energies

MDPI

Article

Multiparameter Regression of a Photovoltaic System by Applying Hybrid Methods with Variable Selection and Stacking Ensembles under Extreme Conditions of Altitudes Higher than 3800 Meters above Sea Level

Jose Cruz [1,*], Christian Romero [1], Oscar Vera [2], Saul Huaquipaco [2,*], Norman Beltran [1] and Wilson Mamani [1]

[1] Faculty of Mecánica Eléctrica, Electrónica y Sistemas, Universidad Nacional del Altiplano, Puno 21001, Peru; cromero@unap.edu.pe (C.R.); nbeltran@unap.edu.pe (N.B.); wilmamanimac@est.unap.edu.pe (W.M.)
[2] School of Ingeniería de Sistemas e Informática, Faculty of Engineering, Universidad Nacional de Moquegua, Moquegua 18001, Peru; overar@unam.edu.pe
* Correspondence: josecruz@unap.edu.pe (J.C.); shuaquipacoe@unam.edu.pe (S.H.)

Abstract: The production of solar energy at altitudes higher than 3800 m above sea level is not constant because the relevant factors are highly varied and complex due to extreme solar radiation, climatic variations, and hostile environments. Therefore, it is necessary to create efficient prediction models to forecast solar production even before implementing photovoltaic systems. In this study, stacking techniques using ElasticNet and XGBoost were applied in order to develop regression models that could collect a maximum number of features, using the LASSO, Ridge, ElasticNet, and Bayesian models as a base. A sequential feature selector (SFS) was used to reduce the computational cost and optimize the algorithm. The models were implemented with data from a string photovoltaic (PV) system in Puno, Peru, during April and August 2021, using 15 atmospheric and photovoltaic system variables in accordance with the European standard IEC 61724-20170. The results indicate that ElasticNet reduced the MAE by 30.15% compared to the base model, and that the XGBoost error was reduced by 30.16% using hyperparameter optimization through modified random forest research. It is concluded that the proposed models reduce the error of the prediction system, especially the stacking model using XGBoost with hyperparameter optimization.

Keywords: multiparametric regression; photovoltaic; stacking; hyperparameter optimization

Citation: Cruz, J.; Romero, C.; Vera, O.; Huaquipaco, S.; Beltran, N.; Mamani, W. Multiparameter Regression of a Photovoltaic System by Applying Hybrid Methods with Variable Selection and Stacking Ensembles under Extreme Conditions of Altitudes Higher than 3800 Meters above Sea Level. *Energies* **2023**, *16*, 4827. https://doi.org/10.3390/en16124827

Received: 20 April 2023
Revised: 9 June 2023
Accepted: 19 June 2023
Published: 20 June 2023

1. Introduction

Photovoltaic renewable energy represents a decentralized energy source, generating energy close to the place of consumption, which reduces the need for large energy transmission infrastructures. The extreme conditions associated with solar radiation, climatic variations, and remote locations can pose difficulties in the use of photovoltaic systems. However, the potential solar energy in high mountains means that studying photovoltaic systems at elevations higher than 3800 m above sea level is important for better understanding their efficiency, adaptability, and potential under extreme conditions and in remote locations so as to more effectively harness solar energy using high mountains [1–4].

Due to the randomness and intermittency of photovoltaic power generation [5], it is necessary to predict its potential to reduce any negative effects. Because of their simplicity, linear regression models generalize the results better than other models, so they can be used for this type of forecasting. This type of model has the disadvantage of overfitting, which can be corrected thanks to regularization techniques. For example, the authors of [6] mention that for large volumes of data in which there are redundant data with noise or with outliers, the selection of features is a good option for improving the regression or classification of the results. In the same way, for the prediction of daily solar energy [7],

the authors propose a feature selection approach based on two factors: the selection and grouping of features based on relevance and redundancy and a hybrid classification and regression prediction algorithm. Additionally, for the prediction of photovoltaic energy through the global horizontal irradiance [8], the authors identified a set of relevant characteristics: the UV index, cloud cover, air temperature, relative humidity, dew point, wind, the duration of insolation, and the time of day, ruling out other factors such as intensity of precipitation, probability of precipitation, wind speed, atmospheric pressure at sea level, the day of the year, and the minute of the hour. For this, they used six variable selection methods grouped into three categories: (i) filter methods: correlation through the coefficient of determination R^2 and mutual information; (ii) wrapper methods: sequential forward selection and sequential backward selection; and (iii) embedded methods: LASSO and random forest. Additionally, for the forecasting of photovoltaic energy one day in advance in French Guiana [9], an SFS with a kernel conditional density estimator (KCDE) was used to select the variables used, for which it compared the results obtained with three other methods: regularization (Pearson correlation), RReliefF, and an SFS based on linear regression. In addition, [10] predicted the energy provided by photovoltaic panels through the generation of dimensional spaces considering environmental variables and satellite measurements that corresponded to a point in space; subsequently, using the SFS algorithm, the authors determined the influence of each of the generated spaces on the place where the system to be forecast was located.

In this investigation, we sought to calculate the power of photovoltaic panels using the stacking technique to generate our model, taking MAE, RMSE, and R^2 as the metrics. Hyperparameter optimization is a subject that has received little attention in terms of the generation of linear regression models. This optimization is usually performed in the training stage without the application of any specific methodology [11]. In order to achieve ultrafast and highly accurate power prediction for a PV system [12], the gray wolf optimization method can be used, among other methods, to find the least-squares optimal support vector machine parameters for the short-term forecast of the production of photovoltaic energy. The authors of [13] used a PV plant in Brazil and Bayesian optimization methods to find the optimal hyperparameters for a system, indicating that the best results were obtained by using MLP in terms of usability and training time. In the same way, for the identification of two photovoltaic systems, one installed in Teresina, Brazil, and the other in Hamburg, Germany [14], the use of genetic algorithms (GAs) and particle swarm optimization (PSO) helped optimize the hyperparameters of the neural models multilayer perceptron, extreme learning machine, and echo state network. For the detection of faults in photovoltaic systems, the authors of [15] proposed two control charts called EWMA and DEWMA based on joint learning using two models: boosted trees and bagged trees. In both algorithms, they used Bayesian optimization to determine the hyperparameters of both models, and, as a result, they indicated that the best model was the DEWMA model. In this same field, but with regard to achieving optimal power-generation-unit scheduling in the presence of uncertainties on both the demand and supply sides, the authors of [16] generated a regression model for power forecasting based on neural networks and time series, such as LSTM, which required hyperparameter optimizations, four of which were proposed by the authors: traditional manual adjustment, automatic adjustment for blocks, a framework called Optuna with a grid search algorithm, and adapting Optuna with a Bayesian optimization framework. In order to improve predictions in terms of error, stacking can be used, which is a machine learning technique that combines various linear regression models to improve the accuracy of the predictions. ML applications can make it possible to exploit the operation of plants in the best way, forecasting weather conditions, such as the exposure of photovoltaic surfaces to the sun, the direction and force of the wind in the case of wind power, or rain on hydroelectric generators. In the context of photovoltaic systems, stacking can be useful because the factors that affect solar energy production can be complex and varied. By combining several linear regression models, with each focused on different aspects of the data, the meta-learner provides the final

solution, giving more accurate and robust predictions [17,18]. In addition, stacking helps reduce overfitting, which occurs when a model overfits the training data and does not generalize well to new data. In general, the use of stacking can help improve the efficiency and accuracy of photovoltaic systems. In the application of stacking on photovoltaic systems, the authors of [19] proposed a stacking model based on the AdaBoost assembly model in the detection of three types of faults: open-circuit faults, short-circuit faults, and degradation faults. The results showed an accuracy of 97.84%, which indicates better results than the algorithms that comprise it. However, the authors of [20] forecasted the load in a photovoltaic system due to heating and cooling systems by proposing a stacking system with a BP neural network, support vector regression, and random forest in its first layer, and an algorithm or meta-learner gradient-boosting decision tree in its second layer. In order to achieve this, they initially performed a correlation analysis between the variables that made up the system, as well as hyperparameter optimization based on the grid search. For the stable and safe integration of photovoltaic energy into an existing electrical network, the authors of [21] proposed a forecast model based on stacking called DSE-XGB using two algorithms: artificial neural networks and short-term memory (as level zero) and the XGBoost gradient-boosting algorithm to integrate the results (as level one). The data used to evaluate the results were divided into four different groups. As a base level with which to evaluate the performance, the same results of the zero-level algorithms were used, achieving an improvement in the stacking set of 10% to 12% with respect to the coefficient of determination R. In the same way, the authors of [22] improved the forecasting of photovoltaic energy regarding its integration within electrical distribution systems; they proposed an algorithm called Stack-ETR, which took as its base algorithms random forest regressors; one with an extreme gradient, and one of adaptive reinforcement. In order to stack the previous models, extra-tree regression was used as a meta-learning algorithm. The validation was carried out using data from three photovoltaic systems for 4 years, comparing the results of the complete model with the results provided by each of the base algorithms, with improvements of 40.2% and 47.2% for RMSE and MAE, respectively. For the same purpose, but using four stacking models called stacking GDBT (with base models XGB, LGB, and RF), XGB stacking (with GBDT, LGB, and RF models), RF stacking (with XGB, LGB, and GBDT models), and finally LGB stacking (with XGB, GBDT, and RF LGB base models), the authors of [23] made their predictions using two datasets, mentioning that the highest precision was achieved when using the GDBT stacking model; a comparison was also made with respect to the original models without stacking. Regarding short-term prediction for the same purpose, the authors of [24] used a stacking model with five random-forest-type base models on a dataset that had been previously classified into five categories using SVM according to climate. Next, as a two-level stacking model, they used linear regression by applying regularization techniques, finding the optimal parameters using Bayesian techniques. Additionally, for short-term predictions, the authors of [25] used stacking to improve prediction, with XGBoost, RF, CatBoost, and LGBM as the first level in the model and support vector regression as a second-level algorithm. In order to demonstrate the efficiency of the results, the model was compared with the results of XGBoost alone and then with combinations of level-one algorithms, obtaining the best result by stacking XGBoost, CatBoost, LGBM, and RF, with a score of 90.85% and an RMSE of 0.1007, using meteorological condition data and parameters for the photovoltaic system. Furthermore, for short-term forecasting, but taking into account the special climatic conditions of an arid region, the authors of [26] proposed a stacking system with level-one algorithms XGBoost, random forest, and multiple linear regression, and linear regression with LASSO-type regularization as a second-level algorithm. They performed a correlation analysis for the choice of features for level one in the models. They also performed the optimal selection of hyperparameters using the hyperopt library. In the same way, the authors of [27] proposed a new framework for the forecasting of photovoltaic energy called the "enhanced deep belief network" due to the volatility of this type of energy. In order to do this, they used stacking, with extreme learning machines, extremely randomized

trees, k-nearest neighbor, and a deep belief network (DBN) as the level-one algorithms and the tree-structured algorithm of the Parzen estimators as a parameter optimization algorithm as a level-two algorithm. The results showed an improvement in terms of reducing the absolute error from 7.5 kW to 2.70 kW with respect to the original DBN model. Likewise, the authors of [28] used forecast algorithms based on support vector regression (SVR) with different kernel functions, with SVR used again in the second layer along with the artificial fish swarm algorithm to optimize the parameters. They used k-fold cross-validation techniques using public data from Spain for the year 2015, indicating a result for the best MAPE model of 0.88%.

For the evaluation or performance of the proposed models in this study, the mean absolute error (MAE), the coefficient of determination (R^2), the adjusted coefficient of determination (R^2_{adj}), and the mean squared error (MSE) were used as the metrics. These metrics were used by [29] to guarantee the stable operation of a photovoltaic system based on predictions made with four stacking models (XGB, LGB, and RF with GBDT stacking; XGB, GBDT, and RF with LGB stacking; XGB, LGB, and GBDT with RF stacking; and GBDT, LGB, and RF with XGB stacking). In the same way, the authors of [30] used MRE, MAE, NRMSE, and R^2 to compare the performance of their proposed stacking model with an artificial neural network, a deep neural network, support vector regression, short-term memory (in the long term), and a convolutional neural network as the level-one algorithms, and a recurrent neural network as the level-two algorithm. In addition, the authors of [31] used a short-term solar energy-prediction model based on a CNN-stacked LSTM technique with several steps and also used the same metrics (as above) to evaluate their results.

Considering the advantages of models with variable selection, stacking, and hyperparameter optimization, the contributions of this research are as follows:

- The implementation of a photovoltaic system under the extreme conditions of an altitude 3800 m above sea level;
- The implementation of four hybrid models (of various selections) using regularization and a sequential feature selector;
- The implementation and validation of a multiparameter regression meta-model based on super-learning for a photovoltaic system using hyperparameter optimization techniques.

2. Materials and Methods

2.1. Dataset

The dataset was obtained from measurements made using a string PV photovoltaic system (following the European standard IEC 61724-20170) in April and August 2021 in the department of Puno, Peru, with the coordinates 15°29′20″ S, 70°9′6″ W and at an altitude 3800 m above sea level. Temperatures ranged from −2 degrees Celsius to 27 degrees Celsius, and the maximum irradiance was 1522 Wh/m^2. The photovoltaic system comprised a set of photovoltaic solar panels that were connected in series and with a single DC/AC converter, as shown in Figure 1.

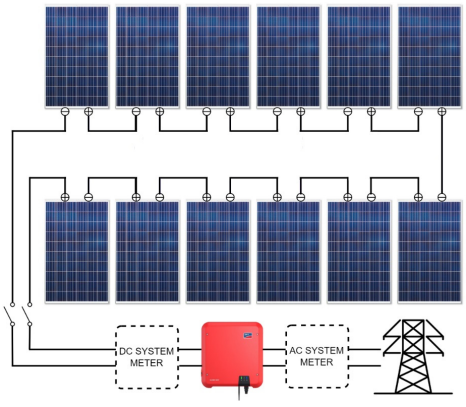


Figure 1. String PV photovoltaic system.

The dependent variable was Active power, and the independent variables were AC voltage, AC current, Active power, Apparent power, Reactive power, Frequency, Power factor, Total power, Daily power, DC voltage, DC current, DC power, Irradiance, Module temperature, and Ambient temperature, which represent the eigenvalues of the photovoltaic system and the atmospheric variables in extreme altitude conditions. To be applied to linear regression, these variables must have a normal or close-to-normal distribution. Therefore, the distribution of some of the variables is shown in Figure 2.

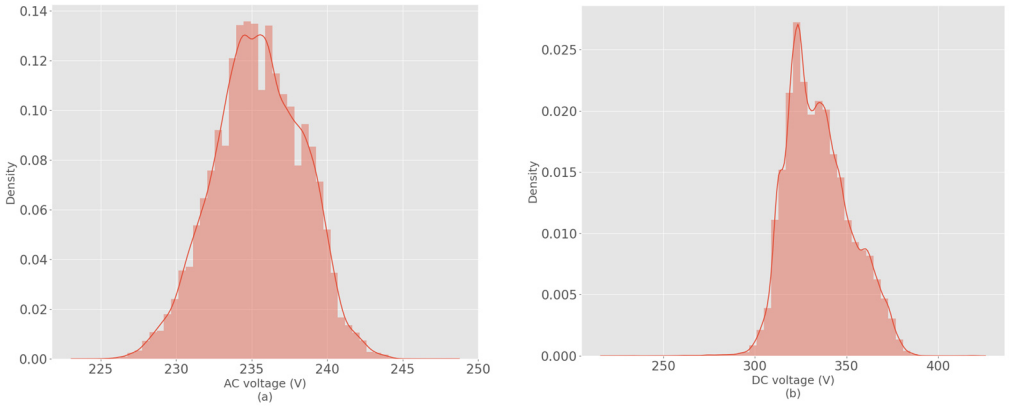


Figure 2. Normal distribution of selected variables: (**a**) AC voltage; (**b**) DC voltage.

Figure 2 shows the distribution of two variables, AC voltage and DC voltage, as well as the other variables with a normal distribution. In the same way, the outliers or atypical values should not be representative, which is visualized in the boxplots of the independent variables used. Figure 3 shows the distribution of the outliers, which denotes that there is a low number of outliers in the data.

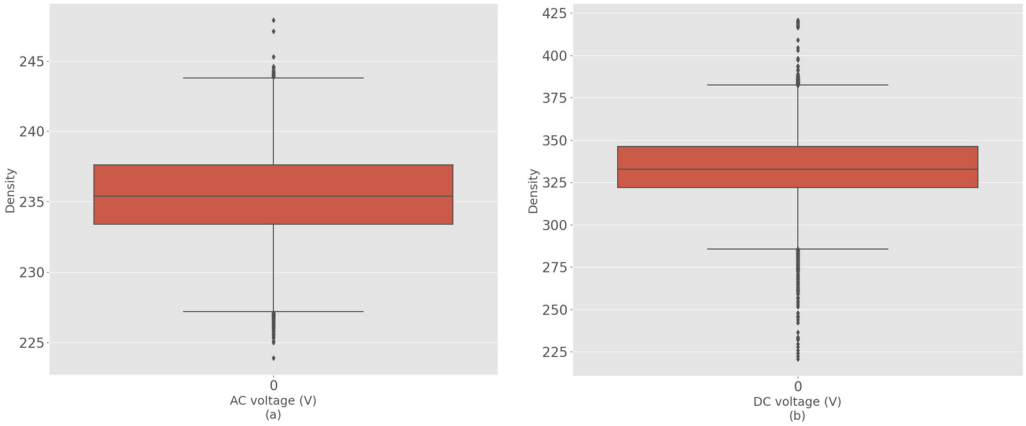


Figure 3. Boxplots of selected variables: (**a**) AC voltage; (**b**) DC voltage.

2.2. Basic Model

ElasticNet

Linear regression models are applied to predict a continuous quantitative value as an output. Linear regression involves an algorithm that finds patterns in the linear relationships between the independent variables and the dependent variable. One of its disadvantages is overfitting, which causes a loss of generalization to the regression. To correct this, regularization techniques, such as L1, also known as Ridge; L2, known as LASSO; and L3, which uses a combination of both of the previous techniques (L1 + L2) and is called ElasticNet, are used. This type of regularization is modeled by the following equations.

The sum of squared errors used by Ridge is shown in Equation (1).

$$E = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \tag{1}$$

The y_i refers to the actual value and $\hat{y}_i$ refers to the predicted value. The data have a Gaussian distribution, which is given by $N(\mu, \sigma^2)$; that is, with X as the input matrix. The probability x_i is represented in Equation (2).

$$P(x_i) = \frac{1}{2\pi\sigma^2} e^{\frac{-1}{2}\frac{(x_i-\mu)^2}{\sigma^2}} \tag{2}$$

The joint probability is represented in Equation (3), considering that all events are independent.

$$p(x_1, x_2, \ldots x_n) = \prod_{i=1}^{N} \frac{1}{2\pi\sigma^2} e^{\frac{-1}{2}\frac{(x_i-\mu)^2}{\sigma^2}} \tag{3}$$

In linear regression, the maximum probability is shown in Equation (4).

$$P(X|\mu) = p(x_1, x_2, \ldots x_n) = \prod_{i=1}^{N} \frac{1}{2\pi\sigma^2} e^{\frac{-1}{2}\frac{(x_i-\mu)^2}{\sigma^2}} \tag{4}$$

To maximize this result, the natural logarithm and a derivation equal to zero were used, as shown in the following equations:

$$\ln(P(X|\mu)) = \ln(p(x_1, x_2, \ldots x_n)) = \ln\left(\prod_{i=1}^{N} \frac{1}{2\pi\sigma^2} e^{\frac{-1}{2}\frac{(x_i-\mu)^2}{\sigma^2}}\right) = \sum_{i=1}^{N} ln\left(\frac{1}{2\pi\sigma^2} e^{\frac{-1}{2}\frac{(x_i-\mu)^2}{\sigma^2}}\right) = \sum_{i=1}^{N} ln\left(\frac{1}{2\pi\sigma^2}\right) - \sum_{i=1}^{N} \frac{1}{2}\frac{(x_i-\mu)^2}{\sigma^2} \tag{5}$$

$$\frac{\partial ln(P(X|\mu))}{\partial\mu} = \frac{\partial \sum_{i=1}^{N} ln\left(\frac{1}{2\pi\sigma^2}\right)}{\partial\mu} - \frac{\partial \sum_{i=1}^{N} \frac{1}{2}\left(\frac{(x_i-\mu)^2}{\sigma^2}\right)}{\partial\mu} \tag{6}$$

$$0 + \sum_{i=1}^{N} \frac{(x_i-\mu)}{\sigma^2} = \sum_{i=1}^{N} \frac{(x_i-\mu)}{\sigma^2} \tag{7}$$

$$\frac{\partial ln(P(X|\mu))}{\partial\mu} = \sum_{i=1}^{N} \frac{(x_i-\mu)}{\sigma^2} = 0 \Rightarrow \mu = \frac{\sum_{i=1}^{N} x_i}{N} \tag{8}$$

By minimizing the error function, we maximize the probability function (likelihood) L; both have a Gaussian distribution with the mean $w^T X$ and variance σ^2 given by Equation (9):

$$\text{y} \sim N\left(\omega^T X, \sigma^2\right) o\ \text{y} = \omega^T X + \varepsilon \tag{9}$$

$\varepsilon \sim N(0, \sigma^2)$ ε is the Gaussian noise that has a zero mean and a variance of σ^2. This is interpreted considering that, in linear regression, the errors are of the Gaussian type and have a linear trend. For new or atypical values, the prediction index decreases, so L2 regularization, also called Ridge, was used, modifying the cost function and penalizing the largest errors, as shown in Equation (10).

$$J_{RIDGE} = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 + \lambda|\omega|^2 \tag{10}$$

The λ refers to the penalty of the model, with

$$|\omega|^2 = \omega^T.\omega = \omega^2.1 + \omega^2.2 + \ldots + \omega^2.D \tag{11}$$

There will be two probabilities:

A posterior:

$$P(Y|X,\omega) = \prod_{i=1}^{N} \frac{1}{2\pi\sigma^2} e^{\frac{-1}{2}\frac{(y_n-\omega^T x_n)^2}{\sigma^2}} \tag{12}$$

A priori:

$$P(\omega) = \frac{\lambda}{\sqrt{2\pi}} e^{(-\frac{\lambda}{2}\omega^T\omega)} \tag{13}$$

For LASSO, we have

$$J_{LASSO} = \sum_{n=1}^{N} (y_i - \hat{y}_i)^2 + \lambda\|\omega\| \tag{14}$$

Maximizing the likelihood:

$$P(Y|X,\omega) = \prod_{n=1}^{N} \frac{1}{2\pi\sigma^2} e^{\frac{-1}{2}\frac{(y_n-\omega^T x_n)^2}{\sigma^2}} \tag{15}$$

and Prior (previous) are given by

$$P(\omega) = \frac{\lambda}{2} e^{(-\lambda|\omega|)} \tag{16}$$

so that

$$J = (Y - X_\omega)^T (Y - X_\omega) + \lambda|\omega| \tag{17}$$

and

$$\frac{\partial J}{\partial \omega} = -2X^T Y + 2X^T Y + 2X^T X\omega + \lambda sign(\omega) = 0 \tag{18}$$

where

$$(\omega) = 1\ if\ x > 0\ and\ -1\ if\ x < 0\ and\ 0\ if\ x = 0 \tag{19}$$

The two previous penalties are combined with their respective indices, as shown in Equation (20):

$$J_{ELASTICNET} = \sum_{n=1}^{N} (y_i - \hat{y}_i)^2 + \lambda|\omega|^2 + \sum_{n=1}^{N} (y_i - \hat{y}_i)^2 + \lambda\|\omega\| \tag{20}$$

Expressed as a function of the penalties in Equation (21):

$$ELASTICNET_{PENALTY} = (\alpha * L_{1PENALTY}) + ((1-\alpha) * L_{2PENALTY}) \tag{21}$$

The independent variables have different influences in the regression model, considering the algorithm used and the regularization techniques applied; the influence of each one of the independent variables is represented in a Graph called Label Importance [32,33] For the base model (ElasticNet), Figure 4 shows the importance of each variable in the model.

Figure 4 shows that the most influential variable is Module temp, followed by the Power factor and the Apparent power, which implies the correct discrimination of the models with respect to the input variables.

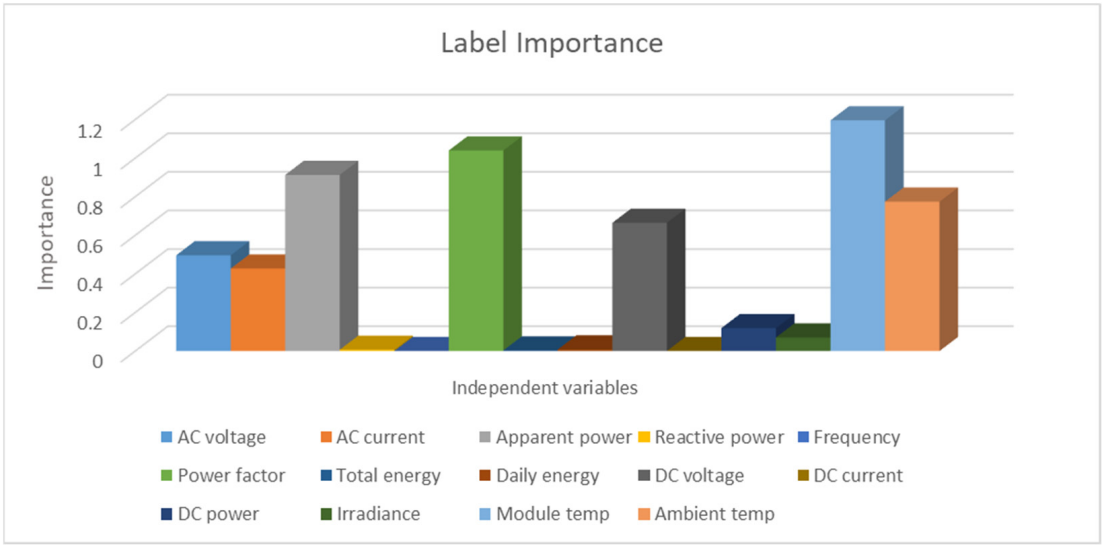


Figure 4. Label importance of the variables in the base model.

2.3. Stacking

The probability that a single model captures the patterns in the data is low, so integrating multiple models can result in optimal predictions. Each model that is added will be able to capture more characteristics of the data, and by combining all the models, a meta-model is obtained, which contains more information on the characteristics of the data. This action of joining different models is called an assembly, and there are three types: bagging, reinforcement, and stacking. Bagging reduces variance, whereas reinforcement reduces bias, and both types are susceptible to noisy and outlier data, while stacking corrects these disadvantages. Figure 5 shows the stacking flowchart.

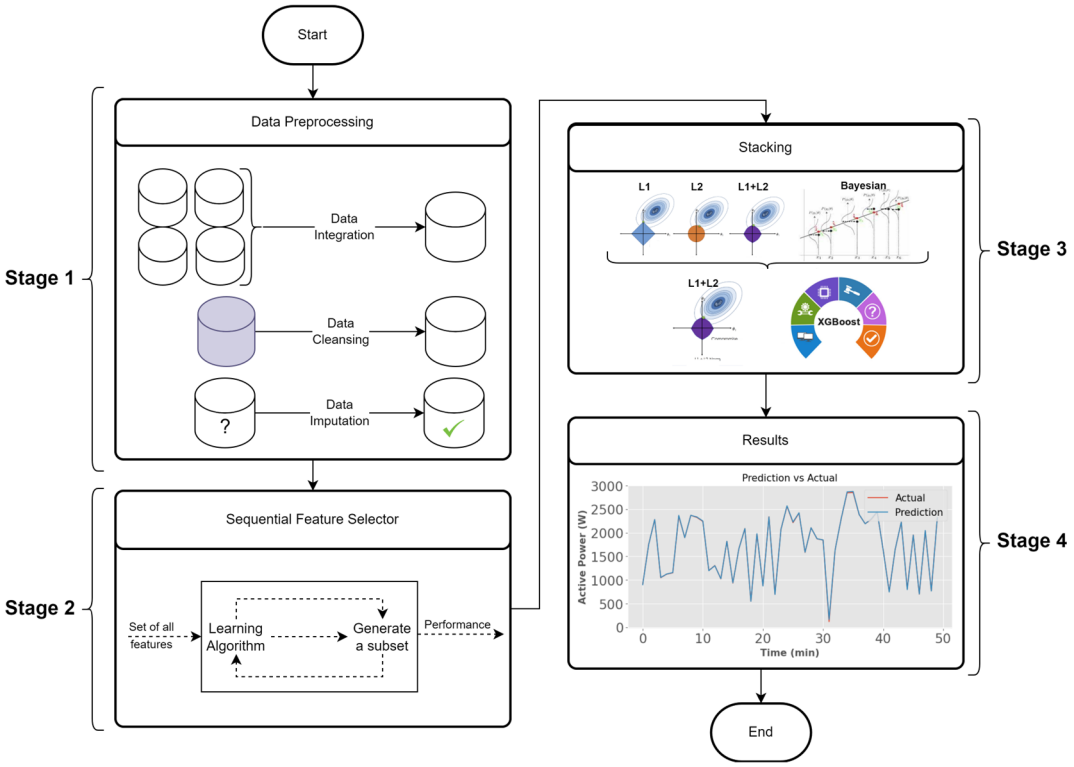


Figure 5. Flowchart of the stacking model for regression.

Feature selection was used to reduce the computational cost and optimize the algorithm; the technique used for this selection process was Wrapper, which improves accuracy because it takes into account the relationships between the independent variables and not only its content as the filtering technique is performed. In this way, it generalizes the results better, avoiding overfitting when using cross-validation techniques.

For the SFS algorithm shown in Figure 6, the data are first split using cross-validation techniques, and the SFS algorithm initially feeds n variables ($n = 1$), adding more variables iteratively ($n + 1$) later on until reaching the maximum number of variables available. Of all these combinations, nine variables were obtained according to the negative mean squared error.

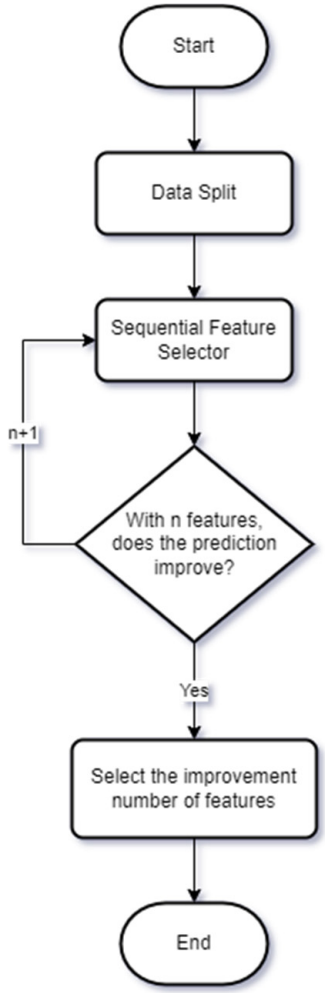
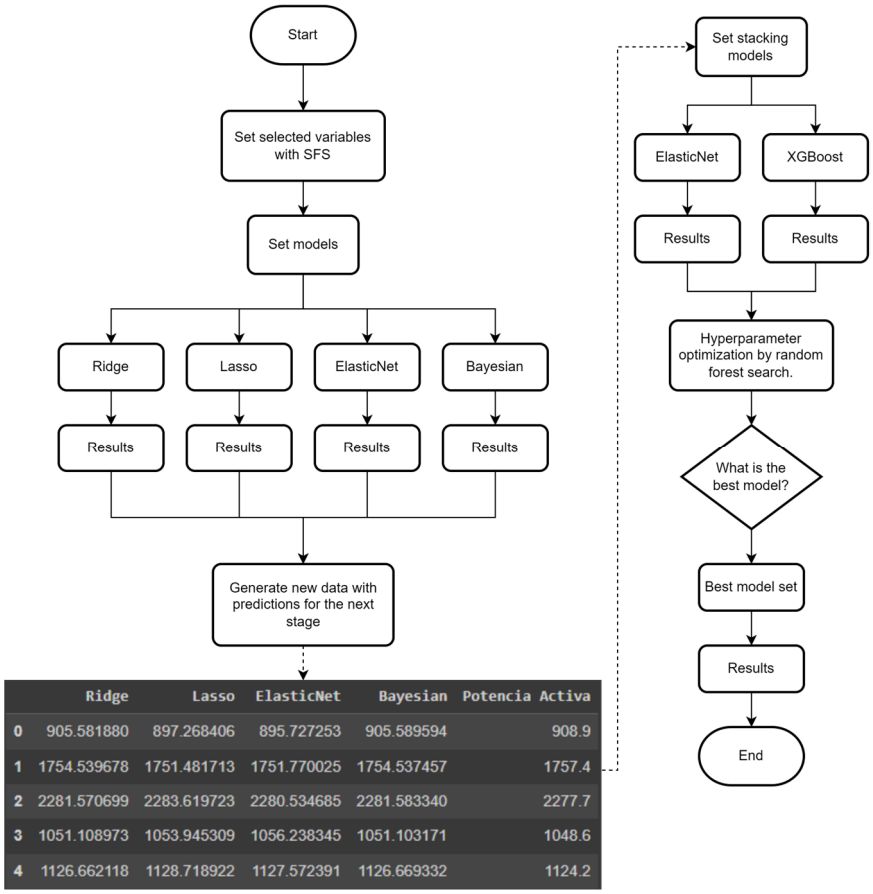


Figure 6. SFS algorithm flowchart.

Figure 7 shows stage three of the stacking model for the regression proposed, whereby after obtaining the variables provided by the SFS algorithm, these are trained with the following algorithms in level one: Ridge, LASSO, ElasticNet, and Bayesian, obtaining a hybrid feature selection model. Then, a new dataset is generated using the independent variables from the prediction of the four trained models. In level two, the ElasticNet and XGBoost algorithms are used to perform the stacking. Finally, the search for hyperparameters is carried out by applying random forest research to obtain the meta-model.



	Ridge	Lasso	ElasticNet	Bayesian	Potencia Activa
0	905.581880	897.268406	895.727253	905.589594	908.9
1	1754.539678	1751.481713	1751.770025	1754.537457	1757.4
2	2281.570699	2283.619723	2280.534685	2281.583340	2277.7
3	1051.108973	1053.945309	1056.238345	1051.103171	1048.6
4	1126.662118	1128.718922	1127.572391	1126.669332	1124.2

Figure 7. Hyperparameter search and stacking flowchart.

2.4. Optimization

Hyperparameter optimization for machine learning models has become an essential part of applying these models to forecasting problems in the application of renewable energy. Hyperparameters are variables that control learning processes in an automatic model. In other words, they guide the model on how to learn the specific relationships between the input and output variables; in this case, to make predictions on the photovoltaic system. As components of artificial intelligence techniques, hyperparameters are not given by one-size-fits-all values or recipes, as they will vary from model to model. An ideal or optimal model cannot be found, but models that minimize the error or loss function can be found. All optimization models are limited by processing capacity and available time, so searching requires techniques that take these factors into account. The authors of [34,35] performed two types of searches: grid search and random search. A combination of both is currently used with influences from other techniques, such as swarm algorithms, to improve searching.

Among the methods currently used are the following:

- Bayesian method: this method seeks global optimization through the iterative construction of a probabilistic model of the distribution of functions from the values of the hyperparameters to the objective function. Probabilistic models capture beliefs about the behavior of a function by constructing the posterior distribution of the objective function. The acquisition function is then constructed using the posterior distribution to determine the next point with the best probability of improvement. This type of optimization has the disadvantage of exploration and exploitation; that is, finding a balance between global searches to find the best solution in all the available space and local searching to refine the results and try to avoid wasting resources. Within this type of algorithm, we have Parzen, Gaussian process regressor, and kriging.

- Early stop method: by using statistical searches, this method discards the search spaces that offer the worst results and that do not contain a global minimum. Its result is given based on the comparison of the intermediate scores of the model with the set of hyperparameters. As an example, we have halving and hyperband. Their main disadvantage is that they have to go through the entire space to deliver their final result.
- Evolutionary method: based on the principles of evolution given by Charles Darwin, this method begins by extracting an initial sample of the hyperparameter search space to later evaluate them based on their relative fitness. The worst-performing hyperparameters are discarded, and new sets of hyperparameters are generated through crossover and mutation. This process is repeated until no further growth is observed in the results or the process stops due to processing time.
- Bayesian optimization methods: in order to solve exploration and exploitation problems, two strategies are used: sampling, working in areas where better results are obtained, and pruning, which is based on stopping early if optimal results are found.
- The flowchart used for optimization in this investigation is shown in Figure 8:

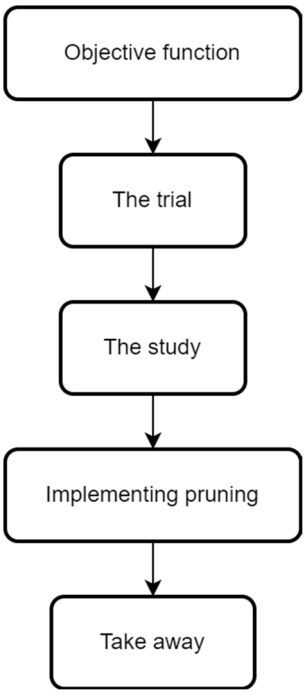


Figure 8. Optimization process flowchart.

In Figure 8, the objective function that sets the general training and testing conditions of the model is first defined. In the trial, the information on the hyperparameters to be optimized is stored. In the study, the objective function is optimized, and the best combination for the hyperparameters is determined. The process is iterative until a target value is reached or a user-defined maximum time is reached. In the implementation of pruning to obtain intermediate results, the calculation time can be reduced. In "Take away", algorithms are implemented to speed up the hyperparameter adjustment process through the use of a machine learning pipeline or parallelization.

2.5. Performance Evaluation

In order to validate the performance of the proposed models, the following metrics will be used:

- R^2 score: indicates the precision of the model in terms of residual distance; it is used as an equivalent metric of the classification models. Its main advantage is that it allows models to be compared more easily, but its disadvantage is that, when working with

many variables, it tends to overfit the model, obtaining very high R^2 values. R^2 is given by Equation (22):

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \quad (22)$$

- Mean absolute error (MAE): this is the sum of all the differences between the actual and predicted values divided by the total amount of data. It is used to understand how close the predictions are with respect to the real model calculated on average. Its main advantage is that it is a differentiable function, which is why it is used as a loss function to be minimized. Its disadvantage is that it is affected by outliers, and it is difficult to interpret; that is, interpreting between which ranges the result is acceptable. MAE is given by the following equation:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \quad (23)$$

- Mean squared error (MSE): this is defined as the mean square distance. It works like MAE, but it is used when there are large errors in the prediction, making them noticeable in the total value. Its main advantage is that it is also differentiable and is usually easier to interpret than MSE. Its main disadvantage is that it is greatly affected by outliers. Its equation is as follows:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \quad (24)$$

- Adjusted R^2 score: this is used to penalize a model that has too many independent variables that are not significant for the prediction. The main advantage is that it can indicate overfitting in the model, but the disadvantage is that it is affected by highly biased models. Its equation is as follows:

$$R^2_{adj} = 1 - \left[\frac{(1 - R^2)(n-1)}{n-k-1}\right] \quad (25)$$

3. Results

3.1. Dataset

During the period from April to August 2021, data were collected in the department of Puno, located at 15°29′20″ S, 70°9′6″ W, and at an extreme altitude of 3800 m above sea level, using the StecaGrid 3010 inverter from the German company StecaElectronics GmbH. The variables analyzed were AC voltage, AC current, Active power, Apparent power, Reactive power, Frequency, Power factor, Total energy, Daily energy, DC voltage, DC current, DC power, Irradiance, Module temp, and Ambient temp. The PT1000 sensors measured the ambient temperature and the temperature of the photovoltaic panel, whereas the Atersa-brand calibrated cell measured the irradiance without taking temperature as a parameter. All equipment worked in accordance with the European standard IEC 61724-20170. A total of 320,281 values were collected, which were reduced to 119,753 because of null values, such as those obtained at night. The statistics of the collected data (median, standard deviation, maximum and minimum values, and interquartile ranges) are presented in Table 1.

Table 1. Statistics of the variables used.

	Count	Mean	Std	Min	25%	50%	75%	Max
AC voltage (V)	119,753.0	235.440155	2.940976	223.900000	233.400000	235.400000	237.600000	247.900000
AC current (A)	119,753.0	6.964209	2.931848	0.580000	4.633000	7.558000	9.430000	12.416000
Active power (W)	119,753.0	1621.629792	708.451947	0.000000	1070.000000	1762.500000	2219.200000	2879.200000
Apparent power (W)	119,753.0	1642.922003	696.644029	135.000000	1089.800000	1777.900000	2232.500000	2898.000000
Reactive power (W)	119,753.0	220.097268	66.293499	−843.900000	196.300000	228.500000	256.300000	485.100000
Frequency (Hz)	119,753.0	60.002564	0.046009	59.500000	60.000000	60.000000	60.000000	60.500000
Power factor	119,753.0	0.951686	0.188032	−0.990000	0.983000	0.991000	0.994000	0.998000
Total energy (W/h)	119,753.0	5224.543496	1013.532902	3894.300000	4183.400000	5904.500000	6171.200000	6427.600000
Daily energy (W/h)	119,753.0	127.544866	86.504498	0.000209	56.418152	113.251184	189.551443	342.905747
DC voltage (V)	119,753.0	334.812376	17.338490	220.800000	321.900000	332.800000	346.100000	420.800000
DC current (A)	119,753.0	5.556741	2.390009	0.000000	3.620000	5.890000	7.650000	10.780000
DC power (W)	119,753.0	1831.112472	737.393141	0.000000	1260.304000	1972.189000	2450.420000	3142.272000
Irradiance (W/m^2)	119,753.0	668.877765	292.047458	0.000000	432.000000	706.000000	926.000000	1522.000000
Module temp (°C)	119,753.0	35.115793	11.256891	2.400000	27.600000	37.000000	44.200000	60.300000
Ambient temp (°C)	119,753.0	16.611160	3.769045	−2.000000	14.500000	17.400000	19.400000	27.700000

3.2. Feature Selection

Figure 9 shows the behavior of the SFS algorithm between the number of variables and the performance. The best performance was obtained with nine variables: AC voltage, AC current, Apparent power, Reactive power, Power factor, DC voltage, DC current, DC power, and Ambient temp, taking into account the negative mean squared error. This graph shows the notable improvement in the feature selection through this selection of variables to optimize the regression model.

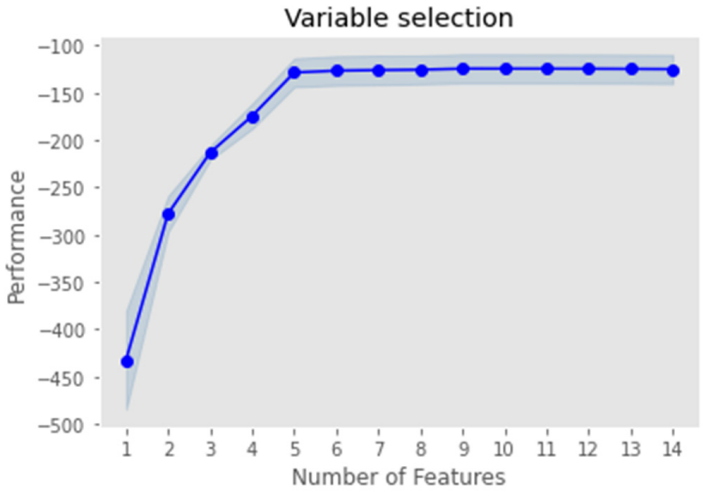


Figure 9. SFS algorithm behavior curve.

3.3. Stacking: Level One

From Figure 10 for the Ridge model, the importance variables are AC current, Power factor, and DC current. According to Figure 11 for the LASSO model, the most important variables are AC current, Power factor, and AC voltage. According to Figure 12 for the ElasticNet model, Power factor, Ambient temp, and Apparent power are the most significant variables. According to Figure 13 for the Bayesian model, AC current, Power factor, and DC current are the most important variables.

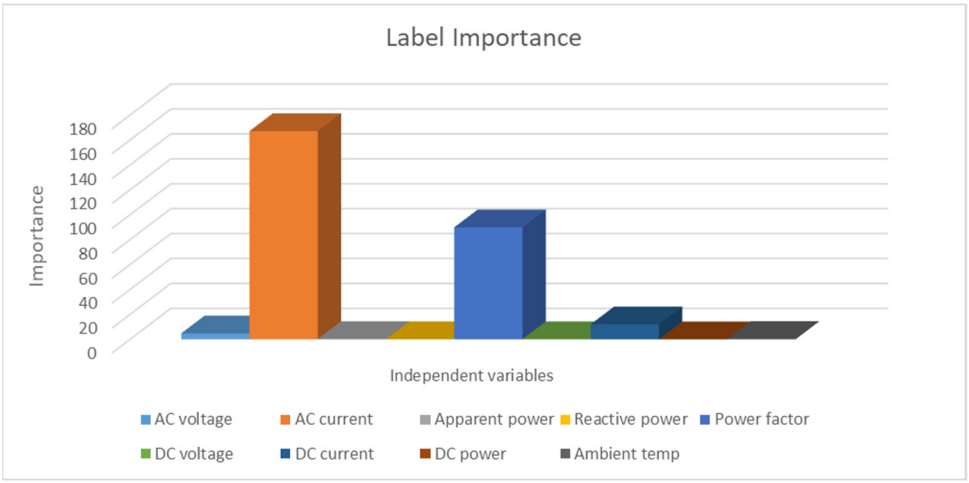


Figure 10. Label importance of the variables: Ridge model.

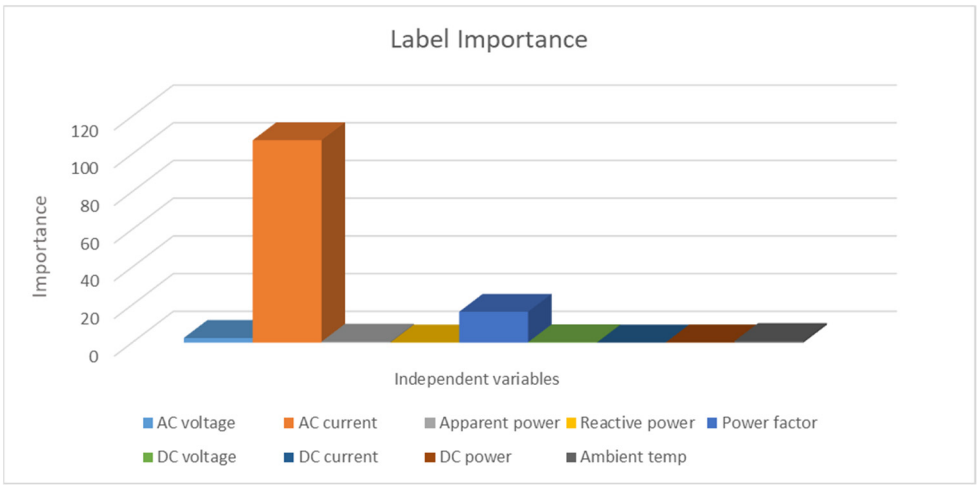


Figure 11. Label importance of the variables: LASSO model.

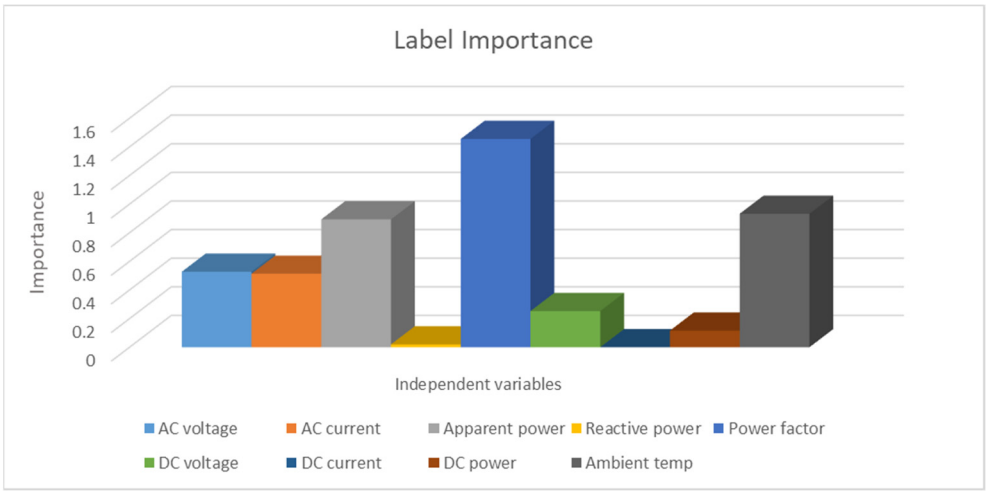


Figure 12. Label importance of the variables: ElasticNet model.

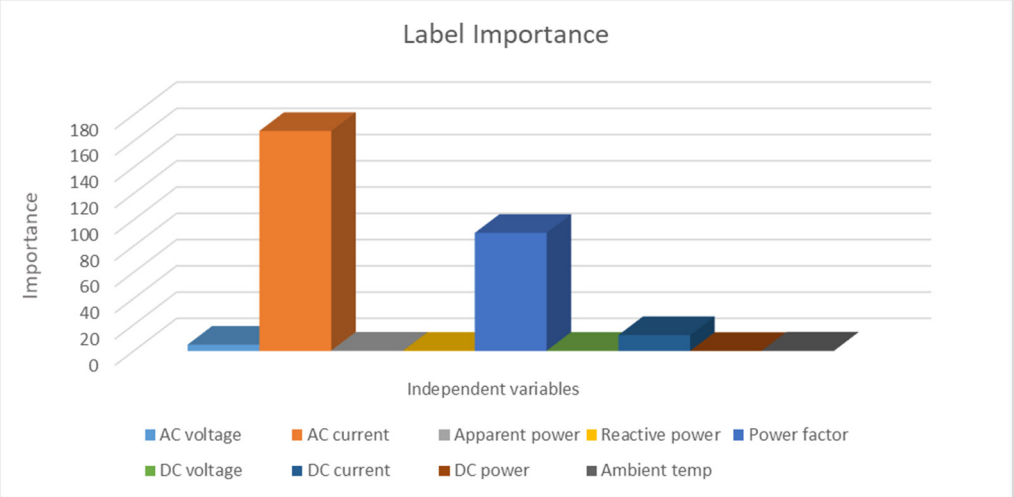


Figure 13. Label importance of the variables: Bayesian model.

Figures 10–13 show that each model has a different importance label, which guarantees that the stacking model collects different characteristics from the different models; this allows for better predictions by capturing more complex patterns. Additionally, [36] indicates that the "important labels" highlight the characteristics of the models that help to better understand the data set and the proposed model, and [37] uses the "important labels" to determine the importance of each model on the meta-model obtained and that ponders each specific characteristic. Finally, [38] uses "label important" to note the effect of the variables on two models with regularizations L1 and L2 for the selection of characteristics.

In order to determine the efficiency of the models, the MAE metric was mainly used, which is the average difference between the actual and forecast data. This metric was chosen because it is more robust when the data have some outliers, such as the data of the present investigation.

Table 2 and Figures 14 and 15 contain the results for the dependent variable "Active Power" of the models with different metrics; the Ridge model had the best results, with a Score of 99.96%, an MAE of 6.017, an MSE of 10.64, a coefficient of determination of 99.98%, and an adjusted coefficient of determination of 99.94%, presenting a reduction in the absolute mean error; 30.11% compared to the base model.

Table 2. Results of stacking: level one.

	Model Base	Ridge	LASSO	ElasticNet	Bayesian
MSE	17.48539	10.63716	14.40150	18.33607	10.63832
R^2	0.99939239	0.99977513	0.99958782	0.99933183	0.99977508
R^2_{adj}	0.99939232	0.99977512	0.99958778	0.99933178	0.99977507

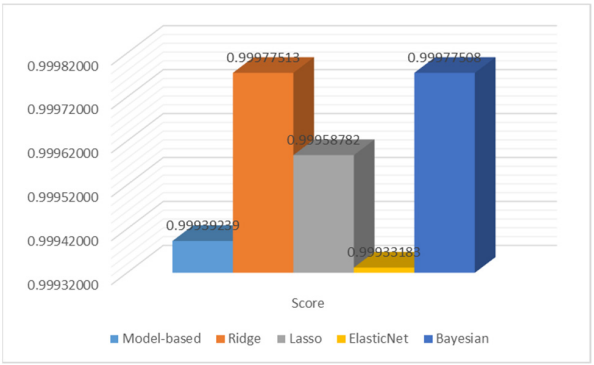


Figure 14. Results of stacking: level one: Score.

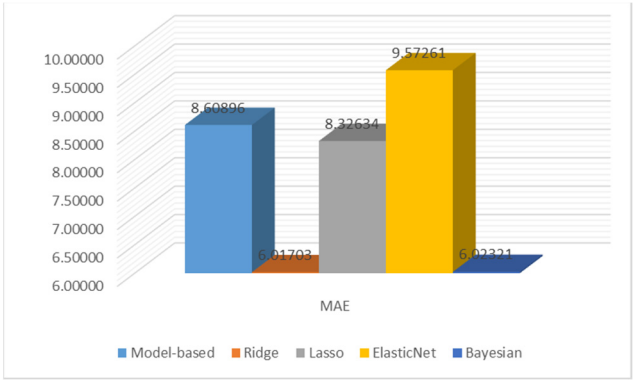


Figure 15. Results of stacking: level one: MAE.

According to Figure 14, the Ridge model has the best score over the Bayesian, LASSO, and ElasticNet models. In Figure 15, the model that has the smallest absolute error measured is Ridge, followed by Bayesian, LASSO, and ElasticNet.

3.4. Stacking: Level Two

After training the four models with the data, level-two stacking was carried out with the creation of a new dataset, where the input variables or independent variables would now be the predictions of the four models and the output variable, or the dependent variable would continue to be the Active power. For level two, two models were generated: ElasticNet and XGBoost, with the aim of capturing complex patterns through the predictions of level one. Table 3 and Figures 16 and 17 show the results obtained for the dependent variable "Active Power".

Table 3. Stacking results: level two.

	Model Base	**ElasticNet**	**XGBoost**
MSE	17.48539	10.63704	10.06582
R^2	0.99939239	0.99977514	0.99979864
R^2_{adj}	0.99939232	0.99977512	0.99979862

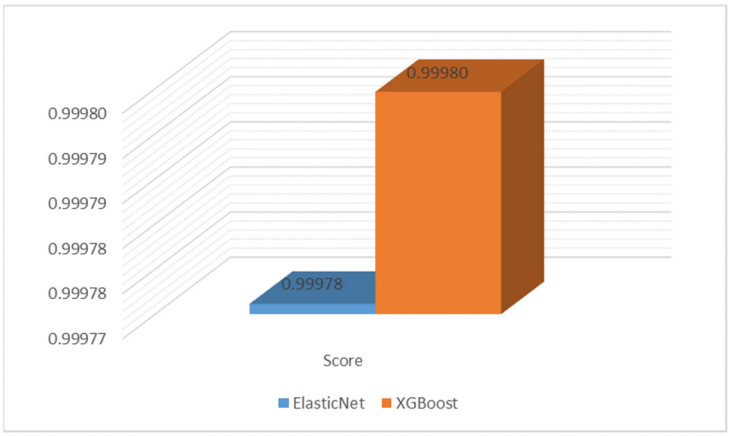


Figure 16. Results of stacking: level two: Score.

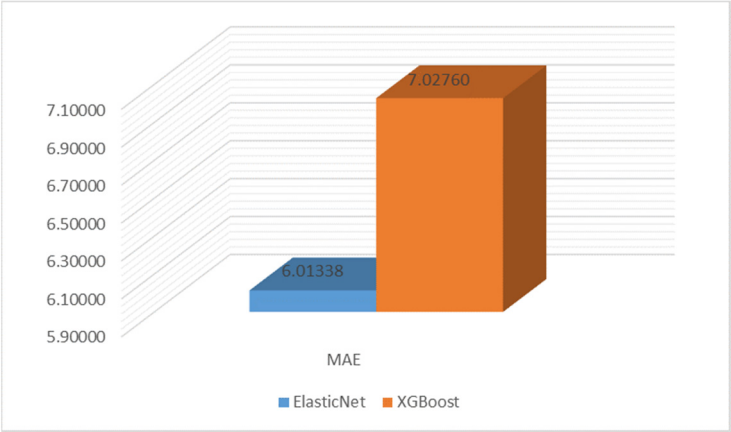


Figure 17. Results of stacking: level two: MAE.

From Table 3 and Figures 16 and 17, it can be seen that the two new level-two stacking models improved in terms of all the parameters of the base model. Likewise, in Figure 17, it is shown that the model with the lowest mean absolute error is ElasticNet, and in Figure 16, the model with the best score is XGBoost when compared to ElasticNet. Equation (26) was used to calculate the percentage increase or decrease between the base model and the new model.

$$Percentage\ of\ improvement = \frac{(Base\ Model - New\ Model)}{Base\ Model} * 100 \quad (26)$$

From Table 3, it can be seen when making a comparison between the ElasticNet model and the base model using Equation (26), there is an improvement in Score of 0.038%, a decrease in MAE and MSE of 30.150% and 39.166%, respectively, and an improvement in the coefficient of determination and the adjusted coefficient of determination of 0.038% in both cases.

3.5. Stacking Meta-Model

After performing the level-two stacking, the hyperparameters of the ElasticNet and XGBoost models were adjusted using the modified random forest research technique, which combines the advantages of several types of optimization methods, such as Bayesian and decision trees. A "trial" was set, which contained the hyperparameters to be optimized and determined by functions such as "int" (iteration of integer values), "float" (iteration of floating values), and "categorical" (iteration of categorical values); the objective function determines the improvement (or lack thereof) of the model. In this investigation, the mean absolute error was used as the objective function.

The search for hyperparameters in the XGBoost model was performed with 'n_estimators', 'max_depth', 'reg_alpha', and 'reg_lambda', with an iteration of integer types between the ranges 10–1000, 3–20, 0–1, and 0–1, respectively. Additionally, a float-type iteration was used for the 'learning_rate' and 'gamma' hyperparameters, setting values of 0.001–1.0 and 10^{-8}, 1.0.

In order to determine the best meta-model, different metrics were calculated to observe the behavior of both models for the dependent variable "Active Power", as shown in Table 4 and Figures 18 and 19.

Table 4. Stacking results: level two with hyperparameter adjustment.

	Base Model	**ElasticNet**	**XGBoost**
MSE	17.48539	10.63704	10.63698
R^2	0.99939239	0.999775137	0.999775140
R^2_{adj}	0.99939232	0.999775116	0.999775118

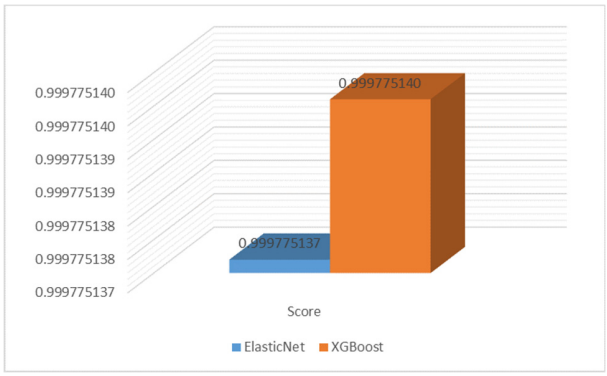


Figure 18. Results of the meta models: Score.

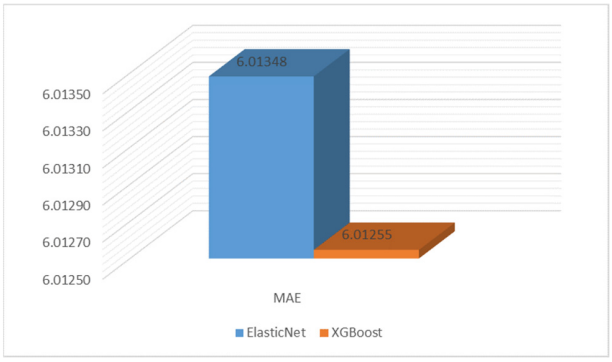


Figure 19. Results of the meta models: MAE.

R^2_{adj} Regarding the meta models, it is shown that XGBoost has a better score (Figure 18). In Figure 19, the XGBoost model has a lower mean absolute error, which makes it a more optimal model.

From Table 4, it is noted that the XGBoost model has an improvement of 0.038% with respect to the base model; the error metrics, such as the MAE and MSE, have decreases of 30.160% and 39.166%, respectively; and finally, the coefficient of determination and the adjusted coefficient of determination show an improvement of 0.0383%.

Figure 20 shows the best models for each level of stacking, with Ridge in level one representing the model with the highest error, followed by ElasticNet, where the stacking was carried out without a hyperparameter search, and finally, the XGBoost model, which has the lowest error using hyperparameter searching.

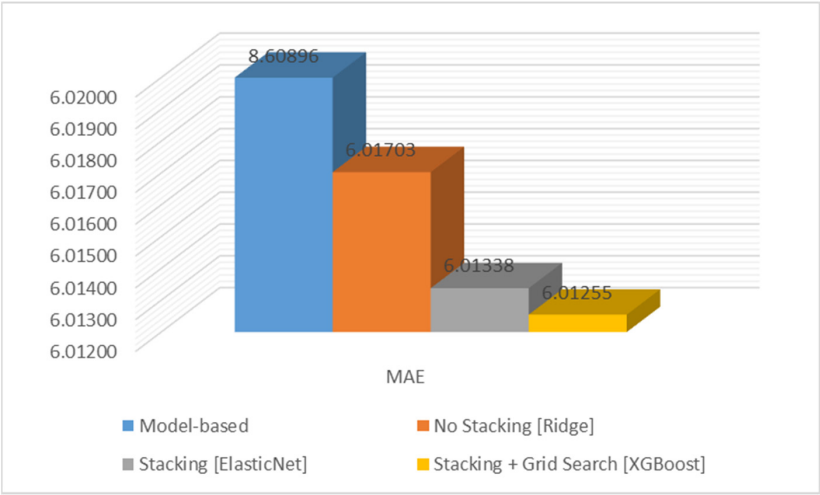


Figure 20. Results of the best models in each level of stacking.

Figure 21 shows the data from the prediction made by the meta-model (XGBoost), where it can be seen that the fit of the model is optimal and contains very few errors, with a low deviation.

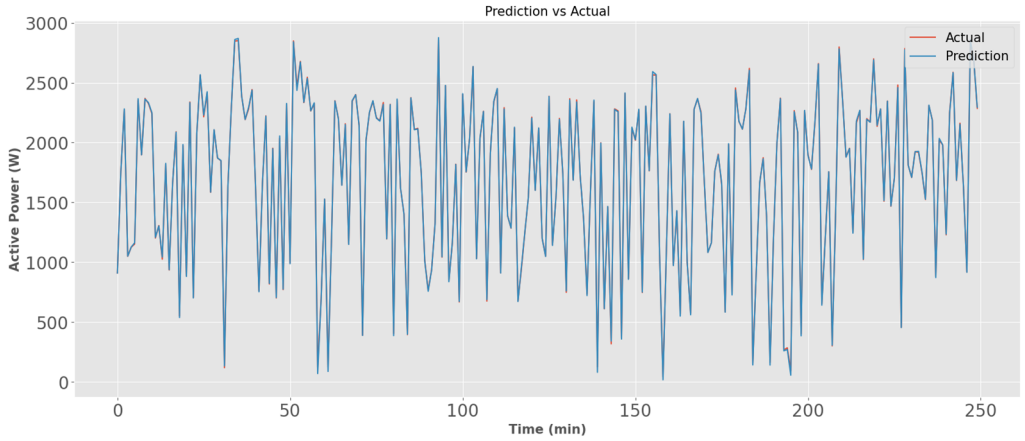


Figure 21. Real data vs. predicted data.

Figure 22 also shows the R^2 of the meta-model, which expresses the current value on the X-axis and the predictive value on the Y-axis. The image shows a direct and optimal relationship between the behavior of the meta-model for these two variables.

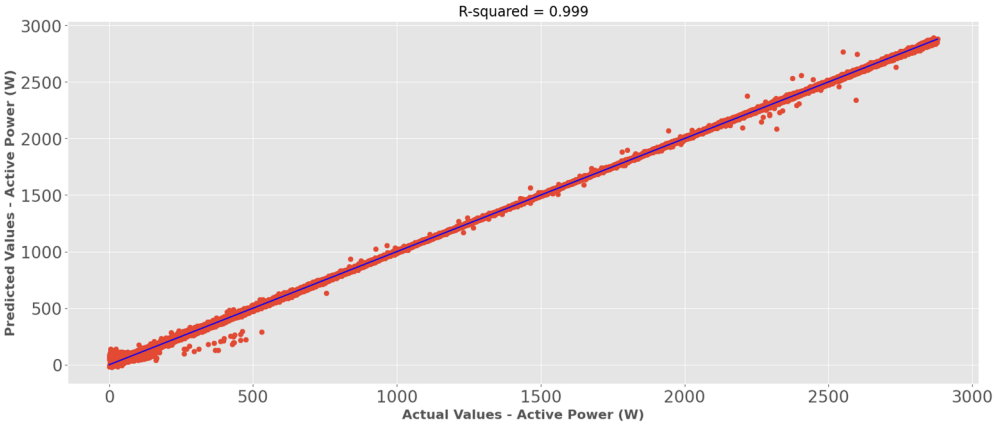


Figure 22. Prediction data vs. actual data.

4. Conclusions

Because the factors that affect the production of solar energy at altitudes higher than 3800 m above sea level are complex and varied, such as climatic variations, extreme solar radiation, and hostile environments, the production of solar energy is not constant and is difficult to predict. For this reason, it is necessary to carry out predictions using models for overall efficient use. Regression models only generally collect a few features, so it is necessary to implement models that collect the largest number of features; this is carried out by using stacking techniques. In this study, two stacking techniques—ElasticNet and XGBoost—were presented, both taking LASSO, Ridge, ElasticNet, and Bayesian as the base or level one in the models, to which SFS feature selection was applied to reduce computational cost and optimize the algorithm. The models were implemented using data from a string photovoltaic system under the European standard IEC 61724-20170 in the months of April and August 2021 in the department of Puno, Peru, using 15 variables between the atmospheric and typical conditions of the photovoltaic system. As a result, the mean absolute error (MAE) was reduced by 30.15% using ElasticNet with respect to the base model. In order to obtain better results, hyperparameter optimization processes were carried out using the modified random forest research technique; the mean absolute error

was reduced by 30.16% using the XGBoost meta-model with respect to the base model. It is concluded that the proposed models reduce the error of the prediction system, especially the stacking model that used XGBoost and hyperparameter optimization. The results can be improved by implementing other regression techniques both in level one and level two, such as models based on neural networks.

Author Contributions: Conceptualization, J.C., W.M. and S.H.; methodology, J.C., W.M. and O.V.; software, W.M., N.B. and C.R.; validation, J.C., C.R. and S.H.; formal analysis, N.B, C.R. and O.V.; investigation, J.C., W.M. and C.R.; resources, J.C., W.M. and N.B.; data curation, J.C., W.M. and O.V.; writing—original draft preparation, J.C., W.M. and S.H.; writing—review and editing, J.C., W.M., O.V. and S.H.; visualization, J.C., W.M. and C.R.; supervision, J.C., W.M. and S.H.; project administration, J.C., W.M. and O.V.; funding acquisition, S.H. and N.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Acknowledgments: The authors wish to thank the Universidad Nacional del Altiplano and Universidad Nacional de Moquegua.

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

References

1. Dubey, S.; Sarvaiya, J.N.; Seshadri, B. Temperature Dependent Photovoltaic (PV) Efficiency and Its Effect on PV Production in the World—A Review. *Energy Procedia* **2013**, *33*, 311–321. [CrossRef]
2. Gupta, V.; Sharma, M.; Pachauri, R.K.; Babu, K.D. Comprehensive review on effect of dust on solar photovoltaic system and mitigation techniques. *Sol. Energy* **2019**, *191*, 596–622. [CrossRef]
3. Aglietti, G.S.; Redi, S.; Tatnall, A.R.; Markvart, T. Harnessing high-altitude solar power. *IEEE Trans. Energy Convers.* **2009**, *24*, 442–451. [CrossRef]
4. Ebhota, W.S.; Tabakov, P.Y. Impact of Photovoltaic Panel Orientation and Elevation Operating Temperature on Solar Photovoltaic System Performance. *Int. J. Renew. Energy Dev.* **2022**, *11*, 591–599. [CrossRef]
5. Li, G.; Xie, S.; Wang, B.; Xin, J.; Li, Y.; Du, S. Photovoltaic Power Forecasting with a Hybrid Deep Learning Approach. *IEEE Access* **2020**, *8*, 175871–175880. [CrossRef]
6. El Motaki, S.; El Fengour, A. A statistical comparison of feature selection techniques for solar energy forecasting based on geographical data. *Comput. Assist. Methods Eng. Sci.* **2021**, *28*, 105–118.
7. Nejati, M.; Amjady, N. A New Solar Power Prediction Method Based on Feature Clustering and Hybrid-Classification-Regression Forecasting. *IEEE Trans. Sustain. Energy* **2022**, *13*, 1188–1198. [CrossRef]
8. Castangia, M.; Aliberti, A.; Bottaccioli, L.; Macii, E.; Patti, E. A compound of feature selection techniques to improve solar radiation forecasting. *Expert Syst. Appl.* **2021**, *178*, 114979. [CrossRef]
9. Macaire, J.; Salloum, M.; Bechet, J.; Zermani, S.; Linguet, L. Feature Selection using Kernel Conditional Density Estimator for day-ahead regional PV power fore-casting in French Guiana. In Proceedings of the International Conference on Applied Energy, Bangkok, Thailand, 29 November–2 December 2021.
10. Zambrano, A.F.; Giraldo, L.F. Solar irradiance forecasting models without on-site training measurements. *Renew. Energy* **2020**, *152*, 557–566. [CrossRef]
11. Huaquipaco, S.; Macêdo, W.N.; Pizarro, H.; Condori, R.; Ramos, J.; Vera, O.; Cruz, J.; Mamani, W. Cross-validation of the operation of photovoltaic systems connected to the grid in extreme conditions of the highlands above 3800 meters above sea level. *Int. J. Renew. Energy Res.* **2022**, *12*, 950–959. [CrossRef]
12. Zhao, Z.; Chen, K.; Chen, Y.; Dai, Y.; Liu, Z.; Zhao, K.; Wang, H.; Peng, Z. An Ultra-Fast Power Prediction Method Based on Simplified LSSVM Hyperparameters Optimization for PV Power Smoothing. *Energies* **2021**, *14*, 5752. [CrossRef]
13. Andrade, C.H.; Melo, G.C.; Vieira, T.F.; Araújo, Í.B.; Medeiros Martins, A.D.; Torres, I.C.; Brito, D.B.; Santos, A.K. How Does Neural Network Model Capacity Affect Photovoltaic Power Pre-diction? A Study Case. *Sensors* **2023**, *23*, 1357. [CrossRef] [PubMed]
14. Silva, R.C.; de Menezes Júnior, J.M.; de Araújo Júnior, J.M. Optimization of narx neural models using particle swarm optimization and genetic algorithms applied to identification of photovoltaic systems. *J. Sol. Energy Eng.* **2021**, *143*, 051001. [CrossRef]
15. Harrou, F.; Taghezouit, B.; Khadraoui, S.; Dairi, A.; Sun, Y.; Arab, A.H. Ensemble Learning Techniques-Based Monitoring Charts for Fault Detection in Photovoltaic Systems. *Energies* **2022**, *15*, 6716. [CrossRef]
16. Pravin, P.; Tan, J.Z.M.; Yap, K.S.; Wu, Z. Hyperparameter optimization strategies for machine learning-based stochastic energy efficient scheduling in cyber-physical production systems. *Digit. Chem. Eng.* **2022**, *4*, 100047. [CrossRef]

17. Tina, G.M.; Ventura, C.; Ferlito, S.; De Vito, S. A State-of-Art-Review on Machine-Learning Based Methods for PV. *Appl. Sci.* **2021**, *11*, 7550. [CrossRef]
18. Mosavi, A.; Salimi, M.; Ardabili, S.F.; Rabczuk, T.; Shamshirband, S.; Varkonyi-Koczy, A.R. State of the Art of Machine Learning Models in Energy Systems, a Systematic Review. *Energies* **2019**, *12*, 1301. [CrossRef]
19. Lodhi, E.; Wang, F.-Y.; Xiong, G.; Dilawar, A.; Tamir, T.S.; Ali, H. An AdaBoost Ensemble Model for Fault Detection and Classification in Photovoltaic Arrays. *IEEE J. Radio Freq. Identif.* **2022**, *6*, 794–800. [CrossRef]
20. Chen, B.; Wang, Y. Short-Term Electric Load Forecasting of Integrated Energy System Considering Nonlinear Synergy Between Different Loads. *IEEE Access* **2021**, *9*, 43562–43573. [CrossRef]
21. Khan, W.; Walker, S.; Zeiler, W. Improved solar photovoltaic energy generation forecast using deep learning-based ensemble stacking approach. *Energy* **2022**, *240*, 122812. [CrossRef]
22. Abdellatif, A.; Mubarak, H.; Ahmad, S.; Ahmed, T.; Shafiullah, G.M.; Hammoudeh, A.; Abdellatef, H.; Rahman, M.M.; Gheni, H.M. Forecasting Photovoltaic Power Generation with a Stacking Ensemble Model. *Sustainability* **2022**, *14*, 11083. [CrossRef]
23. Feng, Y.; Yu, X. Deployment and Operation of Battery Swapping Stations for Electric Two-Wheelers Based on Machine Learning. *J. Adv. Transp.* **2022**, *2022*, 8351412. [CrossRef]
24. Lateko, A.A.H.; Yang, H.-T.; Huang, C.-M. Short-Term PV Power Forecasting Using a Regression-Based Ensemble Method. *Energies* **2022**, *15*, 4171. [CrossRef]
25. Guo, X.; Gao, Y.; Zheng, D.; Ning, Y.; Zhao, Q. Study on short-term photovoltaic power prediction model based on the Stacking ensemble learning. *Energy Rep.* **2020**, *6*, 1424–1431. [CrossRef]
26. Abdelmoula, I.A.; Elhamaoui, S.; Elalani, O.; Ghennioui, A.; El Aroussi, M. A photovoltaic power prediction ap-proach enhanced by feature engineering and stacked machine learning model. *Energy Rep.* **2022**, *8*, 1288–1300. [CrossRef]
27. Massaoudi, M.; Abu-Rub, H.; Refaat, S.S.; Trabelsi, M.; Chihi, I.; Oueslati, F.S. Enhanced Deep Belief Network Based on Ensemble Learning and Tree-Structured of Parzen Estimators: An Optimal Photovoltaic Power Forecasting Method. *IEEE Access* **2021**, *9*, 150330–150344. [CrossRef]
28. Tan, Z.; Zhang, J.; He, Y.; Xiong, G.; Liu, Y. Short-Term Load Forecasting Based on Integration of SVR and Stacking. *IEEE Access* **2020**, *8*, 227719–227728. [CrossRef]
29. Zhang, H.; Zhu, T. Stacking Model for Photovoltaic-Power-Generation Prediction. *Sustainability* **2022**, *14*, 5669. [CrossRef]
30. Lateko, A.A.H.; Yang, H.-T.; Huang, C.-M.; Aprillia, H.; Hsu, C.-Y.; Zhong, J.-L.; Phương, N.H. Stacking Ensemble Method with the RNN Meta-Learner for Short-Term PV Power Forecasting. *Energies* **2021**, *14*, 4733. [CrossRef]
31. Michael, N.E.; Mishra, M.; Hasan, S.; Al-Durra, A. Short-Term Solar Power Predicting Model Based on Multi-Step CNN Stacked LSTM Technique. *Energies* **2022**, *15*, 2150. [CrossRef]
32. Satinet, C.; Fouss, F. A Supervised Machine Learning Classification Framework for Clothing Products' Sustainability. *Sustainability* **2022**, *14*, 1334. [CrossRef]
33. Adler, A.I.; Painsky, A. Feature Importance in Gradient Boosting Trees with Cross-Validation Feature Selection. *Entropy* **2022**, *24*, 687. [CrossRef] [PubMed]
34. Li, L.; Jamieson, K.; Rostamizadeh, A.; Gonina, E.; Ben-Tzur, J.; Hardt, M.; Recht, B.; Talwalkar, A. A system for massively parallel hyperparameter tuning. *Proc. Mach. Learn. Syst.* **2020**, *2*, 230–246.
35. Akiba, T.; Sano, S.; Yanase, T.; Ohta, T.; Koyama, M. Optuna: A next-generation hyperparameter optimization framework. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage, AK, USA, 4–8 August 2019; pp. 2623–2631.
36. Li, D.; Zhang, X.; Liu, D.; Wang, T. Aggregation of non-fullerene acceptors in organic solar cells. *J. Mater. Chem. A* **2020**, *8*, 15607–15619. [CrossRef]
37. Huang, W.; Zhang, J.; Ji, D. Extracting Chinese events with a joint label space model. *PLoS ONE* **2022**, *17*, e0272353. [CrossRef] [PubMed]
38. Chakraborty, D.; Mondal, J.; Barua, H.B.; Bhattacharjee, A. Computational solar energy—Ensemble learning methods for prediction of solar power generation based on meteorological parameters in Eastern India. *Renew. Energy Focus* **2023**, *44*, 277–294. [CrossRef]