Article

Not peer-reviewed version

Shor-Term Forecasting of Photovoltaic Power Using MLPNN, CNN AND kNN

Kelachukwu Iheanetu * and KeChrist Obileke

Posted Date: 8 May 2024

doi: 10.20944/preprints202405.0490.v1

Keywords: renewable energy; solar; photovoltaic; forecasting; data-driven; machine learning; modelling

Article

Shor-Term Forecasting of Photovoltaic Power Using MLPNN, CNN AND *k*NN

Kelachukwu Iheanetu * [1] and KeChrist Obileke [2]

[1] Rhodes University, Grahamstown, Makana, 6139, South Africa; kelaonline@gmail.com
[2] Department of Physics, University of Fort Hare, Alice, 5700, South Africa
* Correspondence: kelaonline@gmail.com

Abstract: This work focuses on short-term forecasting of PV output power. The multilayer perception (MLP), convolutional neural networks (CNN), and *k*-nearest neighbour (*k*NN) neural networks have been used singly or in a hybrid (with other algorithms) to forecast solar PV power or global solar irradiance with good success. Their performances in forecasting PV power have been compared with other algorithms but not with themselves. The study aims to compare performance of a number of neural network algorithms in solar PV energy yield forecasting under different weather conditions. The performance of MLPNN, CNN and kNN are being compared using solar PV (hourly) energy yield data for Grahamstown, Eastern Cape, South Africa. The choice of location is part of the study parameters to provide insight into renewable energy power integration in specific areas in south Africa that may be prone to extreme weather conditions. The *k*NN algorithm was found to have RMSE value of 4.95% and MAE value of 2.74% at its worst performance, and RMSE value of 1.49% and MAE value of 0.85% at its best performance. It outperformed the others by a good margin and *k*NN could serve as a fast, easy, and accurate tool to forecast solar PV output power.

Keywords: renewable energy; solar; photovoltaic; forecasting; data-driven; machine learning; modelling

1. Introduction

The world's energy suppliers are shifting towards using clean, renewable energy sources to reduce the pollution caused by fossil fuel energy sources. Photovoltaic and wind energy sources are the most favoured renewable energy sources because they have zero emissions, require minimal maintenance, and their initial installation cost is also coming down [1] recently. The output power of solar photovoltaic (PV) energy systems is highly dependent on constantly changing weather and environmental conditions like solar irradiance, wind speed, ambient temperature, module temperature, etc. There is need to forecast its output power to effectively plan and integrate the solar PV energy system into the main grid.

Many approaches and techniques have been used to predict solar PV output power. The physical models, the statistical models, and a hybrid (combination of physical statistical) models [2–5] are some of the major approaches which have been used to predict PV output power. The physical approach is designed using the global irradiance of the solar PV cells and a mathematical model describing the solar PV system [6]. The techniques produce high accuracy when the weather conditions are stable throughout the prediction period. The total sky imagers [7] and image technique [8] which analyses the solar surface irradiance of retrieved satellite images to make prediction are example of some implementations of the physical method. The statistical techniques are designed mostly for the principle of persistence. Using tested scientific processes, they predict the PV output power by establishing a relationship between the input variables (vectors) and the target output power. The weather parameters (solar irradiance, wind speed, ambient temperature, module temperature, rain, humidity etc.) which directly or indirectly affect the solar panels' electricity

 doi:10.20944/preprints202405.0490.v1

generation constitute the input vectors, while the PV output power is the predicted output. Traditional statistical methods [9] use regression analyses to produce models that forecast the PV output power.

Artificial intelligence (AI) or machine learning (ML) is another way of applying this technique. A good example of the AI techniques which have been used to forecast PV output power is artificial neural networks (ANN) [10], long short-term memory (LSTM) [11,12], support vector machines (SVM) [9,10] etc. The multilayer perceptron (MLP) neural network [13], the convolutional neural network (CNN) [14,15], gated recurrent units (GRU) [16,17] and k-nearest neighbour (*k*NN) [17,18] are some instances of the ANN which have been successfully used to model and forecast solar PV output power. Ratshilengo et al. [19] compared the results of modelling the global solar irradiance with the generic algorithm (GA), recurrent neural network (RNN), and *k*NN techniques and showed that GA outperformed others in accuracy. Most of these researches focused on a single technique or forecasted solar irradiation (when they worked with more than one technique), but in this study, we will compare the results of modelling the actual solar PV output power using MLPNN, CNN and *k*NN algorithms and show that the *k*NN method had the best overall performance on our data. It is more important to model the output power instead of solar irradiance because the generated PV output power also captures the impact of the ambient and module temperatures, whose rise negatively affects the PV output power as well as the impart of other factors that affect solar irradiance. A comparative performance analysis has not been done on these modeling solar PV output power forecasting.

In Section 2 of this work, we present a brief review of PV power output forecasting, and in Section 3, we present a detailed review of the artificial neural network. Section 4 presents data description, variable selection, and evaluation metrics. Section 5 presents the results and discussion, while Section 6 considers the challenges of PV output power forecasting, while conclusions are drawn in Section 7.

2. A Brief Overview of Solar PV Power Prediction in the Literature

Numerous studies have been published on forecasting PV output power. When solar panels receive irradiance, they convert the incident irradiance to electricity. Hence, solar irradiation strongly correlates with solar PV panels' output power. Machine learning techniques like the ANN [20], support vector machines (SVMs) [21], *k*NN, etc., have been used to forecast solar irradiance. ML techniques are equipped with the ability to capture complex nonlinear mapping between input and output data. Efforts have been made to model solar PV output power with ANNs. Liu and Zhang [12] model the solar PV output power using *k*NN and analyse the performance of their model for cloudy, clear sky and overcast weather conditions. Ratshilengo et al. [19] compared the performance of generic algorithm (GA), recurrent neural networks (RNN) and *k*NN in modelling solar irradiance. They found GA outperformed the other two using their performance metrics. A combination of autoregressive and dynamic system approach for hour-ahead global solar irradiance forecasting was proposed by [22]. Table 1 summarises some previous study on solar PV output power prediction.

Table 2. A summary literature review of PV power output forecasting showing references, forecast horizon, technique, and results.

Ref.	Forecast horizon	Target	Forecast method	Forecast error
[19]	Short-term	PV power	LSTM	RMSE=67.8 %, MAE=43.8%, NRMSE=0.19%
			CNN	RMSE=38.5%, MAE=4.0%, NRMSE=0.04%

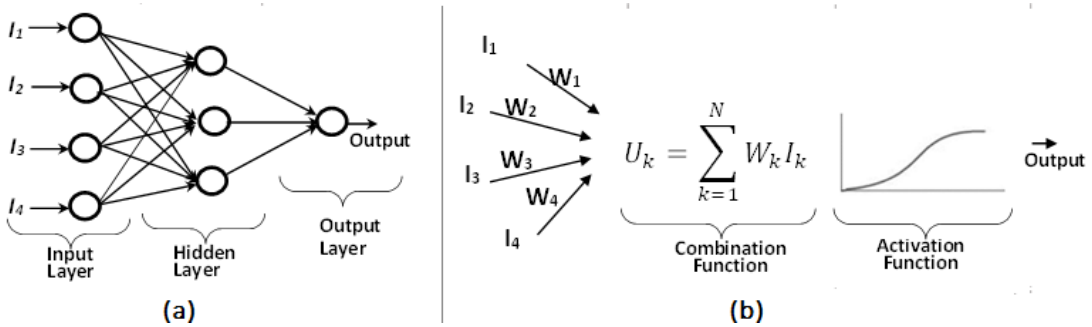
 doi:10.20944/preprints202405.0490.v1

			CNN-LSTM	RMSE=5.2%, MAE=2.9%, NRMSE=-0.03%
[19]	Short-term	Irradiance	RNN	RMSE=56.89%, MAE=20.18%, rRME=7.54%, rMAE-4.49%
			KNN	RMSE=57.48%, MAE=20.94%, rRME=7.58%, rMAE=4.58%
			GA	RMSE=35.50%, MAE=26.74%, rRME=5.95%, rMAE=5.17%
[23]	Very-short-term	PV power	Persistence, MPL, CNN, LSTM	RMSE=15.3%
[24]	Short-term	PV power	Similarity algorithm, KNN, NARX, and smart persistence models	RMSE=2.3%
[25]	Short-term	PV power	Hybrid model of wavelet decomposition and ANN	RMSE values between 7.193%–19.663%
[26]	Short- and long-term	PV power	Prophet, LSTM, CNN, C-LSTM	MAE range 2.9 - 16730.3, RMSE range 5.2 - 21753.2 NRMSE range 0.0 - 30.59
[13]	Short-term	Irradiance	MLPNN	MAPE=6.15%
[27]	Short-term	Wind power	K-means clustering method	MAPE ≈ 11%

Some ways to forecast solar PV power are by modelling irradiance (indirectly modelling PV output power) or directly modelling the PV output power. A lot of research has been published in this regard.

2. Artificial Neural Network

ANN is one technique which has been used extensively to model and forecast solar PV output power with high accuracy [28,29]. This comes from its ability to capture the complex nonlinear relationship between the input features (weather and environmental data) and corresponding output power. ANN is a set of computational systems composed of many simple processing units inspired by the human nervous system. Figure 1a shows a schematic representation of a basic ANN, with the input, hidden and output layers, connections, and neurons. Data of the (input) features are fed into the input layer. The hidden layer (which could be more than one) processes and analyses this input data. The output layer completes the process by finalising and providing the network output. The connections connect neurons in the adjacent layer together with the updated weights.



 doi:10.20944/preprints202405.0490.v1

Figure 1. (**a**) Schematic representation of a typical ANN - having the input, hidden and output layers. (**b**) A pictorial presentation of a mathematical model of an ANN cell [5].

Figure 1b presents a pictural representation of basic ANN mathematics. It shows that the neuron of a basic ANN cell is made of two parts – the activation and combination functions. The network sums up all the input values using the activation function, making the activation function act like a squeezing transfer function on the input to produce the output results. Some commonly used activation functions are sigmoid, linear, hyperbolic tangent sigmoid, Gaussian radial basis, bipolar linear and unipolar step. The basic mathematical expression of an ANN is given as [30]:

$$U_j = b + \sum_{k=1}^{N} (W_k \times I_k), \qquad (1)$$

where U_j is the predicted network output, b is the bias weight, N is the number of inputs, W_k is the connection weight, and I_k is network input. There are many types of neurons and interconnections used in ANN. Some examples of this are feedforward and backpropagation NN. Feedforward NNs pass information/data in one forward direction only. The backpropagation NN allows the process to cycle through over again. it loops back, and information learnt in the previous iteration is used to update the hyperparameters (weights) during the next iteration to improve prediction. Deep learning is a type of ANN where its layers are arranged hierarchically to learn complex features from simple ones [14]. One weakness of the deep learning NN is that it takes a relatively long time to train the model.

There are two basic stages of the ANN – training and testing. The data for modelling PV output power is often split into training and test sets. Generally, 80% of the data is set aside for training, while 20% is reserved for testing. During the training stage, the neural network uses the training dataset to learn and find a mapping relationship between the input data by updating the synaptic weights. Prediction errors are calculated using the forecasted and measured values. The magnitude of the errors is used to update the weights and biases, and the process is repeated until the desired accuracy level is achieved. The testing dataset is used to test the final model produced in the training stage, and the ANN model's performance is evaluated. A statistical approach that considers each experimental run as a test called the design of experiment approach, was described by [31] for use with ANNs.

Neural networks having a single hidden layer is usually enough to solve most data modelling problems but complex nonlinear mapping patterns between the input and output data may require the use of two or more hidden layers to obtain accurate results. Multilayer feedforward neural networks (MLFFNN) [32], adaptive neuro-fuzzy interface systems [33–36], multilayer perceptron neural networks (MLPNN) [13,37], convolutional neural networks (CNN) [14,37] are some examples of ANN with multiple layers. In this study, we will compare the results of modelling solar PV output power using ANN-MLPNN, CNN, gated recurrent units (GRU) and *k*NN models. Subsequent sections will present a brief overview of these techniques.

2.1.1. Multilayer Perceptron Neural Networks (MLPNN)

MLPNN is a special type of ANN that is organised in layers and can be used for classification and regression depending on the activation function used. A typical MLPNN has three layers like most ANN – the input, output and hidden layers. The hidden layer can have more than one hidden unit depending on the complexity of the problem at hand. Let I_p be a p-th point in an N-dimensional input to MLPNN, the output be Y_p, and the weight of on hidden layer be W_h. To keep the discussion simple, take the case of a single-layer MLP. The output of the first hidden unit L_1 can be expressed as:

$$L_1(i) = \sum_{k=1}^{N+1} W_h(i,k) I_p(k). \qquad (2)$$

A linear activation function could be given as:

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 May 2024 doi:10.20944/preprints202405.0490.v1

$$O_p(i) = f(L_p(i)). \quad (3)$$

And the nonlinear activation function could be given as:

$$f(L_p(i)) = \frac{1}{(1 + e^{-L_p(i)})}. \quad (4)$$

MLPNN algorithm applies the weight of the previous iteration when calculating that of the next iteration. Let W_1 be the weight of the input to the hidden layer, and W_2 that of the hidden to the output layers. Then, the overall output Y_p is given as [38]:

$$Y_1(i) = \sum_{k=1}^{N+1} W_1(i,k) I_p(k) + \sum_{j=1}^{N_h} W_2(i,j) O_p(j). \quad (5)$$

Every layer of the MLP receives input from the previous layer and sends its output to the next layer, which receives it as input and so on. Hence, every layer has input, weight, bias, and output vectors. The input layer has an activation function but no thresholds. It connects and transfers data to successive layers. The hidden and the output layers have weights assigned to them together with their thresholds. At each layer, the input vectors are multiplied with the layers corresponding threshold and passed through the activations function, which could be linear or nonlinear [39]. Some of the advantages of MLPNN are that it requires no prior assumptions, no relative importance to be given to the input dataset and adjustment weights at the training stage [40,41].

2.1.2. Convolutional Neural Networks (CNNs)

The CNNs are another commonly used deep learning feedforward NN used to model PV output power whose inputs are tensors. They have many hidden convolutional layers that can be combined with other types of layers, such as the pooling layer. CNN has been used effectively in image processing, signal process, audio classification and time series processing.

Figure 2 presents a schematic illustration of the CNN with a one-dimensional convolutional layer. It shows the input and a one-dimensional convolution layer, a dropout layer to prevent overfitting, a dense layer of fully connected neurons, a flattening layer, and the output layer.

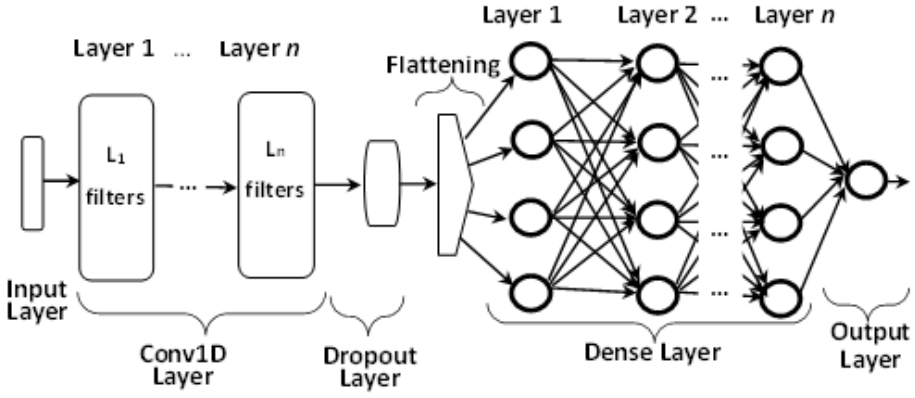


Figure 2. Schematic representation of a convolutional network.

2.1.3. k-Nearest Neighbour (kNN)

The *k*NN is a simple supervised ML algorithm that can be applied to solve regression and classification problems [42]. Supervised ML is a type of ML that require the use of labelled input and output data, while unsupervised ML is the process of analysing unlabeled data. The supervised ML model tries to learn the mapping relationship between the labelled input features and output data. The model is finetuned till the desired forecasting accuracy is achieved. The *k*NN algorithm, like most forecasting algorithms, works by using training data as the "basis" for predicting future values. In the algorithm, *Neighbours* are chosen from the basis and sorted depending on certain similarity

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 May 2024 doi:10.20944/preprints202405.0490.v1

criteria between the attributes of the training data and that of the testing data. The attributes are the training and testing data's weather and PV output power data, while the target is the residual of the difference between them. The mean of the target values of the neighbours is used to forecast the PV power. The measure of similarity (e.g., the Manhattan distance) is given as [43]:

$$d_j = \sum_{k=1}^{n} W_k \left| x_{\text{train},j,k} - x_{\text{test},k} \right|, \qquad (6)$$

where d_j is the distance between the i-th training and test data, W_k is the weight of the j-th attribute, and attribute values of the training and test are x_{train} and x_{test}, respectively. j and k are the indices of the training data and test attributes, respectively, while n is the number of attributes. The weights were calculated using the k-fold Cross Validation [44]. The k target values are used to forecasted residual F_R as:

$$F_R = \frac{\sum_{k=1}^{M} v_k D_{\text{train},k}}{\sum_{k=1}^{M} v_k}, \qquad (7)$$

where D_{train} is the training data-target value, k is the index of the neighbours' chosen training data, and v_k is the weight of the corresponding i-th target value. At the same time, M represents the total number of nearest neighbours. One advantage of the *k*NN is that it requires no training time. Another is that it is simple to apply, and new data samples can easily be added. The *k*NN also has a few disadvantages. These include that it is ineffective in handling very large data and it performs poorly with high dimension data. Another disadvantage is that it is sensitive to noisy data (having outliers and missing values).

The *k*NN algorithm works; thus, start by loading the dataset, then initialise K to the number of neighbours. For every record, calculate the distance between it and the current record, and add the distance and indices to an ascending ordered set of distances. Then, select the first K entries in the ordered list, retrieve the labels for the selected K entries, and return the average of the K labels in the case of regression (or return the mode of the K labels in the case of classification).

3. Data Description and Variable Selection

3.1. Data Description

We have a time series hourly data having fields for PV output power, normal global irradiance, diffused irradiance, sun height, ambient temperature, reflected irradiance, wind speed, and 24-hour time cycle in Grahamstown, Eastern Cape, South Africa for the period from 2009 to 2020. Figure 3 presents the graph of the data – the PV output power.

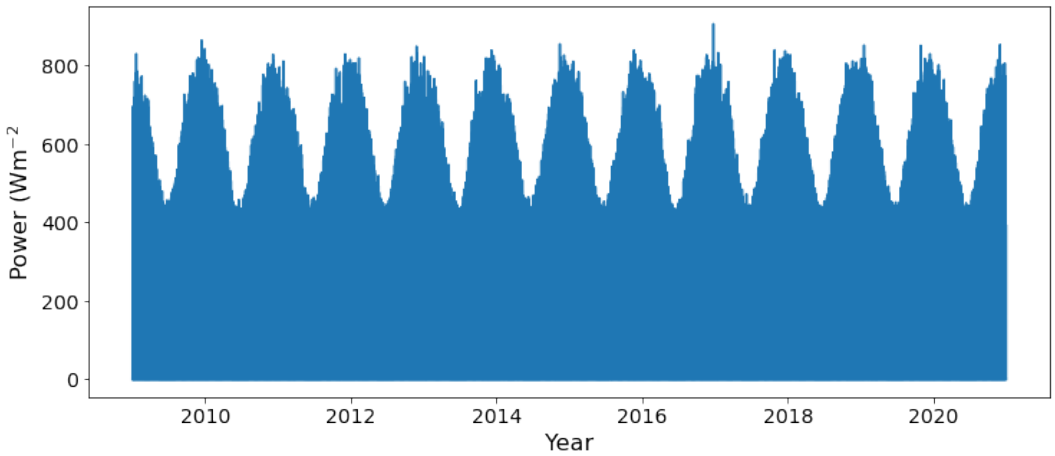


Figure 3. Plot the PV output power from 2009 to 2020.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 May 2024 doi:10.20944/preprints202405.0490.v1

3.2. Selecting Input Variables

The more variables used as input, the better the performance of the algorithms but the higher the execution time and the higher the chances of overfitting. To select the variables that will serve as inputs to the algorithms, we consider the interaction between the variables and their correlation with the output power. Figure 4 presents scatterplots of all pairs of attributes. This figure can help one to see the relationship between the variables.

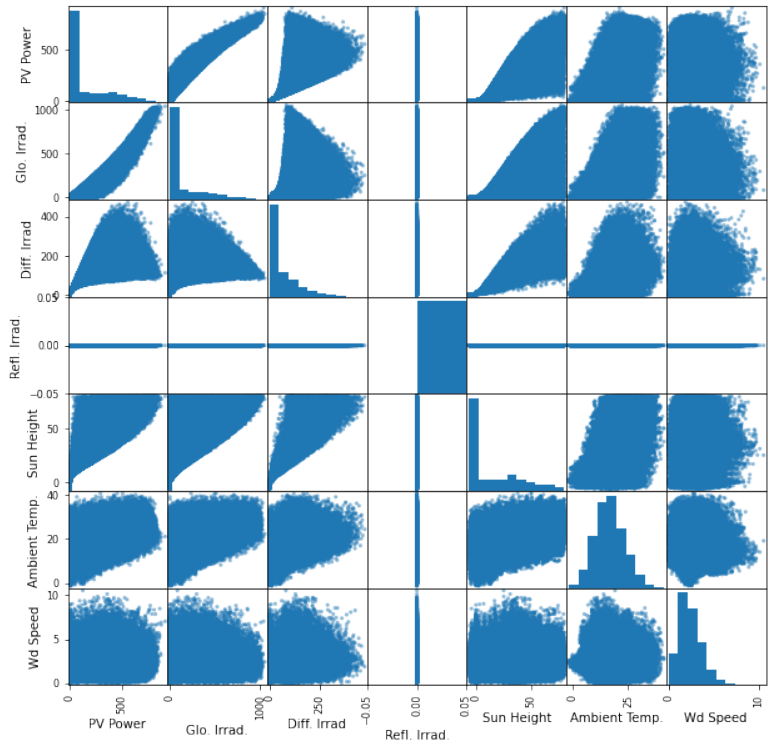


Figure 4. Plots of the variables.

The diagonal plots display the Gaussian distribution of the values of each variable. As expected, there is a strong correlation between global (and the diffused) solar irradiance and PV power but no correlation between reflected irradiance and PV power. The fact will be demonstrated more quantitatively later using the Lasso regression analysis. One cannot precisely say for the other variables. We excluded the reflected solar irradiance from the list of input variables.

3.3. Prediction Intervals and Performance Evaluation

3.3.1. Prediction Intervals

The prediction interval (PI) helps energy providers and operators assess the uncertainty level in electrical energy they supply [45,46]. It is a great tool for measuring uncertainty in model predictions. We will subsequently take a brief look at prediction interval widths.

The prediction interval width (PIW_t) is the estimated difference between the upper (U_t)and lower L_t limits of the values given as:

$$\mathrm{PIW}_t = U_t - L_t \quad t = 1,2,3, \ldots . N. \tag{8}$$

The PI coverage probability (PICP) and PI normalised average width (PINAW) are used to assess the performance of the prediction intervals. The PICP is used to estimate the reliability of the PIs, while PINAW is used to assess the width of the PIs. These two are expressed mathematically as [47]:

$$\mathrm{PICP} = \frac{1}{N}\sum_{t=1}^{N} c_t, \qquad \mathrm{c}_t = \begin{cases} 1 & \text{if } \ y_t \in (L_t, U_t) \\ 0 & \text{otherwise} \end{cases}, \tag{9}$$

 doi:10.20944/preprints202405.0490.v1

$$\text{PINAW} = \frac{1}{N}\sum_{t=1}^{N}\frac{\text{PIW}_t}{y_{\max} - y_{\min}}, \qquad (10)$$

where y_t is the data, $y_{\min}$ and $y_{\max}$ are the minimum and maximum values of PIW, respectively. The PIs are weighted against a predetermined confidence interval (CI) value. One has valid PIs values when the value of PICP is greater than or equal to that predefined CI value. The PI normarlised average deviation (PINAD) defines the degree of deviation from the actual value to the PIs and is expressed mathematically as [47]:

$$\text{PINAD} = \frac{1}{N(y_{\max} - y_{\min})}\sum_{t=1}^{N} c_t, \quad c_t = \begin{cases} L_t - y_t, \text{ if } y_t < L_t \\ 0 \;\; if \;\; L_t \le y_t \le U_t \\ y_t - U_t, \text{ if } y_t > U_t \end{cases}. \qquad (11)$$

3.3.2. Performance Matrices

A good number of performance measurement tools are available in the literature. Some are better fit for particular contexts and target objectives.

The mean absolute error (MAE) is the average of the absolute difference between the measured (y_t) and predicted ($\hat{y}_t$) data. For a total of N predictions, the MAE is given as:

$$\text{MAE} = \frac{1}{N}\sum_{t=1}^{N}|y_t - \hat{y}_t|, \qquad (12)$$

The relative MAE (rMAE) gives an MAE value is comparable to the measured values. The rMAE is given mathematically as:

$$\text{rMAE} = \frac{1}{N}\sum_{t=1}^{N}\frac{|y_t - \hat{y}_t|}{y_t}. \qquad (13)$$

The root mean squared error (RMSE) is the average of the squared difference between the measured and predicted values. The average of the square of the prediction residual. It is always non-negative and is given as:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{t=1}^{N}(y_t - \hat{y}_t)^2}. \qquad (14)$$

The relative RMSE (rRMSE) gives a percentage RMSE value. The rRMSE is given as:

$$\text{rRMSE} = \frac{100}{\bar{y}}\sqrt{\frac{1}{N}\sum_{t=1}^{N}(y_t - \hat{y}_t)^2}. \qquad (15)$$

where $\bar{y}$ is the average of y_t, $t = 1,2,3,\ldots N$. The smaller values for these error metrics, the more accurate the forecasted value.

The R^2 score is another commonly used metric to measure the performance of a forecast. The R^2 score can be expressed mathematically as:

$$R^2 = 1\,\frac{\sum_{t=1}^{N}(\hat{y}_t - y_t)^2}{\sum_{t=1}^{N}(\bar{y}_t - y_t)^2}, \qquad (16)$$

The closer the value of R^2 is close to 1, the more accurate the prediction is the true value.

It is common practice to normalise (or scale) data before passing through the training step, but we did not do this in our case because our data had a few missing records and outliers.

3.4. Selecting Input Variables

It is a common practice to use Lasso analysis to perform variable selection, which uses the ℓ loss function penalty given as [4]:

$$\hat{\beta}_{\text{Lasso}}(\lambda) = \arg min\left\|\vec{y} - X\hat{\beta}\right\|_2^2 + \lambda\hat{\beta}||_1. \qquad (17)$$

In Table 2, we show the parametric coefficient of the Lasso regression analysis. All the variables except for the reflected irradiance are important forecasting variables.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 May 2024 doi:10.20944/preprints202405.0490.v1

Table 2. Parameter coefficient of Lasso regression.

Variables	Coefficients
Global Normal irradiance	0.790206
Diffuse irradiance	0.902841
Reflected irradiance	0.000000
Sun Elevation	-0.412872
Ambient Temperature	-0.817793
Wind Speed	1.017501
24-hour time cycle	0.186437

4. Results

4.1. Prediction Results

Figure 5 presents plots of the data and fits of the different models we used in this study for short-term forecasting (38 hours ahead) of the solar PV output power for two clear sky days and two cloudy days. The graph in blue is the measured data, while that in orange, green, red, and cyan are for MLPNN, CNN and *k*NN models' forecast, respectively. We can see visually from these plots that the prediction produced by *k*NN best fits the data for these two conditions. MLPNN also produces a reasonably good fit on a clear sky day.

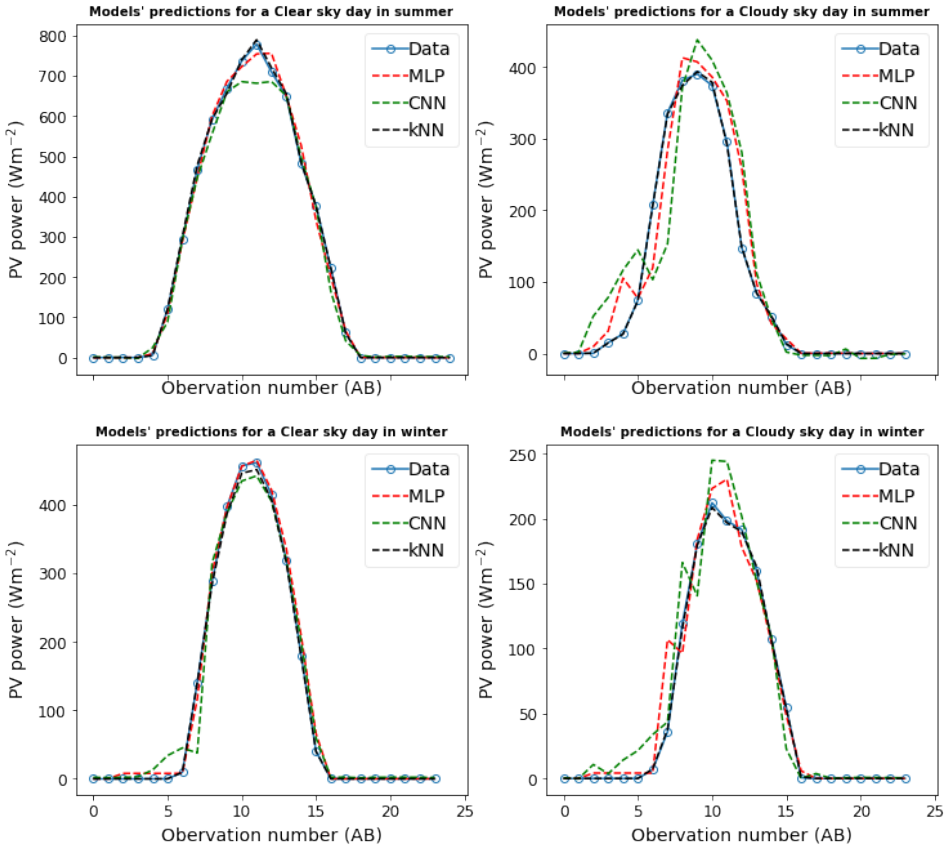


Figure 5. Plots of the Sola PV output power (Data) superimposed with MLPNN, CNN, GRU and *k*NN models' predictions (dash lines) on a summer clear sky day (top left panel) and cloudy day (top right panel). The same plots are show for a winter clear sky day (bottom left panel) and cloudy day (bottom right panel). The solid lines represent the measured data while dashed line the predictions.

Figure 6 presents the measured solar PV output power together density plots and the different models' predictions. The solid blue line graph is the measure data, while the dashed lines represent

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 May 2024 doi:10.20944/preprints202405.0490.v1

the models' forecasts. From these graphs, it can be observed that *k*NN prediction best matches the data, followed closely by the MLPNN predictions. We will subsequently present a qualitative evaluation of these models' performance.

Table 3. Evaluating models' performances on summer (top panels) and winter (bottom panels) for clear and cloudy sky days (left and right columns respectively).

	Clear sky day in summer			Cloudy sky day in summer		
	MLPNN	CNN	KNN	MLPNN	CNN	KNN
RMSE	21.42	23.15	4.95	39.35	67.54	2.08
rRMSE	8.69	9.39	2.01	39.40	67.62	2.08
MAE	12.34	14.04	2.74	21.86	46.19	1.11
rMAE	0.49	0.56	0.11	0.91	1.92	0.05
R^2	0.99	0.99	1.00	0.92	0.77	1.00

	Clear sky day in winter			Cloudy sky day in winter		
	MLPNN	CNN	KNN	MLPNN	CNN	KNN
RMSE	10.96	25.69	4.11	17.22	20.09	1.49
rRMSE	9.71	22.77	3.64	32.59	38.04	2.82
MAE	6.47	14.09	2.00	8.18	12.88	0.85
rMAE	0.27	0.59	0.08	0.34	0.54	0.04
R^2	1.00	0.98	1.00	0.95	0.93	1.00

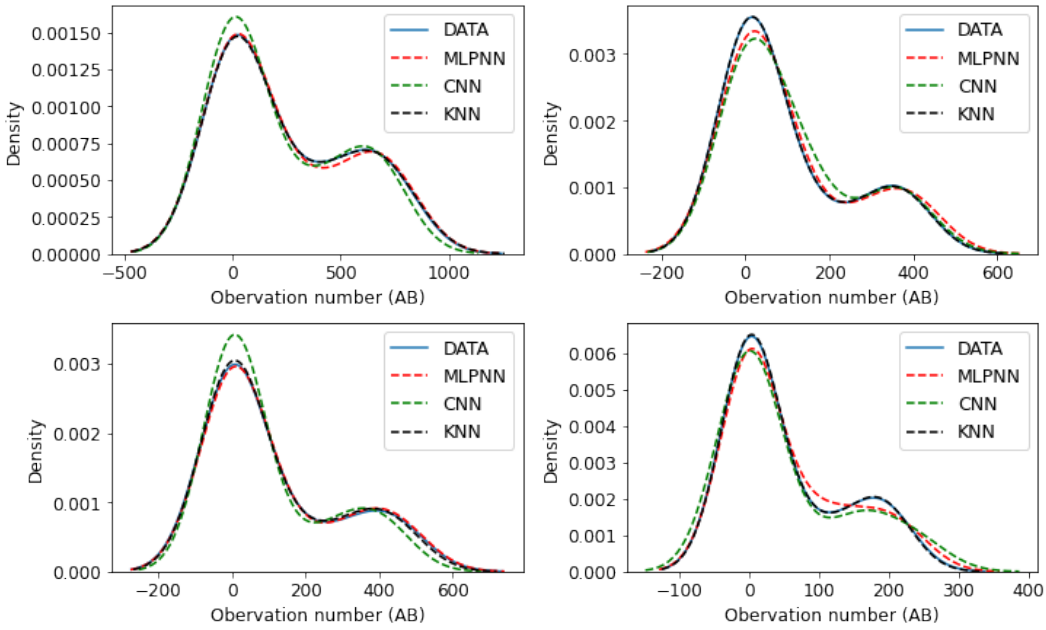


Figure 6. Density **p**lots of the measured data (data – solid line) together with the model's forecast (dash lines). In the top row is the graph for the models' predictions on a summer clear sky day (left panel) and cloudy day (right panel), while the bottom panel presents the same on a winter clear sky day (left panel) and cloudy day (right panel).

Figure 6 presents the density plots of the measure solar PV output power together with the models' predictions during the summer season (top row) on a clear sky day (in the left panel) and a cloudy day (in the right panel). The same is present for a winter clear sky day (left panel) and cloudy day (right panel) on the bottom row. The *k*NN model's density graph produced the closest match to the measured data for all the four weather conditions under investigation.

 doi:10.20944/preprints202405.0490.v1

In Table 3, we present the results of evaluating our models' performance using MAE, rMAE, RMSE, rRMSE, and R^2 metrics for the four weather conditions. The *k*NN has the overall best performance for these metrics, followed by the MLPNN, then CNN.

4.2. Prediction Accuracy Analysis

This section evaluates how the models' predictions are centred using PIs and the forecast error distribution.

4.2.1. Prediction Interval Evaluation

In Table 4 we compare the performance confidence intervals of these modes' predictions using PICP, PINAW and PINAD with a preset confidence level of 95%. Only the *k*NN model has a value of PICP greater than 95% on the clear sky days. The model with the lowest value for PINAD and the narrowest PINAW is the model that best fits the data [47]. *k*NN has the smallest PINAD and has the best overall performance with respect to these prediction interval matrices.

Table 4. Comparing the performance of the models using PICD, PINAW and PINAD on a confidence level set to 95% - on clear sky and cloudy days (top row left and right panels, respectively), while second row presents the same for a winter clear sky and cloudy days (bottom row left and right panels, respectively).

	Clear sky day in summer			Cloudy sky day in summer		
	MLPNN	CNN	KNN	MLPNN	CNN	KNN
PICP	28.0	28.95	96	12.5	4.17	91.67
PINAW	0.631	0.622	0.637	0.561	0.633	0.511
PINAD	0.0520	0.0989	0.0002	0.4510	1.0619	0.0011

	Clear sky day in winter			Cloudy sky day in winter		
	MLPNN	CNN	KNN	MLPNN	CNN	KNN
PICP	20.83	20.83	100.0	12.5	4.16	87.50
PINAW	0.507	0.455	0.481	0. 530	0.526	0.495
PINAD	0.0866	0. 2212	0.0000	0. 2769	0.5736	0.0016

4.2.2. Analysing Residuals

In Table 5, statistical analyses on the residuals of all the models' predictions is presented for MLPNN, CNN, and *k*NN models (with a confidence level of 95%) on a summer clear sky day. The table shows that *k*NN has the smallest standard deviation among the three models under investigation, which implies that it produces the best fit for the data. MLPNN has the next best fit to the data. *k*NN and MLPNN have skewness close to zero meaning their errors have a normal distribution. All the models have a kurtosis value that is less than 3.

Table 5. Comparing residuals of the models' prediction.

	Median	**Min**	**Max**	**Mean**	**Std. Dev.**	**Skewness**	**kurtosis**
MLPNN	0.09	-57.31	67.73	-1.36	22.20	0.28	2.56
CNN	0.17	-77.58	30.96	-6.05	24.62	-1.31	1.68
*k*NN	0.00	-12.54	10.92	1.01	4.39	0.16	1.79

Figure 7 present the whisker and box plots of the residuals of the forecast made with the MLPNN, CNN and *k*NN models for clear sky and cloudy days during summer and winter seasons. The residual of the *k*NN model has the smallest tail compared to the others, followed by the forecast

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 May 2024 doi:10.20944/preprints202405.0490.v1

made with MLPNN although it made a worst prediction in the summer cloudy day under investigation. It also shows that the *k*NN model produced the best overall forecast.

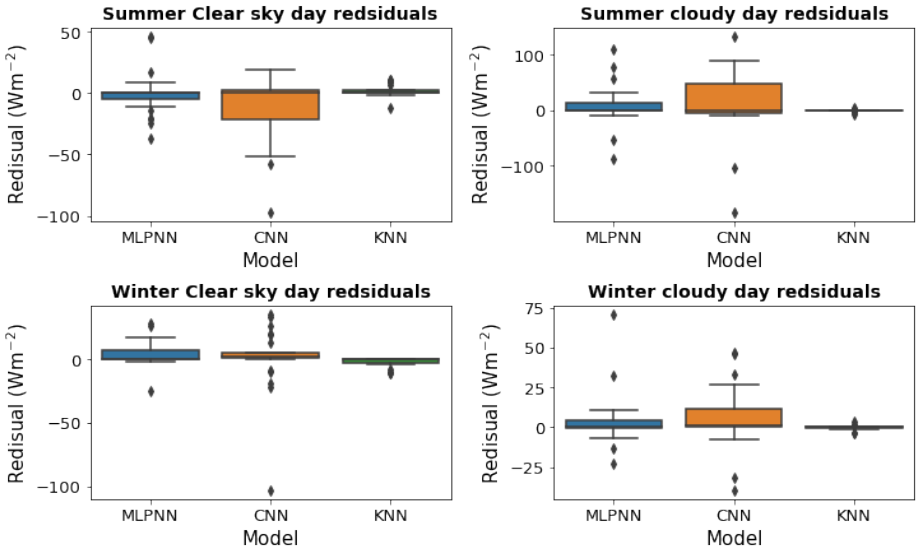


Figure 7. Whisker and box plots of the residuals of the forecast made with MLPNN, CNN and *k*NN models on clear sky and cloudy days during the summer and winter seasons.

4.2. Discussion of Results

This work has focused on modelling and forecasting solar PV output power (hourly) data for Grahamstown, Eastern Cape, South Africa. The data is from January 2009 to December 2020. Modelling this with MLPNN, CNN and *k*NN techniques, the *k*NN algorithm was found to produce the best model for this data based on RMSE, rRMSE, MAE, rMAE and R^2 score metrics. The *k*NN is the best model for our data. Note that the data under investigation has very few spikes (or outliers) and missing records (and is not too noisy), so the *k*NN model perfectly predicted the data. Again, while MLPNN and CNN each takes several minutes to train their respective model, *k*NN has no training step. It goes straight into modelling the PV. So, when it comes to execution time, *k*NN still wins the contest.

We were inspired by the works of [4,20,48]. Mutavhatsindi et al. [48] analysed the performance of support vector regression, principal component regression, feedforward neural networks and LSTM network. Ratshilengo et al. [4] indeed compared the GA algorithm with the RNN and *k*NN algorithms models' performance in forecasting global horizontal irradiance. They found the GA algorithm to have the best overall forecast performance. The *k*NN model applied in this study produced lower metric values for the *k*NN for RMSE, MAE, rRMSE and rMAE than those produced by [4].

5. Challenges of Photovoltaic Power Forecasting

Solar PV output power forecasting has a few challenges or limitations as with any other predictive analytics. One of the main limitations encountered is the accurate prediction of future weather condition when applying physical and indirect methods that require future weather parameters as input [49]. Another limitation is the issue of having a huge amount of data to be processed when applying statistical methods. Although a large amount of data allows this technique to make better prediction, processing the data consume a lot of machine resources which is also a weakness of this method. In addition, once large data is being processed, there is often a compromise on the output's speed and accuracy, particularly for generation plants where real-time data output is required.

In most statistical and hybrid approaches, it is usually considered that a complex model will produce more accurate results. This is not always a guaranteed scenario, as a simpler technique may sometimes yield higher accuracy if the input parameters are properly filtered and preprocessed. This

 doi:10.20944/preprints202405.0490.v1

poses a challenge similar to the view expressed by the authors in [50] to selecting the right model and input parameters.

Furthermore, there is the challenge of cell/module degradation and site-specific losses, which negatively impact medium and long-term power forecasting estimates. Since forecasting models are based on input weather conditions, historical data or both, the forecasted data may deviate significantly when unexpected adverse weather conditions occur. The occurrence may accelerate the degradation of the installed solar cells/modules in a manner different from the predefined constant degradation factor that has been applied in the forecasting model. Thus, even when site-specific modelling has been done, there may be a need to constantly review the model's input parameters over time based on the degradation of the solar PV modules.

6. Conclusions

This study has applied MLPNN, CNN and *k*NN methods to model solar PV output power for (solar PV installation in) Grahamstown, Eastern Cape, South Africa for a short-term forecast horizon. This study's findings will be a useful tool for energy providers (both private and public) who want quick and easy but accurate forecast of their solar photovoltaic installation - to plan energy distribution and expansion of installations in a sustainable and environmentally friendly way.

Author Contributions: Conceptualization, K.J.I.; methodology, K.J.I.; formal analysis, K.J.I.; investigation, K.J.I.; writing—original draft preparation, K.J.I.; writing—review and editing, K.J.I. All authors have read and agreed to the published version of the manuscript.

Funding: Not applicable.

Acknowledgements: We acknowledge the support of my spouse Chidinma, the University of Fort Hare and Rhodes University.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. J. R. Andrade and R. J. Bessa, Improving Renewable Energy Forecasting With a Grid of Numerical Weather Predictions:*IEEE Transactions on Sustainable Energy*, **2017,** vol. 8, pp. 1571–1580, 10.1109/TSTE.2017.2694340.
2. S. Sun, S. Wang, G. Zhang and J. Zheng, "A decomposition-clustering-ensemble learning approach for solar radiation forecasting," *Solar Energy*, 2018, vol. 163, pp. 189–199.
3. X. Yang, F. Jiang and H. Liu, "Short-term solar radiation prediction based on SVM with similar data," in - 2nd IET Renewable Power Generation Conference, **2013**, pp. 1–4.
4. M. Ratshilengo, C. Sigauke and A. Bere, "Short-Term Solar Power Forecasting Using Genetic Algorithms: An Application Using South African Data," *Applied Sciences*, May 6, **2021**, vol. 11, pp. 4214.
5. K.J. Iheanetu, "Solar Photovoltaic Power Forecasting: A Review," *Sustainability* (Basel, Switzerland), **2022**, vol. 14, Dec 19.
6. U.K. Das, K.S. Tey, M. Seyedmahmoudian, S. Mekhilef, M.Y.I. Idris, W. Van Deventer, B. Horan and A. Stojcevski, "Forecasting of photovoltaic power generation and model optimization: A review," *Renewable and Sustainable Energy Reviews*, **2018**, vol. 81, pp. 912–928.
7. J. Zhang, A. Florita, B.M. Hodge, S. Lu, H.F. Hamann, V. Banunarayanan and A.M. Brockway, "A suite of metrics for assessing the performance of solar power forecasting," *Solar Energy*, **2015**, vol. 111, pp. 157–175.
8. P. Blanc, J. Remund and L. Vallance, "Short-term solar power forecasting based on satellite images," Renewable Energy Forecasting: From Models to Applications, **2017**, pp. 179–198.
9. G. Wang, Y. Su and L. Shu, "One-day-ahead daily power forecasting of photovoltaic systems based on partial functional linear regression models," *Renewable Energy*, **2016**, vol. 96, pp. 469–478.
10. C.F.M. Coimbra K., "Overview of solar-forecasting methods and a metric for accuracy evaluation," Boston: Academic Press, **2013**, pp. 171–194.
11. A. Gensler, J. Henze, B. Sick and N. Raabe, "Deep Learning for solar power forecasting - An approach using AutoEncoder and LSTM Neural Networks," 2016 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2016 - Conference Proceedings, **2017**, pp. 2858–2865.

12. H. Wang, H. Yi, J. Peng, G. Wang, Y. Liu, H. Jiang and W. Liu, "Deterministic and probabilistic forecasting of photovoltaic power based on deep convolutional neural network," *Energy Conversion and Management*, **2017**, vol. 153, pp. 409–422.
13. M.A.F.B. Lima, P.C.M. Carvalho, A.P. de S. Braga, L.M. Fernández Ramírez and J.R. Leite, "MLP Back Propagation Artificial Neural Network for Solar Resource Forecasting in Equatorial Areas," *Renewable Energy and Power Quality Journal*, **2018**, vol. 1, pp. 175–180.
14. R.L.d.C. Costa, "Convolutional-LSTM networks and generalization in forecasting of household photovoltaic generation," *Eng Appl Artif Intell*, **2022**, vol. 116, pp. 105458.
15. G. Li, S. Xie, B. Wang, J. Xin, Y. Li and S. Du, "Photovoltaic Power Forecasting with a Hybrid Deep Learning Approach," *IEEE Access*, **2020**, vol. 8, pp. 175871–175880.
16. Y. Wang, W. Liao and Y. Chang, "Gated Recurrent Unit Network-Based Short-Term Photovoltaic Forecasting," Energies, Aug 18, **2018**, vol. 11, pp. 2163.
17. Gao, B.; X. Huang; J. Shi; Y. Tai and R. Xiao, Predicting day-ahead solar irradiance through gated recurrent unit using weather forecasting data:*Journal of Renewable and Sustainable Energy*, **2019.** vol. 11, 10.1063/1.5110223.
18. S. Tajmouati, B. EL Wahbi and M. Dakkon, "Applying regression conformal prediction with nearest neighbors to time series data," Communications in Statistics. Simulation and Computation, Mar 26, **2022**, vol. ahead-of-print, pp. 1–11.
19. M. Ratshilengo, C. Sigauke and A. Bere, "Short-Term Solar Power Forecasting Using Genetic Algorithms: An Application Using South African Data," *Applied Sciences*, **2021** Vol.11, Page 4214, vol. 11, pp. 4214.
20. M.A. Reyes-Belmonte, "Quo Vadis Solar Energy Research?" *Applied Sciences*, Mar 28, **2021**, vol. 11, pp. 3015.
21. V. Cherkassky and Y. Ma, "Practical selection of SVM parameters and noise estimation for SVM regression," Neural Networks, **2004**, vol. 17, pp. 113–126.
22. J. Huang, M. Korolkiewicz, M. Agrawal and J. Boland, "Forecasting solar radiation on an hourly time scale using a Coupled AutoRegressive and Dynamical System (CARDS) model," *Solar Energy*, **2013**, vol. 87, pp. 136–149.
23. A. El Hendouzi, A. Bourouhou and O. Ansari, "The Importance of Distance between Photovoltaic Power Stations for Clear Accuracy of Short-Term Photovoltaic Power Forecasting," Journal of Electrical and Computer Engineering, **2020**.
24. X. Luo, D. Zhang and X. Zhu, "Deep learning based forecasting of photovoltaic power generation by incorporating domain knowledge," *Energy*, **2021**, vol. 225, pp. 120240.
25. H. Zhu, X. Li, Q. Sun, L. Nie, J. Yao and G. Zhao, "A Power Prediction Method for Photovoltaic Power Plant Based on Wavelet Decomposition and Artificial Neural Networks," *Energies*, **2016**, Vol.9, Page 11, vol. 9.
26. R.L.d.C. Costa, "Convolutional-LSTM networks and generalization in forecasting of household photovoltaic generation," Engineering Applications of Artificial Intelligence, Nov. **2022**, vol. 116, pp. 105458.
27. Q. Xu, D. He, N. Zhang, C. Kang, Q. Xia, J. Bai and J. Huang, "A short-term wind power forecasting approach with adjustment of numerical weather prediction input by data mining," IEEE Transactions on Sustainable Energy, **2015**, vol. 6, pp. 1283–1291.
28. F. Aminzadeh and Paul De Groot, "Neural networks and other soft computing techniques with applications in the oil industry," Eage *Publications*, **2006**.
29. M.S. Hossain, Z.C. Ong, Z. Ismail, S. Noroozi and S.Y. Khoo, "Artificial neural networks for vibration based inverse parametric identifications: A review," *Applied Soft Computing*, **2017**, vol. 52, pp. 203–219.
30. Y. Zhang, G.P. Chen, O.P. Malik and G.S. Hope, "An Artificial Neural Network Based Adaptive Power System Stabilizer, " *IEEE Trans.Energy Convers.*, **1993**, vol. 8, pp. 71–77.
31. M.O. Moreira, P.P. Balestrassi, A.P. Paiva, P.F. Ribeiro and B.D. Bonatto, "Design of experiments using artificial neural network ensemble for photovoltaic generation forecasting," Renewable and Sustainable Energy Reviews, 2021, vol. 135, pp. 110450.
32. H.A. Malki, N.B. Karayiannis and M. Balasubramanian, "Short-term electric power load forecasting using feedforward neural networks," *Expert Syst*, **2004**., vol. 21, pp. 157–167.
33. S.M. Chen, Y.C. Chang, Z.J. Chen and C.L. Chen, "MULTIPLE FUZZY RULES INTERPOLATION WITH WEIGHTED ANTECEDENT VARIABLES IN SPARSE FUZZY RULE-BASED SYSTEMS," Http://Dx.Doi.Org/10.1142/S0218001413590027, **2013**, vol. 27.
34. A. Yona, T. Senjyu, T. Funabashi and C.H. Kim, "Determination method of insolation prediction with fuzzy and applying neural network for long-term ahead PV power output correction," *IEEE Transactions on Sustainable Energy*, **2013**, vol. 4, pp. 527–533.
35. P. Srisaeng, G.S. Baxter and G. Wild, "An adaptive neuro-fuzzy inference system for forecasting Australia's domestic low cost carrier passenger demand," Vilnius Gediminas Technical University, **2015**, vol. 19, pp. 150–163.
36. M.N. Ali, K. Mahmoud, M. Lehtonen, M.D. "An efficient fuzzy-logic based variable-step incremental conductance MPPT method for grid-connected PV systems", *Access*, **2021**, Ieeexplore.Ieee.Org, .

37. J. Zhang, R. Verschae, S. Nobuhara and J.F. Lalonde, "Deep photovoltaic nowcasting," *Solar Energy*, **2018**, vol. 176, pp. 267–276.
38. I. Parvez, A. Sarwat, A. Debnath, T. Olowu, M. G. Dastgir and H. Riggs, "Multi-layer Perceptron based Photovoltaic Forecasting for Rooftop PV Applications in Smart Grid," in - 2020 Southeast Con, **2020**, pp. 1–6.
39. L. Hontoria, J. Aguilera and P. Zufiria, "Generation of hourly irradiation synthetic series using the neural network multilayer perceptron," *Solar Energy*, **2002**, vol. 72, pp. 441–446.
40. B.T. Pham, D. Tien Bui, I. Prakash and M.B. Dholakia, "Hybrid integration of Multilayer Perceptron Neural Networks and machine learning ensembles for landslide susceptibility assessment at Himalayan area (India) using GIS," *Catena*, **2017**, vol. 149, pp. 52–63.
41. I. Parvez, M.G.S. Sriyananda, İ Güvenç, M. Bennis and A. Sarwat, "CBRS Spectrum Sharing between LTE-U and WiFi: A Multiarmed Bandit Approach," *Mobile Information Systems*, Aug 29, 2016, vol. 2016, pp. 1–12.
42. Horton, P.; Y. Mukai and K. Nakai, .PROTEIN SUBCELLULAR LOCALIZATION PREDICTION:*The Practical Bioinformatician,* **2004.**pp. 193, -05. 10.1142/9789812562340_0009.
43. Zhao Liu and Ziang Zhang, "Solar forecasting by K-Nearest Neighbors method with weather classification and physical model," Sep **2016,** pp. 1–6.
44. Kohavi, R. and G.H. John, .Wrappers for feature subset selection:*Artificial Intelligence,* **1995.** vol. 97, pp. 273, 10.1016/s0004-3702(97)00043-x.
45. C. Chatfield, "Calculating Interval Forecasts," Journal of Business & Economic Statistics, vol. 11, pp. 121–135, Apr 01.
46. A. Gaba, I. Tsetlin and R.L. Winkler, "Combining Interval Forecasts," Decision Analysis, **1993**, vol. 14, pp. 1–20, Mar. 2017.
47. X. Sun, Z. Wang and J. Hu, "Prediction Interval Construction for Byproduct Gas Flow Forecasting Using Optimized Twin Extreme Learning Machine," Mathematical Problems in Engineering, **2017**, pp. 1–12, Aug 23.
48. T. Mutavhatsindi; C. Sigauke and R. Mbuvha, Forecasting Hourly Global Horizontal Solar Irradiance in South Africa Using Machine Learning Models:*IEEE Access,* **2020.** vol. 8, pp. 198872–198885, 10.1109/ACCESS.2020.3034690.stylefixSengupta, Manajit, Habte, Aron, Wilbert, Stefan, Gueymard, Christian,and Remund, Jan, "Best Practices Handbook for the Collection and Use of Solar Resource Data for Solar Energy Applications: Third Edition," **2021**.
49. A. Dolara, F. Grimaccia, S. Leva, M. Mussetta and E. Ogliari, "A Physical Hybrid Artificial Neural Network for Short Term Forecasting of PV Plant Power Output", **2015** vol. 8, pp. 1138–1153.