Solar Energy 254 (2023) 88–101

Contents lists available at ScienceDirect

Solar Energy

journal homepage: www.elsevier.com/locate/solener

Machine Learning models for the estimation of the production of large utility-scale photovoltaic plants

Ana P. Talayero, Julio J. Melero *, Andrés Llombart, Nurseda Y. Yürüşen

Instituto Universitario de Investigación Mixto CIRCE (Fundación CIRCE - Universidad de Zaragoza), C/ Mariano Esquillor 15, 50018, Zaragoza, Spain

ARTICLE INFO

Keywords:
Photovoltaics
Utility-scale PV plants
Hyperparameters optimisation
Machine Learning models
PV power estimation

ABSTRACT

Photovoltaic (PV) energy development has increased in the last years mainly based on large utility-scale plants. These plants are characterised by a huge number of panels connected to high-power inverters occupying a large land area. An accurate estimation of the power production of the PV plants is needed for failure detection, identifying production deviations, and the integration of the plants into the power grid. Various studies have used Machine Learning estimation techniques developed on very small PV plants. This paper deals with large utility-scale plants and uses all the available information to represent the non-uniform radiation over the whole studied solar field. Variables measured in up to four meteorological stations and distributed across the plant are used. Three PV plants with 1, 2 and 4 meteorological stations have been used to develop Machine Learning models. The hyperparameters were systematically optimised, demonstrating the improvements by comparing with a simple model based on Multiple Linear Regression. The best results were obtained with the Random Forest technique for the three PV plants, providing a RMS error value ranging from 1.9% to 5.4%. The final models were compared with those found in the literature for tiny PV plants showing in general much better performance.

1. Introduction

Solar photovoltaic (PV) is an ever-expanding technology, with an annual growth rate in recent years of more than 20% (International Renewable Energy Agency - IRENA, 2021). The global PV capacity at the end of 2020 was 714 GW (International Renewable Energy Agency - IRENA, 2021), and this figure will be doubled in the next five years. Its growth potential and lower generation costs will enable PV to become the most competitive energy source globally in the coming years.

The modularity of this technology allows an easy and quick installation of different plant sizes, from a few watts for self-consumption to hundreds of megawatts for large utility-scale grid-connected plants. Large utility-scale PV plants are characterised by a huge number of panels (hundreds of thousands) connected to high-power inverters (megawatts of power) and occupying a large land area (tens of hectares). Its size makes finding malfunctioning parts more complicated, and the time to do that can be considerable. So, energy losses due to failures and unavailability can become significant in large plants.

The availability and energy losses of the plants have been evaluated through reliability studies (Spertino et al., 2021b,a; Ketjoy et al., 2021), analysing the root causes of failures, and finding that inverters are responsible for the highest losses and unavailability. It was also concluded that availability in large plants was better due to advantageous maintenance contracts. Energy losses in utility-scale PV plants have also been estimated using performance analysis (Bansal et al., 2022; Dahmoun et al., 2021; Jed et al., 2021). Different climatic zones provided PR values ranging from 70% in hot climate (India) to 87% in cold climate (France) with yearly degradation rates varying from 0.2% to around 1%. It can be concluded that accurate knowledge of the photovoltaic production is essential not only to determine the performance of the plants and its evolution in time but also to characterise energy losses associated with possible component failures and to ensure the integration of the plants into the power grid. Energy production can be obtained from actual measurements, made with the plant equipment or specific monitoring systems (Beránek et al., 2018), or from simulation models that accurately reproduce the plant's behaviour in real conditions. Combining the two methods allows a more accurate estimation of losses than the obtained in performance assessment methods using standard conditions (Bansal et al., 2022; Jed et al., 2021).

The estimation of the power production of a PV plant can be done using parametric and non-parametric models. In both cases, the model

* Corresponding author.
E-mail addresses: aptalayero@fcirce.es (A.P. Talayero), julio.melero@unizar.es (J.J. Melero), allombart@fcirce.es (A. Llombart), 730812@unizar.es (N.Y. Yürüşen).

https://doi.org/10.1016/j.solener.2023.03.007
Received 28 September 2022; Received in revised form 26 January 2023; Accepted 3 March 2023
Available online 15 March 2023

is fed with several inputs, solar irradiation and temperature, for example, providing one output, AC or DC power (Platon et al., 2012; Almeida et al., 2015). The parametric models are based on physical properties of the system represented with analytical equations (Sabbaghpur Arani and Hejazi, 2016). Non-parametric models consider the photovoltaic plant as a black box (Almeida et al., 2015). They are numerical models that learn the plant's behaviour from historical time series data. As a result, these models relate the power generated to other parameters, such as climatological variables, electrical measurements, the temperature of the PV modules, etc. Garoudja et al. (2017), Mellit and Pavan (2010), and even can identify the energy losses associated with faults and shadows (Øgaard et al., 2021).

The techniques applied to model developing range from statistical (Fazai et al., 2019; Nobre et al., 2016) and control charts (Øgaard et al., 2021) to the most advanced machine learning (ML) algorithms (Mellit et al., 2018; Li et al., 2021; Elsheikh et al., 2019), being Artificial Neural Networks (ANNs) the most widely used (Daliento et al., 2017). Recent studies focused on improving the prediction results by combining ANNs with other ML techniques (Theocharides et al., 2020; Moreira et al., 2021). Nevertheless, comparing the performance of the ANN models to other techniques may be challenging due to different data sources, distinct time steps in the time series, and the diverse scales and metrics used. Some papers compared ANNs with other methods such as Multiple Linear Regression (MLR), Support Vector Machines (SVM) and Gaussian Process Regression (GPR) methods (Monteiro et al., 2017; Graditi et al., 2016; Trigo-Gonzalez et al., 2021; Huang et al., 2016), showing ANNs as the most suitable. However, some advanced algorithms found in the literature with promising results, such as the tree-based ones, Random Forests (RF) and Gradient Boosting (GB) (Almeida et al., 2015; AlSkaif et al., 2020), have not been compared.

The proliferation of ML models and their application to PV power estimation has been possible due to the intensive development of ML algorithms in recent years. However, the good use of these algorithms requires optimising their hyperparameters as they can greatly impact the prediction performance. Most of the reviewed documents lack a good reason or justification for choosing the hyperparameters, and only a few papers implemented systematic techniques to determine them (Pan and Tan, 2019; Trigo-Gonzalez et al., 2021; Theocharides et al., 2020; Pan et al., 2020; Piliougine and Spagnuolo, 2022). Given these recent developments, it is clear that ML algorithms require a methodology to optimise the hyperparameters, thus improving the models' performance.

The use of ML to analyse PV systems is still in its infancy and it is being applied to many small-scale technologies (Sohani et al., 2022). When dealing with utility-scale PV installations, their large size has to be considered due to the possible influence in the energy output of the variations of terrain, clouds and shadows across the entire plant. Most of the developed models mentioned above use simulated or experimental data from very small PV installations, varying from some panels (Monteiro et al., 2017) to several tens of kilowatts (Moreira et al., 2021). Up to the authors knowledge, there is only one paper forecasting the production of large utility-scale PV plants with nominal power varying from 958 kWp to 2640 kWp (Almeida et al., 2015). Nevertheless, they used spatial variability indexes calculated with a 12 × 12 km grid that considerably exceeded the plants dimensions. Then, it appears clear that the size of these large plants is a factor that have to be considered when trying to predict their power production.

1.1. Aim of the work

This work aims to accurately estimate the power production of large utility-scale photovoltaic plants. Different ML algorithms have been applied to find the most suitable technique and search for the optimal combination of input variables. Multiple Linear Regression (MLR), Random Forests (RF), Gradient Boosting (GB) and Artificial Neural Networks (ANN) techniques were used with data from three large utility-scale PV plants. Models hyperparameters were systematically optimised. The size of the plants were considered using input variables taken from several meteorological stations distributed across the installations.

The paper is organised as follows; the second section describes the data and the PV plants; next section explains the methodology to adjust the models, the fourth section present the results and compare the different models and the last section compare the results with other literature models. Finally, the summary of the results is presented in the conclusions.

Table 1
Summary of characteristics of the PV plants.

	Installed power	Surface	Terrain type	Altitude	Met stations
PV1	15 MW	0.3 Km^2	Hilly with bushes	Low	1
PV2	30 MW	1.5 Km^2	Flat and clean	Low	2
PV3	100 MW	3 Km^2	Mountain	High	4

2. PV plants data

Data were obtained from three utility-scale fixed-tilt PV plants. They were chosen to cover a range of scenarios regarding topography, latitude, and total installed power. The three plants share the same configuration using centralised inverters with nominal power exceeding 1 MW per unit. Multi-crystalline Si (mc-Si) solar modules with a tilt angle ranging from 10° to 20° are installed. The overall characteristics of the three plants are given in Table 1. More specific details are confidential and cannot be provided.

Apart from topography and location, the main difference between the three plants is their maximum power and thus their total surface area. Larger plants may need more weather stations to reflect the weather conditions. The plant planners use to take this into account by installing more weather stations as the size of the plant increases, as can be seen in Table 1.

The size of the studied PV plants can make data manipulation and the hyperparameters search process unaffordable. Data from only one of the inverters of each plant, accounting for a power value equal to 1 MWp each one, are used in this work. This size is big enough to be representative of the solar field of large PV plants. In contrast, meteorological data are taken from all available weather stations located across the plant to consider the spatial distribution of the solar field and its effect on power production. Total AC power is acquired from the inverter, whereas tilted solar radiation, ambient temperature and module temperature are obtained from the met stations. Fig. 1 shows the layout of the studied inverter including distances and approximate profiles.

Data were measured at a frequency of 1 min, and the average values were recorded every 10 min. Collected data periods and measured variables are shown in Table 2.

3. Methodology and theoretical basis

This section describes the proposed methodology for selecting the optimal PV output estimation model as a function of solar radiation, ambient temperature, and module temperature. ML models should be tuned by optimising their hyperparameters before fitting the definitive models. Hyperparameters are parameters whose values control the learning process and determine the values of model parameters that a learning algorithm ends up learning. Then, it is crucial their fine tuning in order to have good estimation results from the models.

Fig. 2 shows the schematic diagram of the proposed methodology for obtaining the PV production from the measured variables.

The algorithms were coded using the open source R programming language (R. Core Team, 2021) and libraries.

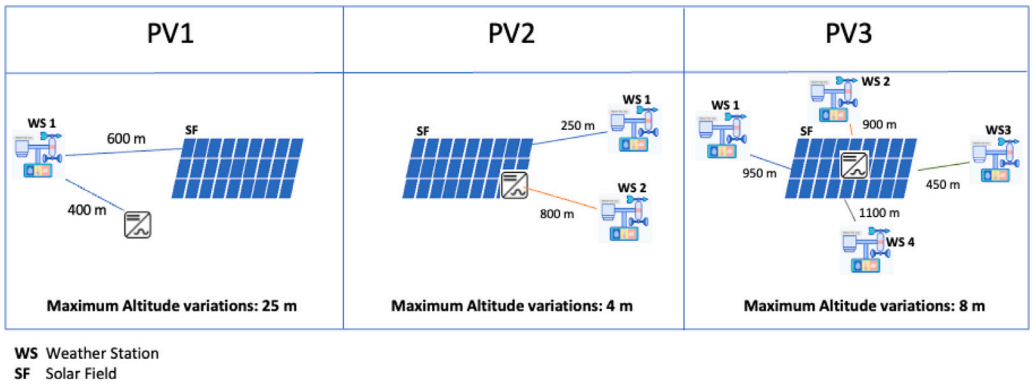


Fig. 1. Schematic layout with the position of the studied inverter and the distances between the solar field, the inverter and the weather stations for each plant.

Fig. 2. Schematic diagram of the methodology.

3.1. Data preprocessing

Data pre-processing consists of three steps, filtering, scaling and partitioning the data. Data filtering controls the presence of wrong measurements or outliers which can cause errors or add uncertainties in the developed models. Scaling is needed in order to have a homogeneous range of variation in the used features. Normalisation procedure is used here according to Eq. (1).

$$x' = \frac{x - min(x)}{max(x) - min(x)} \tag{1}$$

Data partitioning is performed in two steps. First, data are divided into training and test subsets. Then, the training set is further divided into a new training and validation subsets to use the train subset to fit the model parameters and the validation subset to optimise the hyperparameters and fine tune the models using unbiased data. The test subset has been used to evaluate the performance of the final models. The common partition scheme 80/20% was used in both steps.

3.2. Theoretical basis of MLR, RF, GB and ANN

One statistical regressor, MLR, and three automatic machine learning techniques are proposed in this work. They are briefly described in the next subsections.

3.2.1. Multiple linear regression

Multiple linear regression (MLR), is a statistical technique with the purpose of seeking for the linear relationship between a dependent variable (response) and several independent or explanatory variables (features). In essence, multiple regression is the extension of ordinary least-squares (OLS) regression because it involves more than one explanatory variable. MLR model is typically described by Eq. (2),

$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \cdots + \beta_k x_{ki} + \epsilon_i \tag{2}$$

where y_i is the dependent variable, $\beta_0, \beta_1, \beta_2, \ldots, \beta_k$ are the regression coefficients, and ϵ_i's are the errors. The regression coefficients are estimated using the least squares principle, while the error term is usually assumed to be normally distributed with a constant variance σ^2.

Table 2
Information on the studied periods, variables used in the models and weather stations from which the variables were extracted.

	Data period	Records	Variables	Weather stations
PV1	Feb 17–Oct 19 (32 months)	104178	AC Power Radiation Ambient temperature Module temperature	 WS1 WS1 WS1
PV2	Jul 17–Oct 19 (27 months)	122832	AC Power Radiation Ambient temperature Module temperature	 WS1 to WS2 WS1 to WS2 WS1 to WS2
PV3	Mar 19–Aug 21 (29 months)	126288	AC Power Radiation Ambient temperature Module temperature	 WS1 to WS4 WS1 to WS4 WS1 to WS4

3.2.2. Random forests

The Random Forest (RF) algorithm (Breiman, 2001) is a supervised nonlinear technique. It uses ensemble learning for regression, which combines estimations from multiple machine learning algorithms to make a more accurate estimation than a single model. In this case, it uses many decision trees. The estimation is obtained by taking the average or mean of the output from all the trees built. The schematic procedure of the algorithm is shown in Fig. 3. The average of the estimations is more accurate than that of any individual tree, and increasing the number of trees increases the precision of the outcome.

The results obtained with the RF algorithm depend on the model hyperparameters, the number of trees (*Ntrees*), the trees *Depth*, the number of variables randomly considered in each split (*Mtries*) and the sampling rate (*SR*), used to speed up the training process.

3.2.3. Gradient Boosting

Gradient Boosting is a machine learning technique that, when used in regression tasks, provides an estimation in the form of an ensemble of weak estimations, using decision trees similarly to RF (Friedman, 2002). The main difference between RF and GB is that GB builds one tree at a time combining the results along the way with the use of

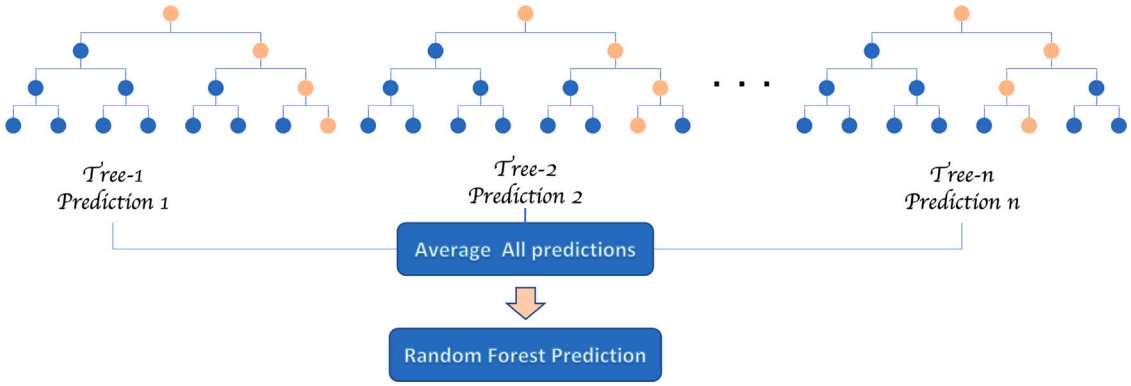


Fig. 3. Schematics of Random Forest algorithm.

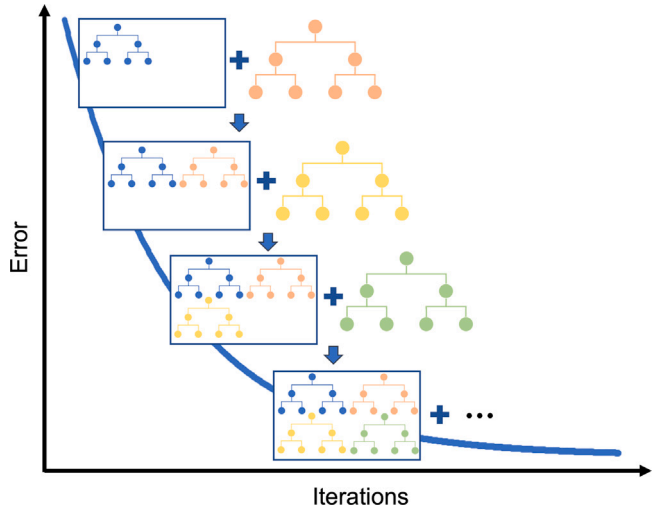


Fig. 4. Gradient Boosting process of adding trees minimising loss function.

a loss function. The final estimation is the sum of estimations of all individual trees. Then, the model is obtained in an iterative process where each tree is built with the estimation errors of the previous one in order to minimise a loss function, as can be seen in Fig. 4. The tuning hyperparameters are the number of trees *Ntrees*, their *Depth* and the sampling rate (*SR*).

3.2.4. Artificial Neural Networks

Artificial Neural Networks are a series of algorithms that try to mimic the way that human brain works. Individual neurons are interconnected with other neurons to recognise relationships in data sets. A neural network contains layers of interconnected nodes. Each node (or neuron) in the net is named Perceptron (Goodfellow et al., 2016). Here, the type of Neural Networks named Multilayer Perceptron (MLP) is used. MLPs consist of an input layer and an output layer stacked with one or more hidden layers in between. Every unit in a layer is connected with all the units in the previous layer. These connections are not all the same as a weight is applied to each one. All inputs to the layer are combined with their weights in a weighted sum and subjected to the activation function, feeding the next layer and repeating the procedure until reaching the output layer (feedforward process). The value of a metric is forwarded in the process and used to obtain its gradient between all input and output pairs in each layer. Then, it is propagated back allowing updating the weights used in each layer with the calculated gradient (backpropagation). The whole process is iterated until convergence in the gradient is reached. Fig. 5 shows the working principles of the MLP.

ANNs and, in particular, MLPs have a wide range of hyperparameters that can be tuned to optimise the models. Here, the chosen hyperparameters to be tuned are the number of hidden layers, the number of neurons in hidden layer j($units_j$), the learning rate (*LR*), the regularisation applied to layer j ($dropout_j$) and the number of *epochs*.

3.3. Methods for hyperparameters optimisation

Hyperparameters in Machine Learning can be thought of as the tuning knobs of the developed models. In this section, an introduction to Hyperparameter optimisation (HPO) is given.

The most popular and simple HPO techniques are Grid Search and Random Search, which consist of exploring the hyperparameters space to find the best combination based on the model metrics (Hutter et al., 2019). Recently, modern bandit-based strategies, such as Hyperband have come into play (Li et al., 2018) to improve hyperparameter optimisation.

Grid Search is the simplest HPO method where the user specifies a finite set of values for each hyperparameter, and grid search evaluates the Cartesian product of these sets. The problem is that the number of function evaluations grows exponentially with the dimensionality of the configuration space. A simple alternative to grid search is Random Search which samples configurations at random until a specific budget for the search is exhausted. Additionally, HyperBand search is a strategy which divides the total computational budget into several combinations of number of configurations versus the budget of each, and then calls successive halving as a subroutine on each set of random configurations. It applies a hedging strategy that includes running some configurations only with the maximum budget. In the worst case, HyperBand takes at most a constant factor more time than vanilla random search on the maximum budget. HyperBand has been shown to improve upon conventional methods on some ML problems (Li et al., 2018).

In this work, the above described search algorithms have been used depending on the number of parameters to tune and the computation resources needed.

3.4. Metrics for model evaluation

The metrics used in this work are the normalised Mean Absolute Error ($nMAE$), the normalised Root Mean Squared Error ($nRMSE$), and the normalised bias ($nBIAS$), widely accepted in the literature (Kumari and Toshniwal, 2021).

The Mean Absolute Error is the absolute mean of the difference between the expected value of the estimator and the actual value of the parameter being estimated. It represents the mean value of the absolute errors in a regression model and is calculated according Eq. (3).

$$nMAE = 100 \times \frac{1}{N} \sum_{i=1}^{N} |\hat{y}_i - y_i| \tag{3}$$

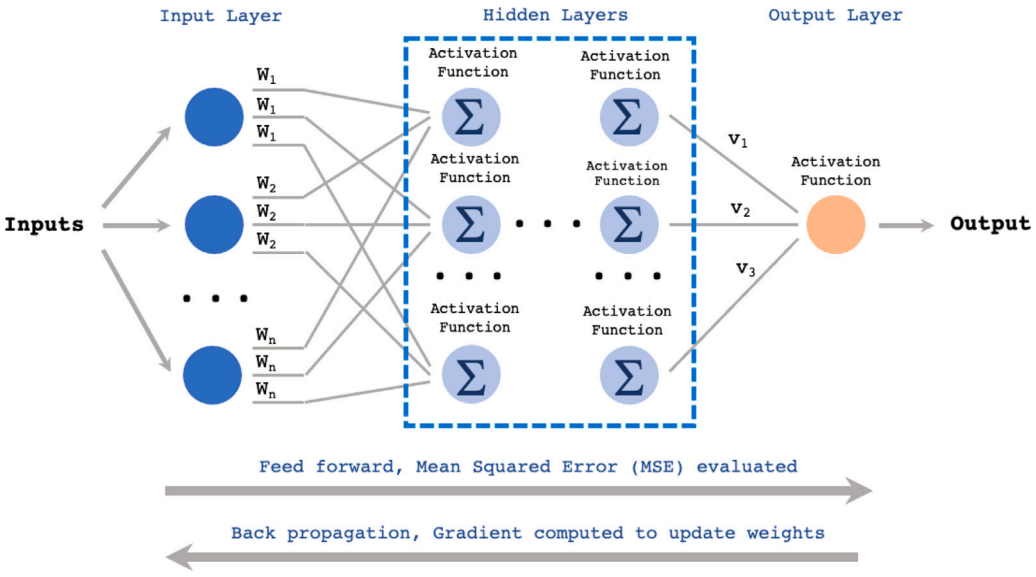


Fig. 5. Working principles of the Multilayer Perceptron.

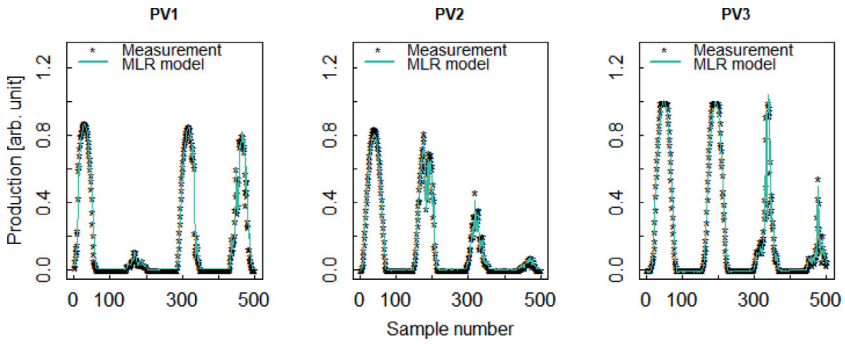


Fig. 6. MLR estimation Power compared with measured power for a sampled dataset of PV1, PV2 and PV3.

where $\hat{y}_i$ and y_i are the estimated and actual normalised values, respectively, of the ith parameter sample.

The Bias is the mean of the difference between the expected value of the estimator and the actual value of the parameter being estimated. It is a measure of the models accuracy and represents the systematic error between the estimated value and actual value and is calculated with Eq. (4).

$$nBIAS = 100 \times \frac{1}{N} \sum_{i=1}^{N} (\hat{y}_i - y_i) \tag{4}$$

The Root Mean Squared Error is the root square of the mean of the squared differences between the expected value of the estimator and the actual value of the parameter being estimated. This metric can be used to identify the outliers of the mean tendency of the model. $nRMSE$ can be obtained following Eq. (5).

$$nRMSE = 100 \times \sqrt{\frac{1}{N} \sum_{i=1}^{n} (\hat{y}_i - y_i)^2} \tag{5}$$

In addition, to better visualise the obtained improvements, the performance of the models is compared with a reference using the skill score (SS), which describes the accuracy of the models' output regarding a baseline model. $SS = 100\%$ indicates a perfect estimation, while $SS = 0\%$ means that the model equals the baseline model and there is no improvement.

$$SS = 100 \times \left(1 - \frac{nRMSE_{predicted}}{nRMSE_{baseline}}\right) \tag{6}$$

Table 3
Metrics for the baseline models of the PV plants.

	nRMSE	nMAE	nBiass
PV1	5.75%	3.02%	0.68%
PV2	2.70%	1.75%	0.26%
PV3	4.91%	3.1%2	0.98%

4. Results

This section presents the results of the models developed for the three PV studied plants including the baseline models, the hyperparameter optimisation and the final results of the models.

4.1. Baseline models

A baseline is a simple model that provides reasonable results without requiring much expertise or time to develop. Baseline models provide a sanity check against improvements and a potential basis for enhancements. Here, Multiple Linear Regression models are the chosen baselines.

The MLR algorithm has been applied estimating the power production of one inverter in each plant. Fig. 6 shows the Power estimation with MLR compared to the Measured Power for a sample dataset of each of the three PV plants.

Table 3 shows the metrics obtained for the reference models, revealing that the model performance is better for simpler terrain plants, such as PV2 (see PV plant terrain types and model inputs in Section 2).

Table 4
Hyperparameters values used in the Grid Search for RF. Where not indicated, values apply for the three plants.

	First sweep			Second sweep		
	Min	Max	Delta	Min	Max	Delta
Depth	5	50	5	5 (PV1) 15 (PV2) 25 (PV3)	15 (PV1) 35 (PV2) 40 (PV3)	1
Mtries	2	3 (PV1) 6 (PV2) 12 (PV3)	1	2 (PV1) 2 (PV2) 4 (PV3)		
SR	0.1	1	0.1	0.6 (PV1) 0.8 (PV2) 0.8 (PV3)	1	0.1
Ntrees	0	1500	10	0	1500	10
Models tested	30000 (PV1) 60000 (PV2) 165000 (PV3)			7500 (PV1) 9000 (PV2) 6750 (PV3)		

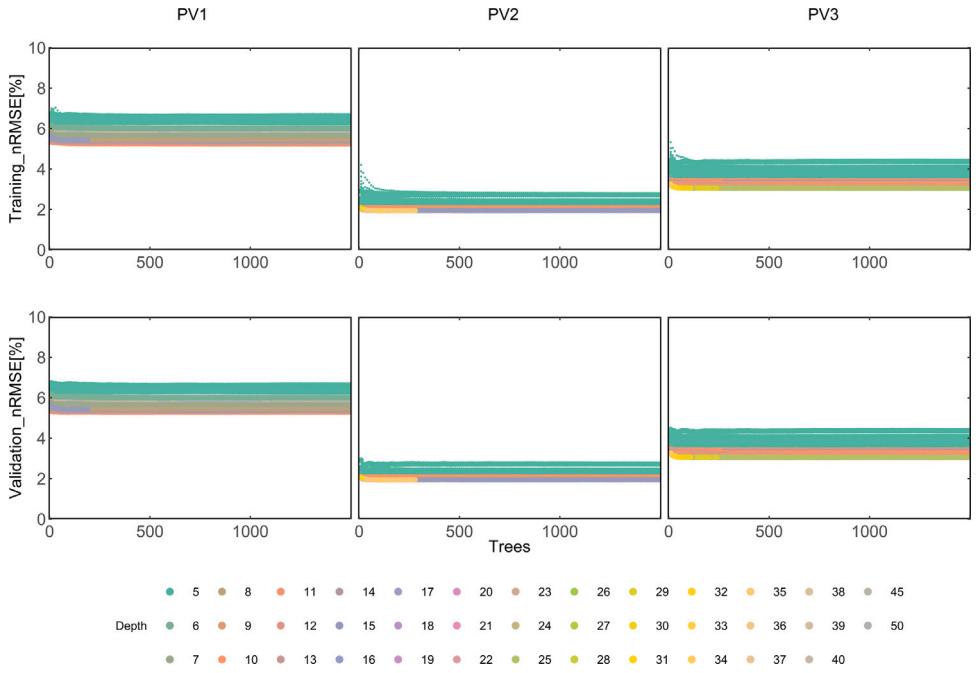


Fig. 7. RF Hyperparemeters search results.

4.2. Hyperparameters optimisation

This section describes the process of hyperparameters optimisation. The improvements of the search process were validated using the nRMSE metric. The Grid Search algorithm was applied using the H2O library (H2O.ai, 2020), while the Hyperband and the Random Search, were coded using the kerastuner package (Abdullayev, 2022). Both libraries allowed parallelisation on the Ryzen 9 processor used for the calculations.

4.2.1. Random Forests hyperparameters optimisation

The algorithm used for the RF hyperparameters optimisation was the Grid Search. The search space was constructed using four variables, *Depth*, *Mtries*, *SR*, *Ntrees*. Two sweeps were performed, using the second one for fine-tuning. Table 4 shows the summary of the values used search process.

Fig. 7 shows the nRMSE errors obtained with the training and validation subsets. The influence of each hyperparameter on the error, which allows its selection, is studied below.

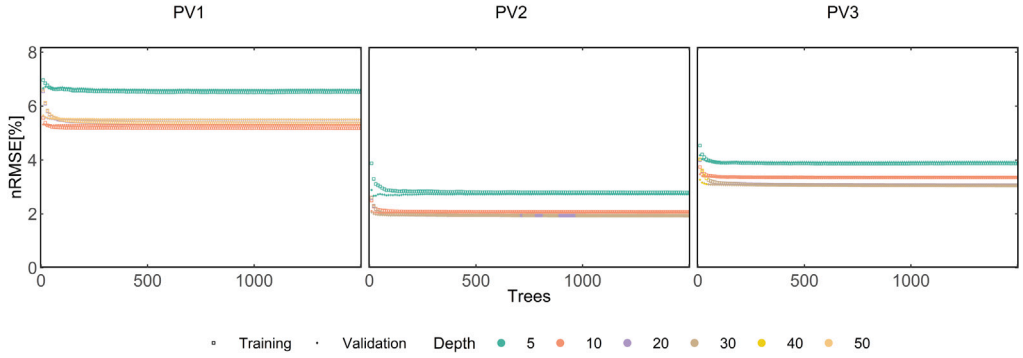
For the *Mtries* hyperparameter, the best values obtained were *Mtries* = 2 for PV1 & PV2 and *Mtries* = 4 for PV3, which coincide with the approximate value of the square root of the number of variables involved in the model, the rule recommended by the used framework.

The variation of the error obtained with the training and validation periods for different *Depth* values, keeping constant *Mtries* and *SR*, is plotted in Fig. 8(a). There is not overfitting, as the values with both periods are similar. It is also observed that, as *Depth* increases, the nRMSE decreases. Then, the optimal *Depth* values were chosen to be small enough at which the error would no longer decrease no matter how much we increased the value of *Depth*.

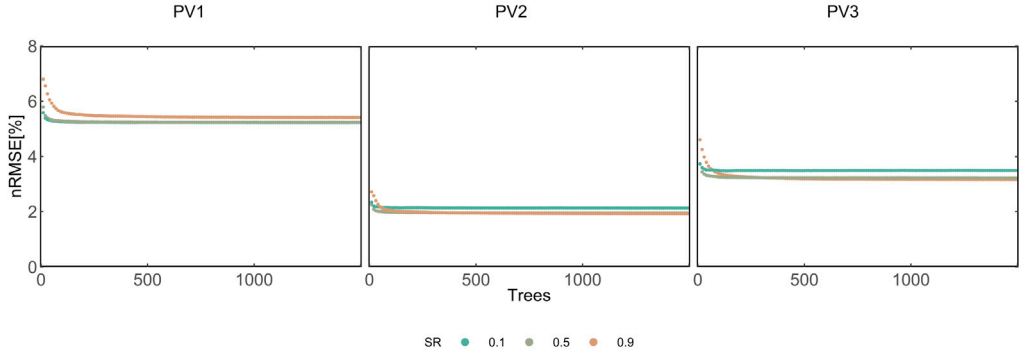
Fig. 8(b) shows that the influence of the sampling rate, *SR*, on the nRMSE is very small in all the plants. It would be expected to obtain better results for higher *SR* values as this parameter is used to reduce the number of input data and speed up the training process.

Finally, the number of trees, *Ntrees* was shown with an asymptotic behaviour in the study of each of the previous hyperparameters. The error decreases as the number of trees increases. Then, the optimal number of trees will be the minimum value that does not improve the error.

A set of hyperparameter values providing minimum errors were selected with the previous analysis. Table 5 shows the summary of results for the hyperparameters search of the RF models.



(a) Influence of $Depth$ ($Mtries = 2$ for PV1 & PV2, $Mtries = 4$ for PV3, $SR = 0.8$).



(b) Influence of SR ($Mtries = 2$ for PV1 & PV2, $Mtries = 4$ for PV3, $Depth = 20$).

Fig. 8. RF Hyperparemeters search results. Influence of *Depth* and *SR*.

Table 5
Selected hyperparameters and metrics for RF.

	PV1	PV2	PV3
Mtries	2	2	4
Depth	10	20	20
SR	0.7	0.7	0.7
Ntrees	560	520	560
nRMSE	5.2%	1.9%	3.1%
Models tested	37500	69000	172250

Table 6
Hyperparameters values used in the Grid Search for GB for each PV plant.

	Min	Max	Delta
Depth	1	15	1
SR	0.1	1	0.1
Ntrees	0	3000	5
Models tested	90000		

4.2.2. Gradient Boosting hyperparameters optimisation

The Grid Search algorithm was also used to look for the optimal hyperparameters of the GB models. Here, only three variables were used to construct the search space, *Depth*, *SR* and *Ntrees*. The search process was performed in one sweep for each plant using the same values for each plant. The values are shown in Table 6.

Fig. 9 plots the nRMSE against the sampled hyperparameters summarising the whole set of models tested for the three plants. The increase of *Depth* decreases the error but produces overfitting from a specific value, as can be seen clearly in the case of PV1 and more faintly for PV2. Then, care has to be applied to avoid the overfitting effect when selecting the optimal values.

The dependence of the nRMSE with the trees *Depth* is not so strong as it was in the case of RF, as shown in Fig. 10(a). Moreover increasing its value produces overfitting, as commented before. Then, the optimal *Depth* values were chosen small enough so that the error would be lower and avoid overfitting.

The influence of the *SR* on the nRMSE, shown in Fig. 10(b), is very small in all the plants and the optimal value was chosen belonging the middle of the variation range. Finally, The number of trees, *Ntrees* has an asymptotic behaviour, as in RF and, for this reason, the optimal number of trees was chosen as the minimum value that does not improve the error.

The set of optimal hyperparameters selected with the previous analysis are shown in Table 7

Table 7
Selected hyperparameters and metrics for GB.

	PV1	PV2	PV3
Depth	2	4	3
SR	0.4	0.4	0.7
Ntrees	2430	1280	1280
nRMSE	5.2%	1.5%	2.7%
Models tested	90000	90000	90000

4.2.3. Artificial Neural Networks hyperparameters optimisation

Neural networks, in particular MLPs, have a wider range of hyperparameters to optimise. Among them, the number of hidden layers, the number of neurons (*Units*) in each hidden layer, the dropout regularisation (*Dropout*) in each layer, the number of epochs (*Epochs*) and the learning rate (*LR*) were selected in this work. After some initial attempts varying the number of epochs, the learning rate and the number of neurons in each hidden layer, it was decided to explore only the number of neurons (*Units*) and the dropout regularisation (*Dropout*) in each hidden layer, keeping the learning rate fixed. The number of epochs was not explicitly explored as it is done automatically

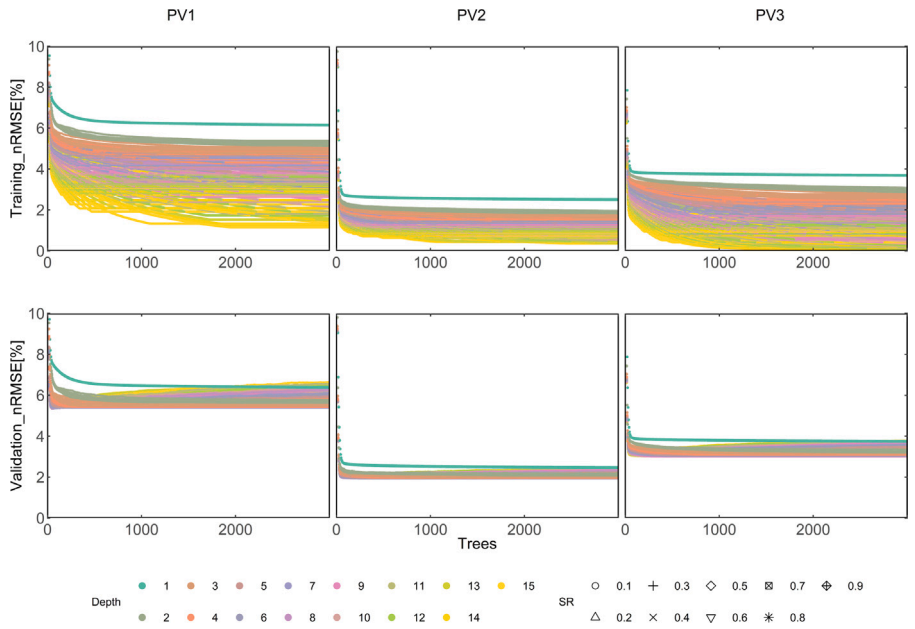
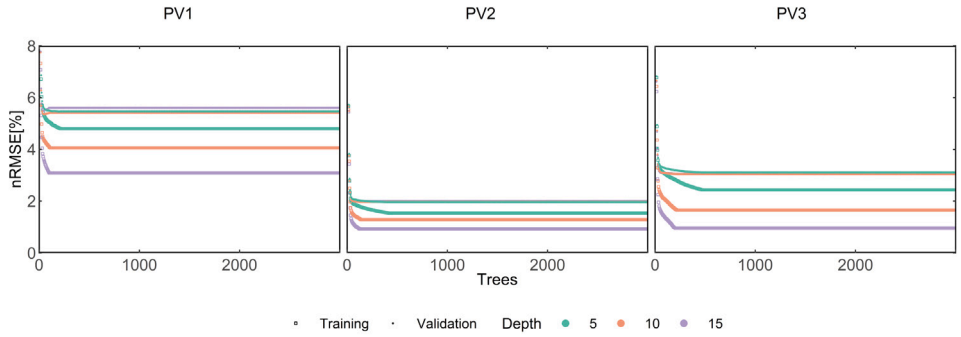
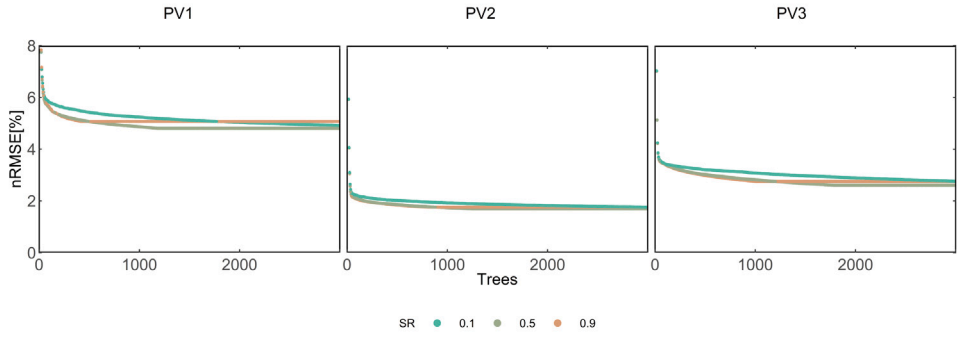


Fig. 9. GB Hyperparemeters search results.



(a) Influence of $Depth$ ($SR = 0.5$).



(b) Influence of SR ($Depth = 5$).

Fig. 10. GB Hyperparemeters search results. Influence of *Depth* and *SR*.

by the Hyperband search method, used in the first sweep. The second sweep was performed more simply using the random search method. Models containing 1, 2 and 3 hidden layers were tested for each PV plant exploring the number of *Units* and the *Dropout* in each hidden layer. The input layer dropout value was also considered. A total computational budget containing 10,550 models was used for each run. The second sweep, with random search, was only performed on the models with one hidden layer for all plants and the model with two hidden layers for PV1, as the rest did not improve the results. The number of explored models equalled 4000 in this second sweep.

Table 8 shows the hyperparameters values for the two sweeps. Four search runs, accounting for 35,650 explored models, for PV2 and

Table 8
Hyperparameters values used in the Hyperband Search for ANN. Where not indicated, values apply for the three plants.

		First sweep hyperband search			*Second sweep random search*		
		Min	Max	Delta	Min	Max	Delta
1 Layer	*Input Dropout*	0	0.9	0.1	–	–	–
	Units L1	2	512	2	1	512	1
	Dropout L1	0	0.9	0.1	0	0.3	0.1
2 Layers	*Input Dropout*	0	0.9	0.1	– (PV1)	– (PV1)	– (PV1)
	Units L1	2	512	2	350 (PV1)	512 (PV1)	1 (PV1)
	Dropout L1	0	0.9	0.1	0(PV1)	0.3 (PV1)	0.1 (PV1)
	Units L2	2	512	2	1 (PV1)	512 (PV1)	1 (PV1)
	Dropout L2	0	0.9	0.1	0 (PV1)	0.3 (PV1)	0.1 (PV1)
3 Layers	*Input Dropout*	0	0.9	0.1			
	Units L1	2	512	2			
	Dropout L1	0	0.9	0.1			
	Units L2	2	512	2			
	Dropout L2	0	0.9	0.1			
	Units L3	2	512	2			
	Dropout L3	0	0.9	0.1			
Models tested		10550 (each run)			4000 (each run)		

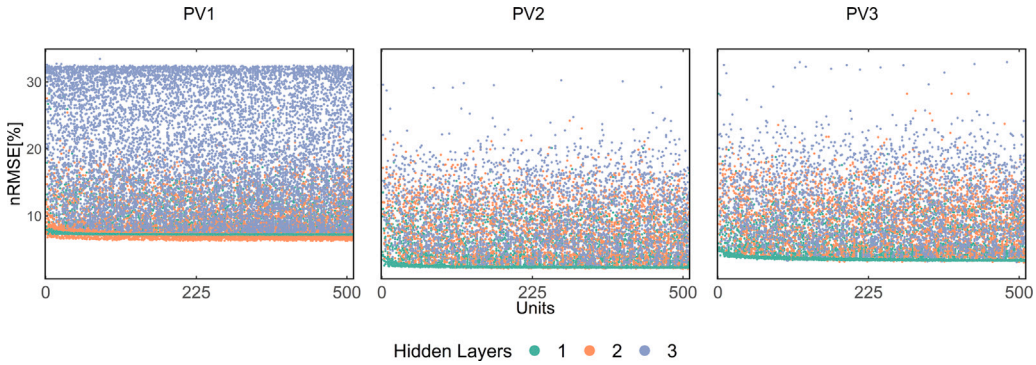


Fig. 11. Overall view of ANN Hyperparemeters search results.

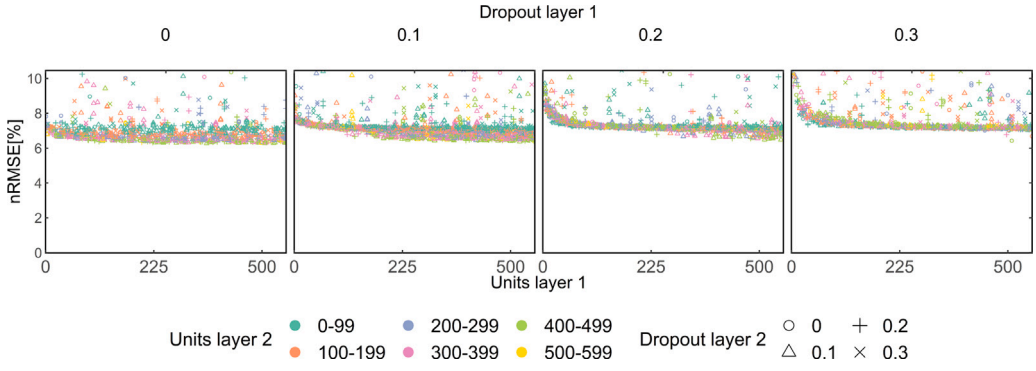


Fig. 12. ANN hyperparameters search results for PV1.

PV3, and five search runs with 39,650 explored models for PV1, were performed.

Fig. 11 shows the error obtained for all the explored models in function of the number of neurons (*Units*) with the colour indicating the corresponding hidden layer number. The error is bounded in a range of values where the upper limit means that the Hyperband stopping criterion is suitable, while the clearer lower limit indicates that proper solutions minimising the error were found.

Fig. 11 reveals that the minimum error is found for models with only one hidden layer for PV2 and PV3, while for PV1, a second hidden layer is needed.

Fig. 12 summarises the dependence of the error with the *Dropout* and the *Units* of each hidden layer in the models explored for PV1. The lowest error is achieved with null *Dropout* in the first hidden layer but not in the second, where 0.1 is selected. The number of *Units* has an asymptotic behaviour in both hidden layers which allowed the proper selection of the number of neurons.

Fig. 13 plots the influence of the *Dropout* and *Units* of the hidden layer on the error in PV2 and PV3. The *Droput* of the hidden layer for PV2 should be null, while in the PV3 plant is a little better to select the value 0.1. The number of neurons were selected with the same criteria used for PV1.

As it was said before, the learning rate was kept constant and equal to 10^{-4} while the number of epochs was equal to 81 for all explored models. Table 9 shows the summary of results for the hyperparameters search of the ANN models for each PV plant.

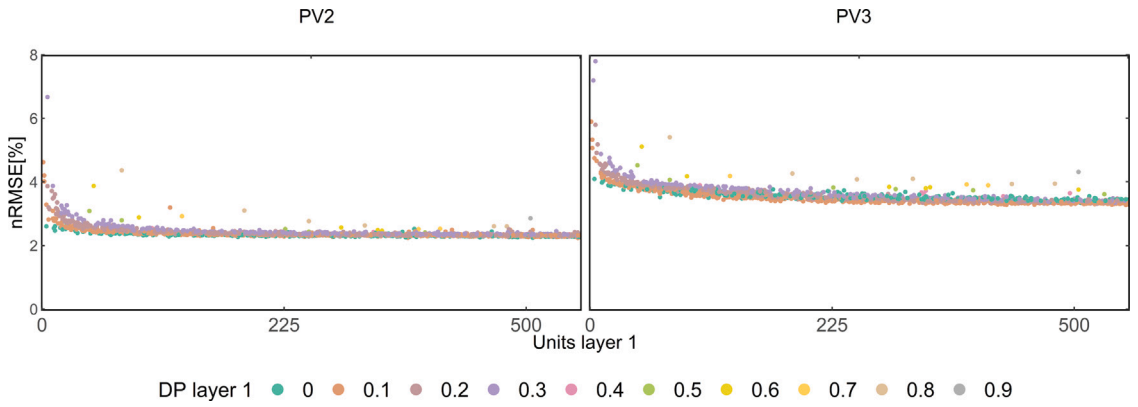


Fig. 13. ANN hyperparameters search results for PV2 & PV3.

Table 9
Selected hyperparameters and metrics for ANN.

	PV1	*PV2*	*PV3*
Input Dropout	0	0	0
Droput L1	0	0	0.1
Droput L2	0	–	–
Units L1	239	204	114
Units L2	475	–	–
Learning rate	10^{-4}	10^{-4}	10^{-4}
Epochs	81	81	81
nRMSE	6.6%	2.3%	3.8%
Explored models	46200	35650	35650

4.3. Performance evaluation and comparison of the optimised models

Here, the performance of the final models is evaluated and compared. The *test* data subsets and the optimised hyperparameters have been used to obtain the results and calculate the metrics each model provides in each PV plant. Fig. 14 shows a scatter plot representing the estimated versus the measured power, for each model and plant. The dispersion observed in the plots indicates the goodness of the models. Less dispersion means better performance of the model. The plots also show the regression (orange) and the unity slope (green) lines.

Plant PV2 has the lowest dispersion and shows the best regression. This can be due to the plant location in a flat area, without shadows, and the uniform distribution of the met-stations covering the whole studied surface. Plant PV1 shows the highest dispersion and the worst regression. This results may be due to the location in complex terrain, which means more shadows in the PV panels, and the availability of only one met-station. Finally, plant PV3 is also in complex terrain but has more met-stations available. Its results are between those of PV1 and PV2, probably due to a better representation of the radiation in the PV field due to the met-stations availability.

Regarding the models, Fig. 14 clearly shows that the MLR is the worst performing with high dispersion and the worst regression. This result was expected because MLR is a naive approximation used as a baseline for evaluating the improvements of the models. Observing the differences among the other three proposed models is difficult because they all look like reasonable options. Table 10 shows the numerical metrics for the three PV plants, including again the results of MLR model as baseline.

The results presented in Table 10 show that the best estimations (in green) for the three PV plants are achieved with RF, closely followed by GB. Only the normalised Bias for PV1 and PV2 favour the MLR. Nevertheless, the nBias results are minimal in all cases and cannot be taken as a bad indicator. It is worth noting that being PV1 the plant with the worst results, its Skill Score for the RF model is the best of the three plants, reaching 30%, while for PV2 and PV3, the SS of the RF model is around 25%. Finally, it is significant that the ANN model, claimed as the best solution in many papers, was found as the worst option for the three tested PV plants.

4.4. Comparison with other models found in the literature

Comparing the results with those obtained by other authors is difficult due to plant sizes, the type of data and their sample rate, the frequency of the estimates and the evaluation metrics. However, this section compares the most similar models found in the literature.

Five articles providing at least the average value of the nRMSE were selected. They include Multiple Linear Regression, Artificial Neural Networks, Support Vector Machines, Gaussian Process Regression (GPR) and Random Forests.

Graditi et al. (2016) developed one parametric and two ML, MLR and ANNs, using a large dataset with 7 years of data. The plant had a power value equal to 1 kWp. The objective was to determine the optimal subset of data using a Genetic Algorithm (GA). The ANN model was constructed with one hidden layer containing 3 neurons without any optimisation. The final models used only two days of data providing error values similar to those obtained with one year of data. Huang et al. (2016) focused on improving the accuracy of ANNs in the estimation of the power including inputs related to the sun position. Data were from two sites with 1 kWp and 78.7 kWp. The ANN model was constructed with one hidden layer containing six and fourteen neurons without clearly explaining the reasons. The results were better for the models including the sun position variables. There was a big difference between the errors in the studied sites, justified by the larger size of the second one. The authors also compared with SVM and GPR. Pan and Tan (2019) proposed to cluster the weather regimes and fit a RF model for each cluster. An ensemble was then constructed using Ridge Regression to obtain the weights of each weather regimen prediction. One year of freely available data of three plants with peak power varying from 1.5 kWp to 5 kWp were used. Theocharides et al. (2020) developed a method based on an ensemble of ANNs built from 5 wheater clusters. They used data from a test-bench with a power of 1.2 kWp and meteorological data obtained from the Weather Research and Forecasting (WRF) Model. Trigo-Gonzalez et al. (2021) used data from one locations with 9.3 kWp and two other with 2.8 kWp. They developed local models with data from each location and one global model using data from the three locations and including the altitude. They optimised the input variables by studying the results obtained with different combinations for ANN, SVM and MLR models. Some optimisation of the hyperparameters of the ANNs were also done. Nevertheless, the range of variation of the number of neurons was very small.

Summarising, the different authors considered important to include in the models not only meteorological variables, sun radiation, temperature, wind speed, module temperature, etc. but also the position of the sun, the weather regime and the location and altitude of the

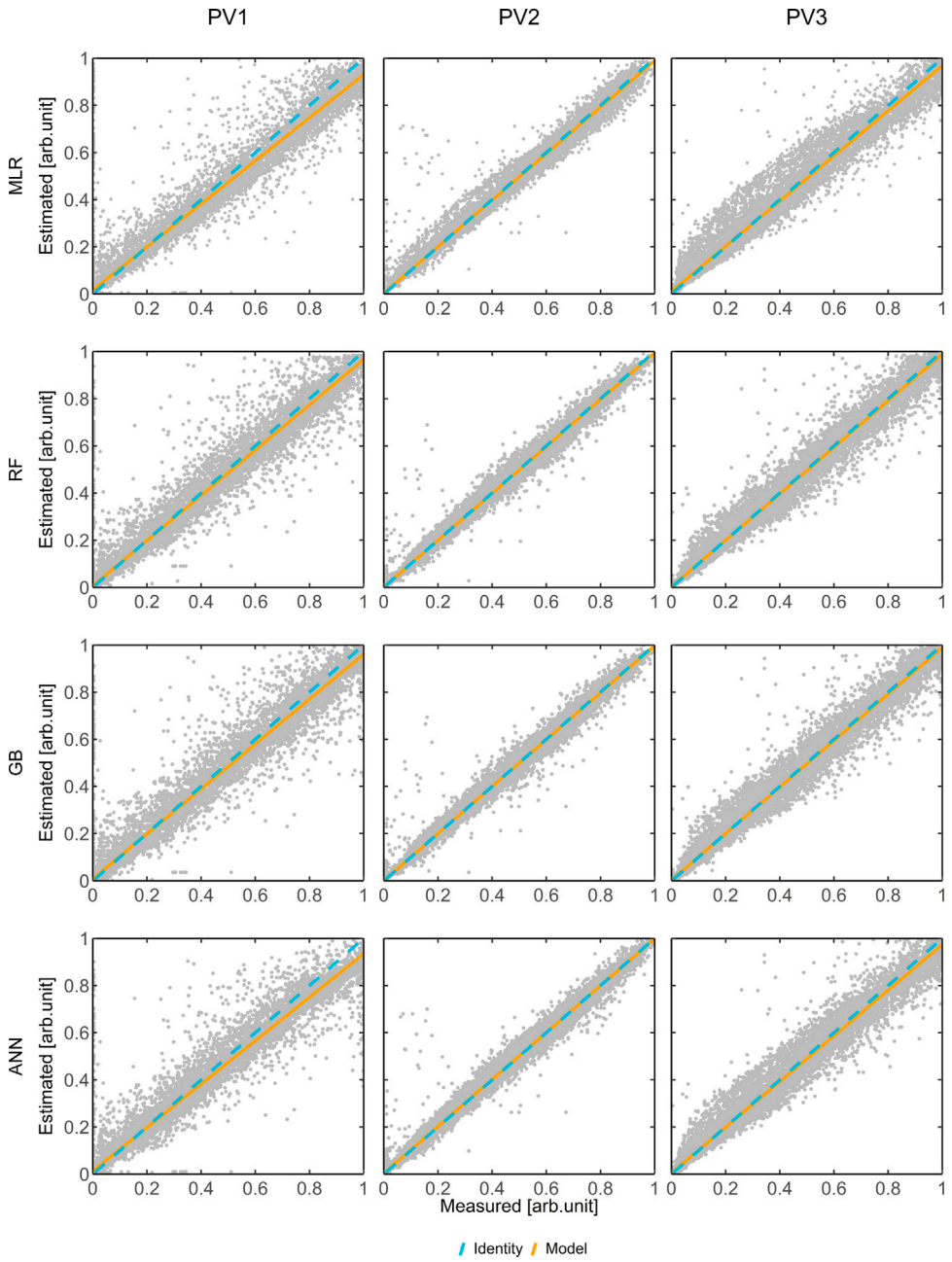


Fig. 14. Plots of the estimated versus measured power for the developed models.

Table 10
Final results of the optimised models.

	PV1				PV2				PV3			
	MLR	RF	GB	ANN	MLR	RF	GB	ANN	MLR	RF	GB	ANN
nMAE (%)	2.846	1.899	2.150	2.436	1.196	0.752	0.797	0.988	2.148	1.253	1.372	1.660
nBIAS (%)	0.005	0.055	0.029	0.048	−0.004	−0.012	−0.021	−0.280	−0.020	−0.001	−0.020	0.047
nRMSE (%)	7.858	5.436	5.852	6.638	2.585	1.926	1.997	2.218	4.093	3.075	3.188	3.534
SS (%)	–	30.821	25.523	15.530	–	25.466	22.746	14.169	–	24.866	22.110	13.669

plant to improve the power estimation. None of them took into account the size of the plant and only one paper justified the bad results with a bigger size of the PV field. Table 11 shows the comparison of the results found with those of this work. The best results were achieved in this work using data from PV2 for all the developed models. The properties of the plant, located in flat terrain without shadows, the use of more than one meteorological station to represent the PV field and the systematical optimisation of the hyperparameters of the models

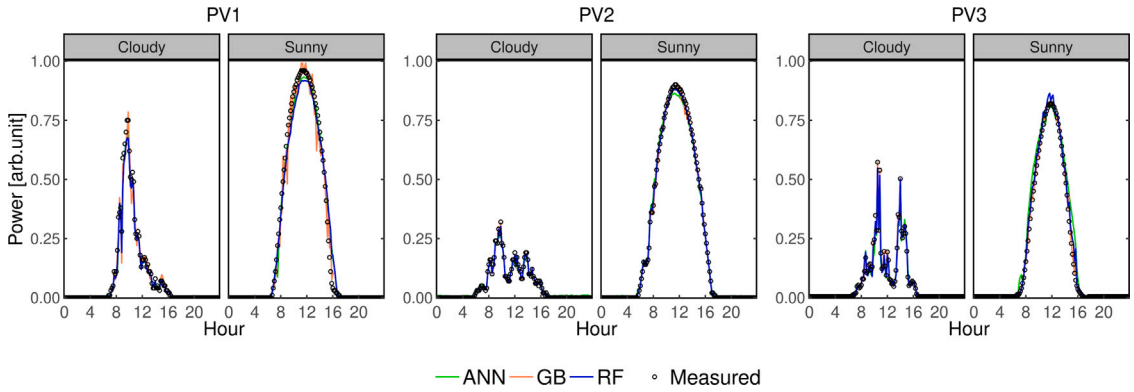


Fig. 15. Measured and the estimated power of PV1, PV2 and PV3 in a cloudy and a sunny day.

Table 11
Comparison with other authors.

Reference	Method	PV plant capacity (kWp)	Average nRMSE (%)
Graditi et al. (2016)	MLR ANN	1.0	5.0 4.6
Huang et al. (2016)	ANN SVM GPR	1.0–79	3.8–7.3 3.8–7.5 3.2–7-0
Pan and Tan (2019)	RF	1.5–5.0	8.8
Theocharides et al. (2020)	ANN	1.2	6.1
Trigo-Gonzalez et al. (2021)	MLR ANN SVN	9.3–2.8	6.3 4.5 3.5
This work			PV1, PV2, PV3
	MLR	1000	7.9, 2.6, 4.1
	RF		5.4, 1.9, 3.1
	GB		5.9, 2.0, 3.2
	ANN		6.6, 2.2, 3.5

allowed those results. Looking at the results of the other two PV plants, it can be seen that PV3 is also better than those in the literature, while the errors of PV1 are slightly higher, which can be justified by the location of the PV field in a complex terrain with shadows and the use of a single measurement station.

5. Discussion and concluding remarks

The low costs of photovoltaic generation have contributed to the developing and installing of many large utility-scale PV plants. The accurate estimation of the production of these plants is essential to characterise their performance, detect energy losses associated with component failures and ensure their grid integration. This work contributes with the development of accurate models to know the production of large plants at the inverter level with a time scenario of 10 min.

Three non-parametric models based on different machine learning techniques to estimate the power production of large utility-scale PV plants are proposed. The methods used were Random Forests, Gradient Boosting and Artificial Neural Networks. They were compared with a Multiple Linear Regression model used as a baseline due to its simplicity. Some of these techniques have been previously applied to PV plants with good results. Nevertheless, the models proposed so far were defined with small plants well below the megawatt range, where the meteorological measurements are in the same location as the generation. Plants installed nowadays can reach hundreds of megawatts with distances between the meteorological stations and the PV modules of hundreds or even thousands of metres. Therefore, it is necessary to somehow include in the models the effect of the size of the plant that affects the non-uniform distribution of solar radiation and the presence of shadows due to clouds and, sometimes, to the orography in which the plant is built. In this work, this effect was taken into account including the measurements of several met-stations distributed along the PV plant. Then, three PV plants with 1, 2 and 4 met-stations were studied.

Machine Learning techniques became very popular in all scientific fields. Nevertheless, the specific tuning of the models' hyperparameters is not always done. This work demonstrates how the hyperparameters can increase the model error up to 5 times, making the need for adjustment clear. A systematic methodology to optimise the hyperparameters has been carried out for each ML technique. Grid search, Hyperband search and Random search methods were used depending on the number of hyperparameters to optimise. This systematic optimisation has revealed better options than neural networks to model the plant. Less complex models with less computational demand, such as decision tree models, are in the same range of error or even improve it. This contribution is significant because these models have a more suitable implementation than neural networks in the day-to-day operation of photovoltaic plants, allowing energy losses to be controlled.

Once defined the hyperparameters, the final results were calculated using the *test* data subset. The results showed that even the simplest model, MLR, provided a reasonable adjustment and was reliable for estimating the production in a PV plant. However, it was also shown that the most advanced models could improve MLR up to 30%. The metrics showed that RF slightly outperforms GB while ANNs have a bit worst results. The results also revealed that using distributed data along the plant enhances the models' performance. Fig. 15 shows the estimated power of PV1, PV2 and PV3 for a cloudy and a sunny day, obtained with the three ML models. It can be seen how the best results were achieved in plant PV2, located in flat terrain and using two met-stations. Plants PV1 and PV3 are both located in more complex terrain, but the results of PV1 are worse than those of PV3 due to the availability of only one met-station in PV1, while PV3 has four.

Comparing the results with those from other authors, the models for PV2, including the simplest MLR, outperform all reported ones, indicating the appropriateness of including distributed data in the models, two met-stations in this case. Models developed for PV3 are in the range of those reported in the literature, while those generated for PV1 are only a bit worse, even with very different plant sizes. Then, the systematic search for optimal hyperparameters is highly recommended when developing ML models.

The results obtained have, however, certain limitations. On the one hand, the developed models allow a very precise estimation of the plant's production and hence the losses. However, it is impossible to determine the origin and root cause of the production losses, and it is necessary to use other information obtained at the plant. On the other hand, the study period is limited compared to the lifetime of the plants. It would be necessary either to use a more representative period of the plant life or to update the models based on actual plant measurements. Finally, although considerable computational resources are no longer necessary once the models have been developed, it should be noted that they are required for the optimisation of the hyperparameters and the initial training of the models.

In summary, the main contribution of this work is an accurate methodology for estimating the production of large utility-scale PV plants. It has been demonstrated that the size of the plants has to be accounted for in the models and this is done using data from several meteorological stations distributed in the plant. Finally, the optimisation of the hyperparameters may decrease the final error up to five times. The methodology can be used to characterise the performance of the plants, detect energy losses associated with component failures and ensure their grid integration.

Declaration of competing interest

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

References

Abdullayev, T., 2022. KerastuneR: R interface to Keras Tuner. URL: https://eagerai.github.io/kerastuneR/index.html. R package version 1.14.4.

Almeida, M.P., Perpiñán, O., Narvarte, L., 2015. PV power forecast using a nonparametric PV model. Sol. Energy 115, 354–368. http://dx.doi.org/10.1016/j.solener.2015.03.006.

AlSkaif, T., Dev, S., Visser, L., Hossari, M., van Sark, W., 2020. A systematic analysis of meteorological variables for PV output power estimation. Renew. Energy 153, 12–22. http://dx.doi.org/10.1016/j.renene.2020.01.150.

Bansal, N., Jaiswal, S.P., Singh, G., 2022. Long term performance assessment and loss analysis of 9 MW grid tied PV plant in India. Materials Today: Proceedings 60, 1056–1067. http://dx.doi.org/10.1016/j.matpr.2022.01.263.

Beránek, V., Olšan, T., Libra, M., Poulek, V., Sedláček, J., Dang, M.Q., Tyukhov, I.I., 2018. New monitoring system for photovoltaic power plants' management. Energies 11, http://dx.doi.org/10.3390/en11102495.

Breiman, L., 2001. Random Forests. Mach. Learn. 45 (1), 5–32. http://dx.doi.org/10.1023/A:1010933404324.

Dahmoun, M.E.H., Bekkouche, B., Sudhakar, K., Guezgouz, M., Chenafi, A., Chaouch, A., 2021. Performance evaluation and analysis of grid-tied large scale PV plant in Algeria. Energy for Sustain. Dev. 61, 181–195. http://dx.doi.org/10.1016/j.esd.2021.02.004.

Daliento, S., Chouder, A., Guerriero, P., Pavan, A.M., Mellit, A., Moeini, R., Tricoli, P., 2017. Monitoring, diagnosis, and power forecasting for photovoltaic fields: A review. Int. J. Photoenergy 2017, http://dx.doi.org/10.1155/2017/1356851.

Elsheikh, A.H., Sharshir, S.W., Elaziz, M.A., Kabeel, A.E., Guilan, W., Haiou, Z., 2019. Modeling of solar energy systems using artificial neural network: A comprehensive review. Sol. Energy 180, 622–639. http://dx.doi.org/10.1016/J.SOLENER.2019.01.037.

Fazai, R., Abodayeh, K., Mansouri, M., Trabelsi, M., Nounou, H., Nounou, M., Georghiou, G.E., 2019. Machine learning-based statistical testing hypothesis for fault detection in photovoltaic systems. Sol. Energy 190 (July), 405–413. http://dx.doi.org/10.1016/j.solener.2019.08.032.

Friedman, J.H., 2002. Stochastic gradient boosting. Comput. Statist. Data Anal. 38 (4), 367–378. http://dx.doi.org/10.1016/S0167-9473(01)00065-2.

Garoudja, E., Chouder, A., Kara, K., Silvestre, S., 2017. An enhanced machine learning based approach for failures detection and diagnosis of PV systems. Energy Convers. Manage. 151 (August), 496–513. http://dx.doi.org/10.1016/j.enconman.2017.09.019.

Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press, URL: http://www.deeplearningbook.org.

Graditi, G., Ferlito, S., Adinolfi, G., 2016. Comparison of photovoltaic plant power production prediction methods using a large measured dataset. Renew. Energy 90, 513–519. http://dx.doi.org/10.1016/j.renene.2016.01.027.

H2O.ai, 2020. H2o: R interface for H2O. URL: http://www.h2o.ai. R package version 3.30.0.6.

Huang, C., Bensoussan, A., Edesess, M., Tsui, K.L., 2016. Improvement in artificial neural network-based estimation of grid connected photovoltaic power output. Renew. Energy 97, 838–848. http://dx.doi.org/10.1016/j.renene.2016.06.043.

Hutter, F., Kotthoff, L., Vanschore, J., 2019. Automated Machine Learning Methods, Systems, Challenges. Springer, http://dx.doi.org/10.1007/978-3-030-05318-5.

International Renewable Energy Agency - IRENA, 2021. Renewable capacity statistics 2021. Technical Report, International Renewable Energy Agency - IRENA, pp. 1–64, URL: https://www.irena.org/publications/2021/March/Renewable-Capacity-Statistics-2021.

Jed, M.E.H., Logerais, P.O., Malye, C., Riou, O., Delaleux, F., Bah, M.E., 2021. Analysis of the performance of the photovoltaic power plant of Sourdun (France). Int. J. Sustain. Eng. 14, 1756–1768. http://dx.doi.org/10.1080/19397038.2021.1971321.

Ketjoy, N., Chamsa-ard, W., Mensin, P., 2021. Analysis of factors affecting efficiency of inverters: Case study grid-connected PV systems in lower northern region of Thailand. Energy Rep. 7, 3857–3868. http://dx.doi.org/10.1016/j.egyr.2021.06.075.

Kumari, P., Toshniwal, D., 2021. Deep learning models for solar irradiance forecasting: A comprehensive review. J. Clean. Prod. 318, 128566. http://dx.doi.org/10.1016/J.JCLEPRO.2021.128566.

Li, B., Delpha, C., Diallo, D., Migan-Dubois, A., 2021. Application of Artificial Neural Networks to photovoltaic fault detection and diagnosis: A review. Renew. Sustain. Energy Rev. 138, 110512. http://dx.doi.org/10.1016/J.RSER.2020.110512.

Li, L., Jamieson, K., DeSalvo, G., Rostamizadeh, A., Talwalkar, A., 2018. Hyperband: A novel bandit-based approach to hyperparameter optimization. J. Mach. Learn. Res. 18, 1–52, URL: https://jmlr.org/papers/volume18/16-558/16-558.pdf.

Mellit, A., Pavan, A.M., 2010. Performance prediction of 20 kWp grid-connected photovoltaic plant at Trieste (Italy) using artificial neural network. Energy Convers. Manage. 51 (12), 2431–2441. http://dx.doi.org/10.1016/j.enconman.2010.05.007.

Mellit, A., Tina, G.M., Kalogirou, S.A., 2018. Fault detection and diagnosis methods for photovoltaic systems: A review. Renew. Sustain. Energy Rev. 91 (February 2017), 1–17. http://dx.doi.org/10.1016/j.rser.2018.03.062.

Monteiro, R.V., Guimarães, G.C., Moura, F.A., Albertini, M.R., Albertini, M.K., 2017. Estimating photovoltaic power generation: Performance analysis of artificial neural networks, Support Vector Machine and Kalman filter. Electr. Power Syst. Res. 143, 643–656. http://dx.doi.org/10.1016/j.epsr.2016.10.050.

Moreira, M.O., Balestrassi, P.P., Paiva, A.P., Ribeiro, P.F., Bonatto, B.D., 2021. Design of experiments using artificial neural network ensemble for photovoltaic generation forecasting. Renew. Sustain. Energy Rev. 135 (October 2020), 110450. http://dx.doi.org/10.1016/j.rser.2020.110450.

Nobre, A.M., Severiano, C.A., Karthik, S., Kubis, M., Zhao, L., Martins, F.R., Pereira, E.B., Rüther, R., Reindl, T., 2016. PV power conversion and short-term forecasting in a tropical, densely-built environment in Singapore. Renew. Energy 94, 496–509. http://dx.doi.org/10.1016/j.renene.2016.03.075.

Øgaard, M.B., Skomedal, A.F., Haug, H., Marstein, E.S., 2021. Robust and fast detection of small power losses in large-scale PV systems. IEEE J. Photovolt. 11 (3), 819–826. http://dx.doi.org/10.1109/JPHOTOV.2021.3060732.

Pan, M., Li, C., Gao, R., Huang, Y., You, H., Gu, T., Qin, F., 2020. Photovoltaic power forecasting based on a support vector machine with improved ant colony optimization. J. Clean. Prod. 277, 123948. http://dx.doi.org/10.1016/j.jclepro.2020.123948.

Pan, C., Tan, J., 2019. Day-ahead hourly forecasting of solar generation based on cluster analysis and ensemble model. IEEE Access 7, 112921–112930. http://dx.doi.org/10.1109/ACCESS.2019.2935273.

Piliougine, M., Spagnuolo, G., 2022. Mismatching and partial shading identification in photovoltaic arrays by an artificial neural network ensemble. Sol. Energy 236, 712–723. http://dx.doi.org/10.1016/j.solener.2022.03.026, URL: https://linkinghub.elsevier.com/retrieve/pii/S0038092X2200192X.

Platon, R., Pelland, S., Poissant, Y., 2012. Modelling the power production of a photovoltaic system: Comparison of sugeno-type fuzzy logic and PVSAT-2 models. In: Europe Solar Conference (ISES).

R. Core Team, 2021. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria, URL: https://www.R-project.org/.

Sabbaghpur Arani, M., Hejazi, M.A., 2016. The comprehensive study of electrical faults in PV arrays. J. Electr. Comput. Eng. 2016, http://dx.doi.org/10.1155/2016/8712960.

Sohani, A., Sayyaadi, H., Cornaro, C., Shahverdian, M.H., Pierro, M., Moser, D., Karimi, N., Doranehgard, M.H., Li, L.K., 2022. Using machine learning in photovoltaics to create smarter and cleaner energy generation systems: A comprehensive review. J. Clean. Prod. 364, 132701. http://dx.doi.org/10.1016/j.jclepro.2022.132701, URL: https://linkinghub.elsevier.com/retrieve/pii/S0959652622022995.

Spertino, F., Amato, A., Casali, G., Ciocia, A., Malgaroli, G., 2021a. Reliability analysis and repair activity for the components of 350 kw inverters in a large scale grid-connected photovoltaic system. Electronics (Switzerland) 10, 1–13. http://dx.doi.org/10.3390/electronics10050564.

Spertino, F., Chiodo, E., Ciocia, A., Malgaroli, G., Ratclif, A., 2021b. Maintenance activity, reliability, availability, and related energy losses in ten operating photovoltaic systems up to 1.8 MW. IEEE Trans. Ind. Appl. 57, 83–93. http://dx.doi.org/10.1109/TIA.2020.3031547.

Theocharides, S., Makrides, G., Livera, A., Theristis, M., Kaimakis, P., Georghiou, G.E., 2020. Day-ahead photovoltaic power production forecasting methodology based on machine learning and statistical post-processing. Appl. Energy 268 (April), 115023. http://dx.doi.org/10.1016/j.apenergy.2020.115023.

Trigo-Gonzalez, M., Cortés, M., Alonso-Montesinos, J., Martínez-Durbán, M., Ferrada, P., Rabanal, J., Portillo, C., López, G., Batlles, F.J., 2021. Development and comparison of PV production estimation models for mc-Si technologies in Chile and Spain. J. Clean. Prod. 281, http://dx.doi.org/10.1016/j.jclepro.2020.125360.