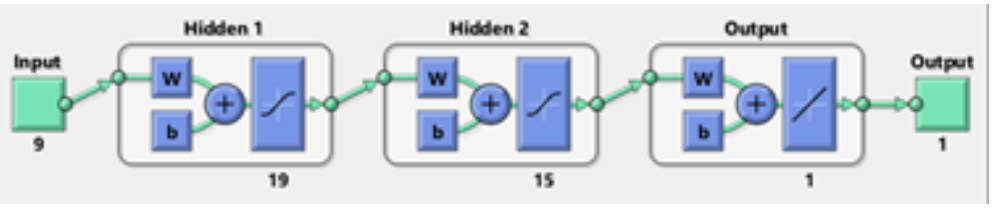
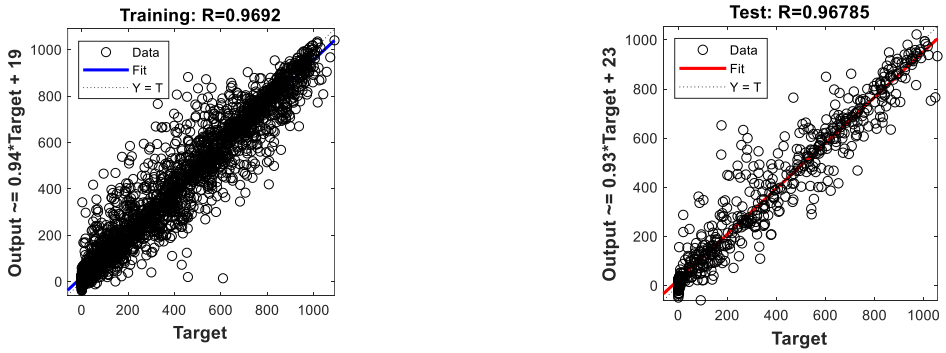


Figure 6. ANN model architecture

After many trials in training different combinations of metrological features and number of neurons, a model with 17 and 16 neurons in the first and second hidden layers gives the best result for the test set in HGSR prediction. Moreover, it's clear that the most important input feature is the solar hour with respect to each HGSR value which leads to giving the most accurate results with the minimum error. Figure 7 below illustrates the close regression values of the training and test data between the measured and the predicted HGSR with an average of 97%.



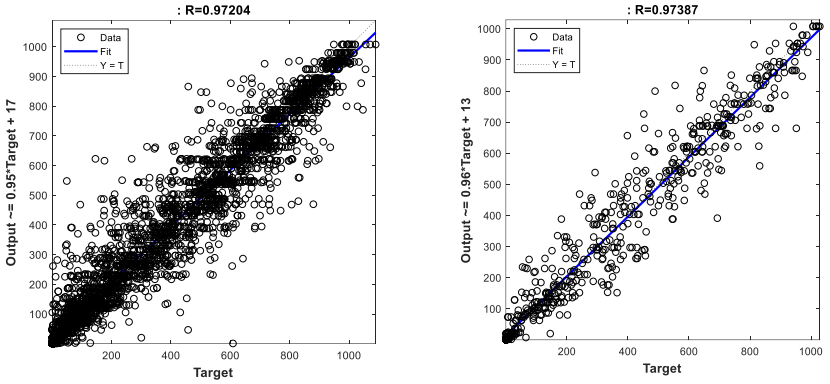
(a) (b)

Figure 7. ANN regression between measured and predicted HGSR for: (a) Training set; (b) Test set.

For estimating HGSR using RT and SVR technique, Machine Learning and Statistics toolbox in MATLAB is used via regression learner application [40]. This app can solve regression problems by applying supervised machine learning methods to predict continuous values. The dataset was divided as 85 % for training and validation and 15% for testing. K-fold cross-validation method is used to validate the model, which can be achieved by dividing the training set into k partitions and 1/k of them are used for validation and the rest for training. In each training session, a different fold is validated, therefore the solution can be reached faster and better. Fitrtree command is used to grow a binary RT that can

build a deep tree without pruning; which may not give acceptable results because of overfitting on testing data. Therefore, after growing an original deep tree consisted of 166 splits, pruning was done by 5 levels which gave the best result among other RT pruning levels. As shown in Figure 8 the solar hour angle is the most essential feature for predicting HGSR as it represents the root node. By increasing the pruning level an overfitting problem occurred because of not categorizing the data precisely.

Moreover, fitrsvm is used to train the data in the SVR model by mapping it in different kernel functions like linear, polynomial, and Gaussian functions. Although the accuracy is increased by increasing the degree of the model complexity, linear and polynomial kernels still don't give good results compared to the Gaussian kernel. And this refers to a lack of linear and polynomial flexibility for accepting infinite data with fixed derivatives, unlike the exponential that almost has the same nonzero derivative. Furthermore, the Gaussian kernel gives the best result even better than the ANN and RT by using box constraint C=200 and ε =10. The graphs in Figures 8 and 10 show the regression of the training and the testing set between the measured and predicted HGSR for the RT and SVR.



(a) (b)

Figure 8. RT regression between measured and predicted HGSR: (a) Training set; (b) Test set

Because of the big number of hourly data, considering the monthly average HGSR eases the statistical evaluation and plotting the models' graphs. This can be done by averaging the daily hourly data for each month. Table 2 shows the error differences between the training and testing sets for each of the ANN, RT, and SVR while the results in bold font represent the best. In addition, Table 3 clarifies the statistical evaluation of the monthly average HGSR for the empirical CPRG and ML models for each month of the year. In conclusion, Figure 11 illustrates the comparison between the empirical and ML models for the monthly average HGSR of the given models in each of March, June, September, and December respectively. It's noticed that the best model among all the models, in general, is the SVR which is closest to the measured HGSR and the worst one is the CPRG model. Therefore, this study confirms that ML methods have better forecasting than classical methods.

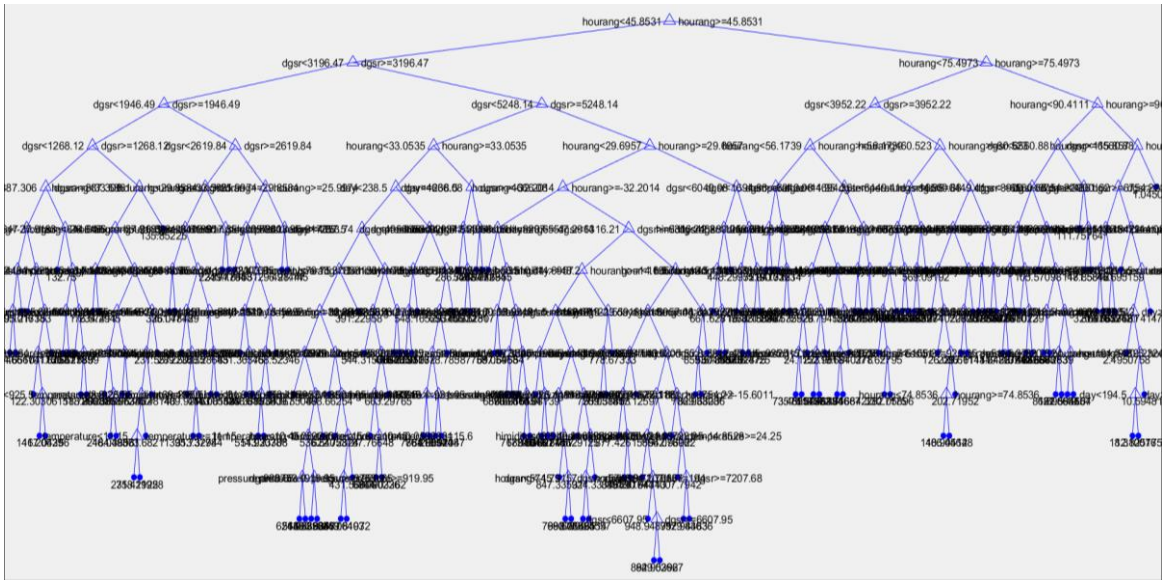
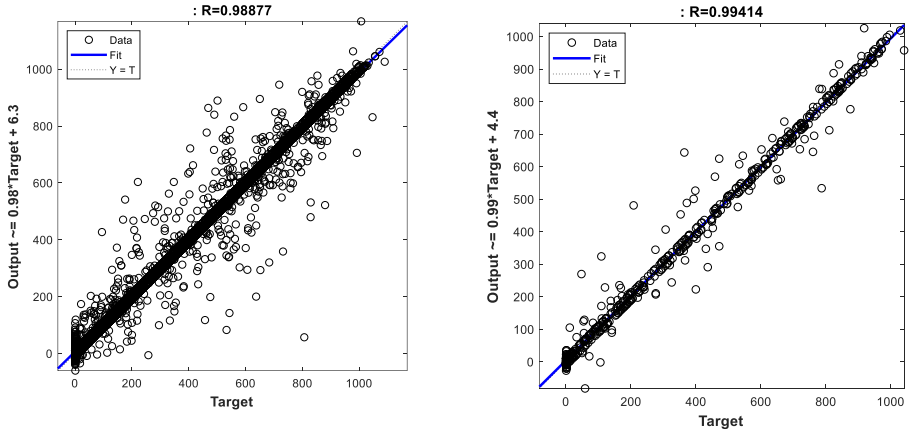


Figure 9. The final pruned RT model showing the solar hour angle as the root node.



(a) (b)

Figure 10. SVR regression between measured and predicted HGSR for: (a) Training set; (b) Test set

Table 2. ML models' statistical errors for evaluating training and testing HGSR.

		RMSE	MBE	MABE	R
ANN	**train set**	74.93054	-0.1753	46.69457	0.971199
	test set	80.59558	2.613323	50.6481	0.967679
RT	**train set**	75.37547	0.289289	44.19076	0.972041
	test set	66.0807	**-0.38557**	39.18967	0.978208
SVR	**train set**	64.0745	-0.01849	32.92851	0.979532
	test set	**63.86054**	-1.32338	**32.3517**	**0.978244**

4. CONCLUSION

To put in words, HGSR plays a significant role in the contribution of electricity generation in solar power systems in terms of a short time interval. Because of the absence of solar radiation measurement devices in some locations of the world, heuristic approaches must be implemented for more precise data prediction. Therefore, a comparison was done between a conventional model and well-known ML techniques to estimate HGSR in Eskişehir, Turkey. Thereby, the CPRG model was taken as an empirical model and soft computing methods like ANN, RT, and SVR were introduced as ML models. In this study, a detailed explanation of the mentioned methods' algorithms was presented. In addition, inputting geographical and metrological like solar time, solar hour angle, Julian day number, daily GSR, altitude, longitude, latitude, hourly average humidity, hourly temperature, and hourly pressure, was good enough to give acceptable results. In order to compare the performances of the ML models and CPRG empirical model, hourly measured data for one year was taken from the Turkish State Meteorological Service. Thus, the predicted output was compared with the measured values. The results gave a guarantee that modern ML methods can work more reliably and efficiently than the CPRG regression model. Nevertheless, SVR performed the best among other ML methods by obtaining the least statistical error. To summarize, this paper proved that SVR and other ML methods can be great alternatives to empirical regression models.

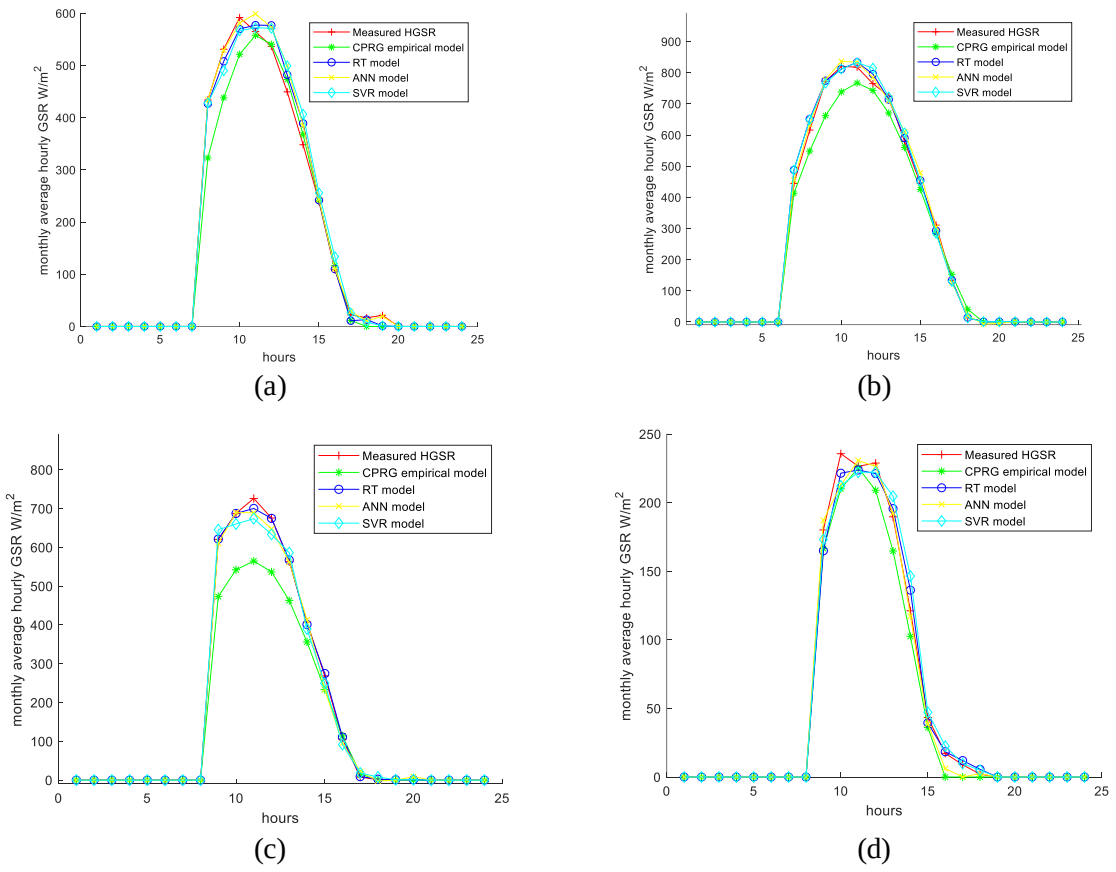


Figure 11. Monthly average HGSR literature model's comparison for: (a) March; (b) June; (c) September; (d) December

Table 3. Literature models' statistical errors for estimating monthly average HGSR

Prediction Method	Error Method	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
CPRG	RMSE	15.03319	26.13627	34.10927	48.70413	54.65664	37.06375	64.09028	54.39849	65.24304	43.5231	22.74272	10.48165
	MBE	-6.91323	-13.8901	-11.3687	-23.3892	-28.3953	-18.3451	-33.1001	-27.0272	-31.8999	-21.498	-12.3292	-5.80161
	MABE	7.504635	14.04189	16.19792	25.34866	29.99144	22.26535	35.63345	30.37257	33.30311	21.89928	12.32916	5.801612
	R	0.998055	0.999388	0.990782	0.994949	0.997057	0.998325	0.99793	0.997638	0.99858	0.998637	0.998437	0.997443
ANN	RMSE	4.268091	13.05792	12.88287	8.500711	13.53732	13.22816	9.85768	8.081314	11.04336	6.957904	4.206912	4.837296
	MBE	-0.46837	-4.41969	0.466351	-0.18462	-3.67558	5.713062	-4.30901	1.412025	-2.84283	-0.06526	1.132302	0.428567
	MABE	2.322021	6.926361	7.462376	4.911877	8.353253	8.337067	5.633042	4.687252	6.082831	3.089959	2.156793	2.48066
	R	0.999325	0.998526	0.998506	0.999494	0.999208	0.999455	0.999777	0.999764	0.999298	0.999392	0.999762	0.998444
RT	RMSE	8.575171	7.205377	15.98313	**4.827218**	12.87851	14.51225	9.02104	8.089537	**5.726949**	12.38327	6.285312	5.777986
	MBE	-1.83437	-0.90093	1.786015	**-0.7644**	-1.9288	4.578205	-1.34898	1.480706	**-0.57905**	-1.33017	0.36603	-0.62329
	MABE	3.887149	3.066967	9.069737	**2.801034**	6.458158	7.805767	5.11163	3.799542	**2.1591**	6.287892	3.525123	3.022709
	R	0.998198	0.999394	0.997739	**0.999836**	0.99914	0.999249	0.999696	0.99975	**0.999806**	0.998352	0.99896	0.998055
SVR	RMSE	**3.894752**	**4.123533**	**4.98863**	7.164349	**8.204289**	**6.984856**	**5.847646**	**5.627628**	6.674847	**4.126794**	**3.933571**	**3.828286**
	MBE	**0.509075**	**-0.44599**	**2.237**	-1.1662	**0.725183**	**-0.27017**	**2.154866**	**-1.02693**	1.501029	**0.608513**	**0.136558**	**0.152744**
	MABE	**1.952629**	**2.498142**	**2.818033**	4.11591	**4.543981**	**4.638173**	**3.077615**	**3.384649**	3.598797	**2.471459**	**1.958032**	**2.149016**
	R	**0.999462**	**0.999795**	**0.999845**	0.999682	**0.999618**	**0.999778**	**0.999937**	**0.99989**	0.999711	**0.999794**	**0.999584**	**0.999021**

REFERENCES

[1] Başaran Filik Ü, Filik T and Gerek Ö, New Electric Transmission Systems: Experiences, in Handbook of Clean Energy Systems, Wiley, 2015.

[2] Shaddel M, Javan DS and Baghernia P. Estimation of hourly global solar irradiation on tilted absorbers from horizontal one using Artificial Neural Network for case study of Mashhad, vol. 53, Elsevier Ltd, 2016, pp. 59-67.

[3] Khosravi A, Koury RN, Machado L and Pabon JJ. Prediction of hourly solar radiation in Abu Musa Island using machine learning algorithms, Journal of Cleaner Production, vol. 176, pp. 63-75, 2018.

[4] Voyant C, Notton G, Kalogirou S, et al. Machine learning methods for solar radiation forecasting: A review, vol. 105, Elsevier Ltd, 2017, pp. 569-582.

[5] Kaba K, Sarıgül M, Avcı M, et al. Estimation of daily global solar radiation using deep learning model, Energy, vol. 162, pp. 126-135, 2018.

[6] Sözen A, Arcaklioğlu E and Özalp M. Estimation of solar potential in Turkey by artificial neural networks using meteorological and geographical data, Energy Conversion and Management, vol. 45, no. 18-19, pp. 3033-3052, 2004.

[7] Elminir HK, Areed FF and Elsayed TS. Estimation of solar radiation components incident on Helwan site using neural networks, Solar Energy, 2005; vol. 79, no. 3, pp. 270-279.

[8] Angela K, Taddeo S and James M. Predicting Global Solar Radiation Using an Artificial Neural Network Single-Parameter Model, Advances in Artificial Neural Systems, vol. 2011, pp. 1-7.

[9] Salem H, Pharma H, Abdelhafez E, et al. Prediction of Hourly Solar Radiation in Amman Jordan by Using Artificial Neural Networks, Int. J. of Thermal & Environmental Engineering, 2017; vol. 14, no. 2, pp. 103-108.

[10] Troncoso A, Salcedo-Sanz S, Casanova-Mateo C, et al. Local models-based regression trees for very short-term wind speed prediction, Renewable Energy, 2015; vol. 81, pp. 589-598.

[11] Mori H and Takahashi A. A data mining method for selecting input variables for forecasting model of global solar radiation, in Proceedings of the IEEE Power Engineering Society Transmission and Distribution Conference, 2012.

[12] Chen JL, Li GS and Wu SJ. Assessing the potential of support vector machine for estimating daily solar radiation using sunshine duration, Energy Conversion and Management, 2013; vol. 75, pp. 311-318.

[13] Mohammadi K, Shamshirband S, Anisi MH, et al. Support vector regression based prediction of global solar radiation on a horizontal surface, Energy Conversion and Management, 2015; vol. 91, pp. 433-441.

[14] Ramedani Z, Omid M, Keyhani A, et al. A comparative study between fuzzy linear regression and support vector regression for global solar radiation prediction in Iran, Solar Energy, 2014; vol. 109, no. 1, pp. 135-143,

[15] Jiang H and Dong Y. Global horizontal radiation forecast using forward regression on a quadratic kernel support vector machine: Case study of the Tibet Autonomous Region in China, Energy, 2017; vol. 133, pp. 270-283.

[16] Bayrakçı HC, Demircan C and Keçebaş A. The development of empirical models for estimating global solar radiation on horizontal surface: A case study, vol. 81, Elsevier Ltd, 2018, pp. 2771-2782.

[17] Whillier A. The Determination of Hourly Values of Total Solar Radiation from Daily Summations, Arch für Meteorol, Geophys und Bioklimatol, 1956; vol. 7, p. 197–204.

[18] Yao W, Li Z, Xiu T, Lu Y and Li X. New decomposition models to estimate hourly global solar radiation from the daily value, Solar Energy, 2015; vol. 120, pp. 87-99.

[19] Liu BYH and Jordan RC. The Interrelationship and of Direct, Diffuse and Characteristic Distribution Total Solar Radiation, Solar Energy, 1960; vol. 4, no. 3, p. 1–19.

[20] Collares-Pereira M and Rabl A. The average distribution of solar radiation-correlations between diffuse and hemispherical and between daily and hourly insolation values, Solar Energy, 1979; vol. 22, no. 2, pp. 155-164.

[21] Gueymard C. Mean daily averages of beam radiation received by tilted surfaces as affected by the atmosphere, Solar Energy, 1986; vol. 37, no. 4, pp. 261-267.

[22] Garg H and Garg S. Improved correlation of daily and hourly diffuse radiation with global radiation for Indian stations, Solar & Wind Technology, 1987; vol. 4, no. 2, pp. 113-126.

[23] Jain P. Comparison of techniques for the estimation of daily global irradiation and a new technique for the estimation of hourly global irradiation, Solar & Wind Technology, 1984; vol. 1, no. 2, pp. 123-134.

[24] Jain P. Estimation of monthly average hourly global and diffuse irradiation, Solar & Wind Technology, 1988; vol. 5, no. 1, pp. 7-14.

[25] El shazly SM. Estimation of Hourly and Daily Global Solar Radiation at Clear Days Using an Approach Based on Modified Version of Gaussian Distribution, Advances in Atmospheric Sciences, 1996; vol. 13, no. 3, p. 349–358.

[26] Newell T. Simple models for hourly to daily radiation ratio correlations, Solar Energy, 1983; vol. 31, no. 3, pp. 339-342.

[27] Ayvazoğluyüksel Ö and Başaran Filik Ü. Estimation methods of global solar radiation, cell temperature and solar power forecasting: A review and case study in Eskişehir, Renewable and Sustainable Energy Reviews, 2018; vol. 91, pp. 639-653.

[28] Maleki SAM, Hizam H and Gomes C. Estimation of hourly, daily and monthly global solar radiation on inclined surfaces: Models re-visited, Energies, 2017; vol. 10, no. 1.

[29] Kumar R, Aggarwal RK and Sharma JD. Comparison of regression and artificial neural network models for estimation of global solar radiations, vol. 52, Elsevier Ltd, 2015, pp. 1294-1299.

[30] Khatib T, Mohamed A, Sopian K, et al. Assessment of artificial neural networks for hourly solar radiation prediction, International Journal of Photoenergy, vol. 2012, 2012.

[31] Alzahrani A, Shamsi P, Dagli C and Ferdowsi M, Solar Irradiance Forecasting using Deep Neural Networks, in Procedia Computer Science, 2017.

[32] Marsland S, Machine Learning & Pattern Recognition: An Algorithmic Perspective, Chapman & Hall/CRC, 2014.

[33] Breiman L, Friedman JH, Olshen RA, et al. Classification and regression trees., Wadsworth, Inc., 1984.

[34] Mori H and Kosemura N. Optimal Regression Tree Based Rule Discovery for Short-term Load Forecasting, 2000.

[35] Lauret P, Voyant C, Soubdhan T, et al. A benchmarking of machine learning techniques for solar radiation forecasting in an insular context, Solar Energy, 2015; vol. 112, pp. 446-457, 1 2.

[36] Dong Z, Yang D, Reindl T, et al. A novel hybrid approach based on self-organizing maps, support vector regression and particle swarm optimization to forecast solar irradiance, Energy 2015; vol. 82, pp. 570-577.

[37] Vapnik V. The Nature of Statistical Learning Theory, Springer-Verlag New York, 1995.

[38] Smola AJ and Schölkopf B. A tutorial on support vector regression, Kluwer Academic Publishers, 2004.

[39] Deep Learning Toolbox, MathWorks, 2019. [Online]. Available: https://www.mathworks.com/products/deep-learning.html.

[40] Machine learning and Statistics toolbox, MathWorks, 2019. [Online]. Available: https://www.mathworks.com/products/statistics.html.