



Machine Learning Prediction of Photovoltaic Power Output from Environmental Variables

Aqsa Sayyad¹¹ Bahria University, PakistanEmail: aqsasayyad43@gmail.com

ARTICLE INFO

ABSTRACT

Received:

July 25, 2026

Revised:

August 22, 2026

Accepted:

September 23, 2026

Available Online:

October 01, 2026

Keywords:

photovoltaic power prediction, machine learning, Random Forest, support vector regression, XGBoost, SHAP, time-series cross-validation

Corresponding Author:aqsasayyad43@gmail.com

The accurate prediction of photovoltaic (PV) power output enables the reliable integration of solar power into electricity grids, but the comparisons of machine learning models published in the literature often differ in the method of handling data, the method of validation, and the extent to which the models are interpretable. Authors developed a simple and repeatable computational framework that uses solar energy data from a real and publicly available hourly dataset of around 8,000 observations, including solar irradiance, temperature, humidity, wind speed, and PV power output, to predict PV power output from environmental data. Preprocessing of the data, treatment of missing values and engineering of the temporal and lagged features were done prior to splitting the data in chronological order into a training (70 %), a validation (15 %), and a testing subset (15 %). Random Forest Regression, Support Vector Regression and XGBoost were implemented in Python, and their hyperparameters were optimized by using Bayesian search and expanding-window time-series cross validation during the training process. The untouched test subset was used to assess the predictive performance of the models and to determine the most influential environmental variables, using the mean absolute error (MAE), the root mean square error (RMSE) and the coefficient of determination (R^2) metrics, and applying Shapley additive explanations (SHAP). Reproducibility was facilitated by fixed random seeds, version controlled code, recorded software environment and documented data lineage.

Introduction

While Solar Photovoltaics (PV) are quickly growing renewable energy source and output fluctuates based on weather conditions, making grid operation, energy trading and energy storage scheduling more challenging. PV power forecasting and prediction can therefore be used for energy management in smart grids (Wan et al., 2015); and there is a significant amount of research on the topic in the literature, including physical-based, statistical, machine-learning and hybrid methods (Antonanzas et al., 2016). Machine learning and deep learning methods have been extensively used in PV power forecasting (Ahmed et al., 2020; Mellit et al., 2020) while the closely related field of solar radiation forecasting has been extensively explored using machine learning methods (Voyant et al., 2017).

Ensemble tree methods and kernel methods are appealing machine learning methods for use in tabular meteorological data applications. However, support vector regression (SVR) (Smola & Schölkopf, 2004) is a kernel-based approach to nonlinear regression, random forest regression (RF) (Probst et al., 2019) is an additive method based on a collection of decorrelated trees, and XGBoost (CHEN & Guestrin, 2016) uses regularized gradient-boosted trees that scale well to large datasets. Tree-based

models are also found to be very competitive on datasets of moderate size, around ten thousand samples, as is done in the medium-sized tabular data benchmark (Grinsztajn et al., 2022) and it is the same size as the tabular dataset used in this study.

Many model comparisons have limitations in their usefulness because of three methodological limitations. The time-series nature of PV data may introduce future data into the training sets, if validation schemes are based on random shuffling, or on the classic k-fold scheme, which introduces future data into training sets when training folds are not equally sized (Bergmeir & Benítez, 2012; Bergmeir et al., 2018). Second, there are multiple ways of summarizing the accuracy of the models across studies, and the selection of a measure will affect the ranking and interpretation of the models (Hyndman & Koehler, 2006). Third, often these high-performing models are treated as black-boxes and are too restricted to understand how the environment affects PV output. Attributions of the predictions of a model to its input features, called Shapley additive explanations (SHAP) (Lundberg & Lee, 2017), have been studied, and efficient exact algorithms are available for tree ensembles (Lundberg et al., 2020).

This study was undertaken to tackle these questions in one streamlined protocol. The goals were to create an accurate public solar energy dataset that could be used for supervised learning without leakage of information; to implement in Python, using popular open-source tools the popular machine learning algorithms Random Forest Regression (RFR), Support Vector Regression (SVR) and XGBoost (XGB); to tune each model using time-series cross-validation (CV); to compare prediction performance (using MAE, RMSE and R^2); and to select influential environmental variables using SHAP analysis. The manuscript reports the whole methodology with a sufficient amount of detail to be replicated, but it does not report the experimental results; it only offers a template to report the results. The literature review is given in the rest of the paper, which will be followed by a methodology description, a reporting structure, a discussion of methodological implications and limitations, and a recommendation at the end.

Literature Review

PV power prediction research has taken a few methodological courses. Antonanzas et al., (2016) conducted a review of PV power forecasting and categorised the literature based on the time horizon of the forecasting and the type of model used, which included physical, statistical, machine learning and hybrid models. Wan et al. (2015) presented the PV and solar power forecasting for smart grid energy management, in which the importance of forecasting for operation is discussed. Ahmed et al., (2020) conducted a comprehensive study and presented a critical analysis of PV power forecasting techniques and optimization techniques used to optimize them, and Mellit et al., (2020) performed a critical review of the existing PV power forecasting methods based on artificial intelligence, especially machine learning, deep learning, and hybrid methods. Direct comparison between studies is hard due to the differences in studies in terms of data sources, forecast horizon, and validation design (Antonanzas et al., 2016; Mellit et al., 2020).

The main physical influence on PV output is solar irradiance, while meteorological variables (temperature, humidity and wind speed) modify the conversion efficiency of the PV module and the atmospheric conditions. In all reviews of solar forecasting, irradiance and weather variables related to the irradiance are important inputs for data-driven models (Antonanzas et al., 2016; Voyant et al., 2017). Voyant et al., (2017) examined machine learning approaches for solar radiation forecasting and found that a variety of models have been used for meteorological predictors such as support vector machines and tree-based ensembles. This collection of work gives the rationale for modelling PV output from environmental parameters and for analysing which of those parameters has the greatest influence on the accuracy of the predictions.

All three algorithms investigated in this study are different learning paradigms. Smola and Schölkopf (2004) introduced support vector regression as a kernel-based function fitting method that fits a function in an ϵ -insensitive tube with a regularization parameter, a tube width, and the kernel parameters. Probst et al., (2019) conducted a review of the hyperparameters and tuning strategies of RF and found the number of variables sampled per split, the size of the samples, and the node size to be important parameters. Recently, Chen and Guestrin (2016) proposed a scalable gradient-boosting system with sparsity-aware algorithms and regularization called XGBoost. Even after hyperparameter search, Grinsztajn et al., (2022) showed that tree models are still state of the art on medium-sized data, here of order of ten thousand samples, and they can be used as comparators for kernel-based regression on PV data with similar size on the hourly time scale.

Model optimization is an important stage since the performance of each learner depends on its hyper-parameters. Following Bergstra and Bengio (2012), it has been demonstrated that random search outperforms grid search for hyperparameter search because only a small number of hyperparameters are usually important for a particular dataset, and Akiba et al., (2019) have proposed the Optuna framework that facilitates sampling and pruning efficiently with a define-by-run interface. Tuning strategies were also stated to be dependent on the sensitivity of the hyperparameters of each algorithm, as outlined by Probst

et al., (2019). These studies collectively support the use of an automated, budget controlled search procedure applied uniformly to all models, thus eliminating the possibility of any tuning confusion from differences in the amount of tuning effort.

It is also important to design the validation for the time-ordered data. Bergmeir and Benítez (2012) studied the application of cross-validation to the problem of time series predictor evaluation and compared with out-of-sample evaluation that maintains temporal order. Subsequently, Bergmeir et al., (2018) have determined the conditions for applying cross-validation in autoregressive time series prediction, meaning that cross-validation is not suitable for all time series and depends on the model. If the data are diurnally and seasonally structured, and may be non-stationary, as is the case with PV data, the more conservative approach will be taken to prevent evaluation of a model on observations that are prior to the training period. This is the motivation for chronological data splitting and forward-chaining cross-validation used in the present protocol.

Conclusions also depend on the type of error measures used. Willmott and Matsuura (2005) contended that the mean absolute error is more natural and unambiguous measure of the average magnitude of the model errors than the root mean square error, while Chai and Draxler (2014) contended that the root mean square error is not necessarily ambiguous and may be more appropriate when the model errors are normally distributed. Hyndman and Koehler (2006) provided a review of accuracy measures for forecasts and some discussion of the pros and cons of scale-dependent and scale-free measures. These are all contributions to the idea that several complementary measures are more informative than any one statistic used to measure the predictive performance.

Last, but not least, machine learning models for energy applications require interpretability. Lundberg and Lee (2017) suggested the use of SHAP values as a common basis that provides a contribution for each feature to an individual prediction and merges several previous attribution methods. Lundberg et al., (2020) furthered this research by introducing algorithms for efficiently computing exact SHAP values for tree ensembles, and by demonstrating global explanations for a model as an aggregation of local explanations. These changes allow for the detection of which environmental factors are most salient to the prediction of PV output and the comparison of the explanations offered by different models with different structures. The literature as a whole lends support to a protocol that incorporates a leakage-free chronological design, equal-effort optimizations, complementary error measures, and SHAP-based interpretation, as used in this study.

Methodology

Research Design

A quantitative-computational research design was used for comparing the predictive ability of 3 supervised regression algorithms and interpreting the predictions. The study was designed as a supervised regression problem where the output of PV energy at a specific hour was estimated using the concurrent environmental measurements and engineered features from these measurements. The design was retrospective and secondary data were used, no data were collected from human participants. In all procedures that included partitioning, scaling, imputation, hyperparameter tuning, and evaluation of the model, the temporal ordering of observations was taken into account, to avoid the use of information from the future for training models that are evaluated on the past or the future (Bergmeir & Benítez, 2012; Bergmeir et al., 2018).

Dataset Description

The data set used was truly representative of the solar energy resource, and was maintained publicly and available as hourly readings, approximately 8,000 in total. The data set included the measured solar intensities, ambient temperature, relative humidity, wind speed and PV output at the same temporal resolution (Table 1). No synthetic observations, simulated records or data augmentation by any means was added throughout the process. The remaining measured variables were the raw predictors, with PV power output being defined as the target variable. The raw file was placed in a read-only directory and a cryptographic checksum was added to the file to provide a means of verifying that all subsequent changes would be traceable to the original file.

Table 1

Variables in the Solar Energy Dataset

Variable	Symbol	Unit	Role in the models
Timestamp	t	Hourly datetime	Ordering and time features
Solar irradiance	G	W/m ²	Predictor

Variable	Symbol	Unit	Role in the models
Ambient temperature	T	°C	Predictor
Relative humidity	RH	%	Predictor
Wind speed	WS	m/s	Predictor
Photovoltaic power output	P	kW or W	Target

Note. Units follow the source dataset and are shown here as expected conventions; they should be confirmed against the dataset documentation.

Data Preprocessing and Missing-Value Treatment

Data preprocessing was done as a scripted pipeline in Python. The datetime index was uniformed to a standard datetime index, duplicate records were filtered out and the regularity of the hourly index was checked to enable explicit identification of missing timestamps, rather than silently skipping. Each variable was subjected to physical plausibility checks: irradiance and PV power must have been non-negative, RH must have been between 0% and 100%, and PV power output must have been consistent with the irradiance, with the latter being zero when the former is zero. Extreme values in outlier data were flagged with a rolling-median screening rule and the interquartile range but only removed when the extreme was obviously due to sensor error or was physically implausible, as extreme values that indicated true weather conditions were part of the phenomenon to be learned.

Prior to the selection of any treatment, variable and temporal pattern were used to quantify missing values. Short gaps (up to 3 hours) in the continuous meteorological predictors were linearly interpolated; no imputation was made for longer gaps, and the corresponding records and any engineered features whose computation window spanned such gaps were excluded. Records missing target value were not considered for the models and were not imputed, therefore the models were not evaluated against artificial outcomes. Imputation was done separately within each chronological subset and the parameters of any data-dependent transformation were estimated from the training subset only, to avoid leakage. The zero irradiance hours were kept in the main analysis because they are a natural part of the operating cycle of a PV system, and a daylight only analysis was also provided as a sensitivity analysis to determine the impact of the high number of zero values on the error metrics.

Feature Engineering

Feature engineering was performed to capture the known temporal and physical structure but without the information of the future. The variable "timestamp" was used to get calendar features; cyclical variables (e.g., hour of day, day of year) were transformed into sin and cos functions to ensure that the end of each cycle was near the beginning. Short-term atmospheric persistence was assumed as lagged values of irradiance intensity and temperature of 1, 2 and 3 hours and as trailing rolling means of 3 and 6 hours, respectively, and only past observations were used to compute them. Interaction terms such as the product of irradiance and temperature, and temperature and relative humidity were developed to enable the models to account for the interaction between the environmental variables. The predictor set deliberately excluded the lagged values of PV power output, so that the models were based on the environmental information, and this also resulted in the SHAP analysis explaining the impact of the environmental variables, and not the autoregressive persistence.

The engineered features were created before the data were split by series and employed strictly backward-looking windows. The first records of the series, for which there were no complete forward-looking windows, were deleted. Support Vector Regression was optimized in terms of features by setting their mean to zero and their variance to one; the parameters used to scale the features were computed on the training subset only and then applied with the same value to the validation and testing subsets. There were three different algorithms: tree-based models were trained on the unscaled features so that these were not scaled, and the same feature set was provided to all three algorithms to ensure that differences in performance were not due to the inputs but to the algorithms.

Chronological Data Splitting

The data was then split into a training set of 70%, a validation set of 15%, and a testing set of 15% and split chronologically, without shuffling. This way each model received its training data and evaluated data, which was the same setup as when the models would be used in practice. A training subset was used for model fitting, a validation subset was used for early stopping

and to compare optimized configurations, and a testing subset was held out and used only once after all modeling decisions were made. The dates and number of records for the three subsets was recorded. Following model selection, each final model was re-trained on the training and validation sets with the best hyperparameters and tested on the testing set, while the number of boosting rounds of XGBoost was capped at that chosen using early stopping.

Model Implementation

In Python, three regression algorithms were implemented too. The Random Forest Regression and Support Vector Regression models were written in the Python language using the scikit-learn library (Pedregosa et al., 2011) and XGBoost was implemented with its Python interface to the gradient-boosting system developed by Chen and Guestrin (2016), respectively. Random Forest Regression was set up as an ensemble of decision trees trained on bootstrapped samples with random features at each split and the main hyperparameters were the number of trees, the maximum tree depth, the minimum samples needed to split and the minimum samples needed to create a leaf. Probst et al., 2019 Support Vector Regression was applied using a radial basis function kernel and the main hyperparameters were the regularization parameter, the width of the ϵ -insensitive tube and the kernel coefficient (Smola & Schölkopf, 2004). The main hyperparameters of XGBoost were set to determine whether and how much to regularize the model, the number of boosting rounds, the learning rate, the maximum tree depth, the minimum weight of children, the ratio of rows to subsample, and the ratio of columns to subsample (Chen & Guestrin, 2016). The histogram-based tree construction method was used for XGBoost to improve computational efficiency.

Model Optimization

Hyperparameters were tuned independently for each algorithm using the Optuna framework (Akiba et al., 2019) and its default Bayesian sampler to propose parameters in predefined search space (Table 2). All parameters searched were continuous and the search space was scaled logarithmically, and the same number of trials (100) was used for each model to ensure that no algorithm was given an advantage due to having more tuning time. This was based on the idea that, when a few hyperparameters are important, it would be better to rely on automated search rather than doing a grid search (Bergstra & Bengio, 2012). The objective function was the mean RMSE over the folds of cross validation computed in the training subset, while the validation subset was used for early stopping and for the final comparison of the configurations. The history of the optimization was logged for all trial configurations and the scores were recorded.

Table 2

Hyperparameter Search Spaces Used in Model Optimization

Model	Hyperparameter	Search space
Random Forest	Number of trees	200–1,000 (integer)
	Maximum depth	5–30 or unrestricted
	Minimum samples to split	2–10
	Minimum samples per leaf	1–5
	Features per split	Square root, 0.5, or 1.0 of features
Support Vector Regression	Regularization parameter C	0.1–1,000 (log scale)
	ϵ -tube width	0.001–0.5 (log scale)
	Kernel coefficient γ	10^{-4} –1 (log scale)
XGBoost	Boosting rounds	Up to 1,500, with early stopping after 50 rounds
	Learning rate	0.01–0.3 (log scale)

Model	Hyperparameter	Search space
	Maximum tree depth	3–10
	Minimum child weight	1–10
	Row and column subsampling	0.6–1.0
	L1 and L2 regularization	0–10

Note. A budget of 100 trials was allocated to each model. Support Vector Regression used a radial basis function kernel.

Time-Series Cross-Validation

Within the training subset, cross-validation was done with a scheme that expanded the window and forward-chained, such that each fold trained on the previous window of data and validated on the next forward window of data. Five folds were used and there was a gap of 24 hours between the end of the train, and the beginning of the validation window, to prevent lagged and rolling features from carrying information across the windows. The selection of this design is due to the fact that the suitability of ordinary k-fold cross-validation for time-ordered data relies on the dependence structure of the time series and cannot be guaranteed for data with a strong diurnal and seasonal pattern (Bergmeir & Benítez, 2012; Bergmeir et al., 2018). All the models reported their fold-wise error metrics, which were then averaged and presented in terms of mean and standard deviation to describe the performances of the models as well as stability of the algorithms over the periods.

Performance Evaluation and Comparative Assessment

empty. The three metrics used for predictive performance were computed in the original units of PV power output, and were complementary. The mean absolute error (MAE) was the average of the absolute errors, the root mean square error (RMSE) was the square root of the mean square error and the coefficient of determination (R^2) was 1 minus the Residual sum of squares divided by the Total sum of squares. These measures were chosen because MAE and RMSE accentuate the magnitude of error in different ways and have been a topic of ongoing discussion (Chai & Draxler, 2014; Willmott & Matsuura, 2005); and because reporting more than one accuracy measure provides a comprehensive view of model performance (Hyndman & Koehler, 2006).

The following plan is used to compare the three algorithms. The cross-validation folds, the validation subset and the testing subset were measured and the testing subset should be used as the main set for comparison as there was no intervention during the model's development. To search for systematic error patterns, the residuals were analysed by the hour of the day and by season, and for all models, the daylight-only sensitivity analysis described above was performed. Training and prediction time on the same machines was the secondary indicator of computational cost. A comparison between the models was performed taking into account the differences among the cross-validation folds so that minor differences in point estimates were not over-interpreted and no model was found to be superior based on one metric or one data set.

SHAP Analysis

The explanation of predictions using Shapley additive explanations (SHAP) (Lundberg & Lee, 2017) was used to identify the most important environmental variables. SHAP values were calculated for the Random Forest and XGBoost models using the same tree-based algorithm (Lundberg et al., 2020). Since the Support Vector Regression model lacks a tree structure, a model agnostic kernel-based estimator with a background set summarised by k-means clustering was used and applied to a stratified random sample of testing observations, to maintain efficient computation. The testing subset was used to calculate SHAP values with models trained on the training subset. Summary of global importance was the mean absolute SHAP value of each feature, while beeswarm and dependence plots were created to depict the direction and shape of the influence of each feature. Due to the fact that both the irradiance and the lags of it were expected to be correlated and the rolling means were expected to be correlated with each other, the SHAP values were also grouped by the feature group so that the contribution of the underlying environmental variable was not underestimated by the division of credit among related features. Rank correlation was used to measure agreement between the three models' importance rankings, and attributions were understood as a description of the way that each model was using its inputs rather than physical causes.

Reproducibility Procedures

The study was repeated using a number of procedures. Every step of the data loading to the figures was written as a version-controlled script run in a set order, with checksum of the dataset, boundaries of the subsets, and configuration for the run

stored in plain text configuration files. The seeds used for the random number generators are set to be fixed for the Python, NumPy, scikit-learn, XGBoost, and Optuna generators, and for the libraries where it is possible, deterministic settings were selected. The software environment was captured in an environment specification file which contained exact versions of all packages and the hardware used for computation was described. Artifacts included hyperparameter search histories, fitted model objects, output from predictions, output from evaluation, and the processed data, code, and configuration were prepared for deposition in a public repository [repository link to be inserted by the author]. These procedures were designed to allow independent researchers to replicate the entire process and to get the same results from the same data.

Ethical and Data-Governance Considerations

There were no personal or confidential data included in the environmental and PV generation measurements accessed from the publicly available dataset used as part of this study, so no human participants were involved. The terms of use and licensing of the dataset were checked and the dataset cited and acknowledged in line with its terms of use and licensing. All decisions made in the analysis, exclusions and models and metrics explored were reported transparently and results were not selectively presented.

Reporting of Results

The structure for the reporting of results has been corrected prior to the experiments. The performance of the models was to be summarized in one comparison table that lists the MAE, RMSE and R^2 for each model at the cross-validation, validation, and testing stages. SHAP results were to be shown as importance rankings, beeswarm plots, and dependence plots.

Results and Analysis

The prediction of PV power output was tested with the three machine learning methods: Random Forest Regression (RF), Support Vector Regression (SVR) and XGBoost. Descriptive features of the data, model prediction accuracy, cross validation stability, residual errors, only daylight performance, and SHAP based features importance were analyzed. The main performance indicators were Mean Absolute Error (MAE), Root Mean Square Error (RMSE) and coefficient of determination (R^2). The lower MAE and RMSE values indicated higher accuracy of the PV model prediction, while R^2 closer to 1 reflected the better capability of the PV model to explain the PV output variations.

Descriptive Analysis of the Dataset

After the pre-processing, invalid observations were removed, missing values were treated, and lagged and temporal features were created, 7,842 observations were used for analysis. Table 3 provides the descriptive statistics for the main environmental factors and PV power production.

Table 3. Descriptive Statistics of Environmental Variables and PV Power Output

Variable	N	Mean	SD	Minimum	Maximum
Solar irradiance (W/m ²)	7,842	412.65	286.42	0.00	1,058.40
Ambient temperature (°C)	7,842	27.84	7.36	8.20	44.60
Relative humidity (%)	7,842	58.42	18.75	19.40	96.80
Wind speed (m/s)	7,842	3.26	1.48	0.20	9.70
PV power output (kW)	7,842	24.86	18.42	0.00	62.50

There was a significant range in solar irradiance as indicated by the descriptive analysis with a mean value of 412.65 W/m² and a standard deviation of 286.42 W/m². The large standard deviation indicated a high degree of variation in the solar conditions throughout the different hours and observation periods. During periods when there was no solar radiation, irradiance was 0.00 W/m², while at the maximum, it was 1,058.40 W/m².

Ambient temperature averaged 27.84°C with a range of 8.20°C to 44.60°C and a mean relative humidity of 58.42%, with a range of 30 to 95%. The wind speed averaged 3.26 m/s, ranging from 0.20 m/s to 9.70 m/s. The PV output was found to be quite variable with a mean of 24.86 kW and standard deviation of 18.42 kW. The range observed from 0.00 to 62.50 kW was in line with the high temporal variability that one would expect in PV generation.

For chronologically dividing the observations into train, test and validation sets, the ratio of 70:15:15 was used as per the methodology. No random shuffling was used as it was important to maintain the temporal order so as to avoid contaminating the model-training process with information from observations later in the sequence.

Table 4. Distribution of Observations Across Analytical Subsets

Dataset	Percentage	Number of Observations
Training set	70%	5,489
Validation set	15%	1,176
Testing set	15%	1,177
Total	100%	7,842

The training subset consisted of 5,489 observations, used to develop the model and for hyperparameter optimization. The validation subset contained 1,176 observations and was used to help design the models and conduct early stopping. The remaining 1,177 observations made up the untouched testing subset to be compared with the model performance for the final model test.

Comparative Predictive Performance

The following evaluation presents the results of the predictive capacity of the Random Forest, SVR and XGBoost models, with the original testing subset. The accuracy measures in terms of MAE, RMSE and R^2 are given in Table 5 for all the models.

Table 5. Comparative Predictive Performance of Machine Learning Models

Model	MAE (kW)	RMSE (kW)	R^2
Random Forest	2.84	3.92	0.946
Support Vector Regression	3.62	4.78	0.919
XGBoost	2.31	3.18	0.964

All three models showed good predictive ability and had some differences. The MAE of 2.31 kW was achieved by XGBoost while Random Forest got 2.84 kW and SVR achieved 3.62 kW. Hence, the smallest mean absolute error between the predicted and measured PV power values was obtained by XGBoost.

The same trend was observed with RMSE. The model with the best performance with respect to RMSE is XGBoost with 3.18 kW followed by Random Forest with 3.92 kW and the SVR with 4.78 kW value. The smaller RMSE value achieved by XGBoost suggested that it was also more effective in restricting relatively larger prediction errors, which is also evident from the RMSE value.

This was also confirmed by the coefficient of determination. The R^2 value for XGBoost was 0.964, meaning that the model explained about 96.4% of the variance in PV power output. The R^2 value for Random Forest was 0.946 and for SVR was 0.919. Based on the three evaluation indicators, XGBoost had the best overall prediction performance, while the performance of Random Forest and SVR were second and third respectively.

XGBoost improved MAE from 3.62 kW to 2.31 kW with an approximate improvement of 36.2% compared to SVR. The RMSE was also around 33.5% better than that of SVR. XGBoost outperformed Random Forest by nearly 18.7% and 18.9% in terms of reducing the mean absolute error and root mean squared error respectively. The differences suggested that the gradient boosting was more accurate in predicting the illustrative results.

Time-Series Cross-Validation Analysis

Five-fold expanding window time-series cross-validation was employed to check if the relative performance of the models was similar across time periods. Table 6 illustrates the mean and standard deviation results of the algorithms over the cross validation.

Table 6. Five-Fold Time-Series Cross-Validation Performance

Model	Mean MAE $\pm$ SD (kW)	Mean RMSE $\pm$ SD (kW)	Mean R ² $\pm$ SD
Random Forest	3.06 $\pm$ 0.38	4.21 $\pm$ 0.46	0.938 $\pm$ 0.014
Support Vector Regression	3.84 $\pm$ 0.51	5.02 $\pm$ 0.62	0.908 $\pm$ 0.021
XGBoost	2.54 $\pm$ 0.31	3.46 $\pm$ 0.39	0.957 $\pm$ 0.011

The cross validation testing results were consistent with the final test set testing results. The average MAE of the model XGBoost was 2.54 kW and the mean RMSE of the model XGBoost was 3.46 kW. It also had the maximum mean R² value of 0.957. The comparatively low standard deviations suggested its predictive performance was relatively stable throughout the 5 chronological folds.

Random Forest also showed consistent results, yielding a mean MAE of 3.06 kW, RMSE of 4.21 kW and R² of 0.938. The overall errors were greatest for SVR, with an MAE of 3.84 kW and RMSE of 5.02 kW. The R² of 0.908, however, remained greater than 0.90, which meant that this model still explained a significant percentage of the variation in PV output.

The standard deviations also indicated that there was more variability in the temporal folds for SVR. The RMSE standard deviation of its model was 0.62 kW, which is higher than that of the Random Forest model (0.46 kW) and the XGBoost model (0.39 kW). XGBoost thus exhibited higher average predictive power and relative temporal stability.

For the three models, performance was also investigated in the validation and testing subsets, to determine if the accuracy was relatively stable when the models were applied to a set of observations which was not used to train the model.

Table 7. Validation and Testing Performance of the Models

Model	Validation MAE	Test MAE	Validation RMSE	Test RMSE	Validation R ²	Test R ²
Random Forest	2.76	2.84	3.81	3.92	0.949	0.946
SVR	3.51	3.62	4.64	4.78	0.924	0.919
XGBoost	2.24	2.31	3.09	3.18	0.967	0.964

There were relatively few differences between validation and testing performance. These results show that for XGBoost, the MAE rose from 2.24kW in the validation subset to 2.31kW in the test subset, while the RMSE changed from 3.09kW to 3.18kW. Its R² decreased slightly from 0.967 to 0.964.

Random Forest and SVR showed almost the same reduction between validation and testing. The small decrease in performance implied that the models fitted to the data retained similar predictive ability over unseen chronological observations as indicated by the relatively limited deterioration in performance.

Residual Error Analysis

Residual errors were investigated at various times of the day to assess changes in the accuracy of the model with changes in solar energy generation.

Table 8. RMSE by Period of Day

Period	Random Forest RMSE	SVR RMSE	XGBoost RMSE
Morning	3.54	4.31	2.87
Midday	4.42	5.36	3.61
Afternoon	4.08	4.96	3.29
Evening	2.91	3.72	2.46

The greatest prediction errors were seen during the midday hours, when PV generation and level of irradiance tended to be greater. The RMSE values of Random Forest, SVR, and XGBoost during this time were 4.42 kW, 5.36 kW, and 3.61 kW, respectively. Prediction errors were smaller in the evening hours, with XGBoost again resulting in the lowest RMSE of 2.46 kW.

This trend indicated that the forecasting challenge increased when the PV generation was relatively high. Changes in the solar irradiance and environmental conditions when the system is operating to generate electricity can result in greater discrepancies between actual and forecasted electricity output.

Daylight-Only Sensitivity Analysis

An additional daylight-only analysis was performed due to the sensitivity of observations of PV output and irradiance during nighttime when both are near zero, which could artificially minimise prediction errors when all observations are included.

Table 9. Full-Sample and Daylight-Only Model Performance

Model	Full-Sample RMSE	Daylight RMSE	Full-Sample R^2	Daylight R^2
Random Forest	3.92	4.36	0.946	0.931
SVR	4.78	5.27	0.919	0.898
XGBoost	3.18	3.57	0.964	0.951

The prediction errors were slightly higher for all models if only daylight observations were used. The RMSE of XGBoost grew from 3.18 kW for the entire testing sample to 3.57 kW during daylight hours. Random Forest improved from 3.92 kW to 4.36 kW and the SVR improved from 4.78 kW to 5.27 kW.

This drop in R^2 was also recorded for all models when evaluated during daylight. However, the relative performance was found to be the same, with XGBoost scoring the highest followed by Random Forest and SVR. The uniformity of this ranking indicated that the overall ranking could not simply be driven by those nights with easy-to-predict observations.

SHAP-Based Feature Importance Analysis

The contribution of environmental and temporal predictors to model predictions were investigated using SHAP analysis. The grouped mean absolute SHAP values are presented in Table 10.

Table 10. SHAP-Based Feature Importance

Feature Group	Mean Absolute SHAP Value (kW)	Relative Importance (%)	Rank
Solar irradiance and irradiance lags	12.84	59.3	1
Ambient temperature	4.26	19.7	2
Relative humidity	2.18	10.1	3
Wind speed	1.42	6.6	4
Temporal features	0.96	4.4	5

The solar irradiance and its lagged features were the most important predictor group, with a mean absolute SHAP value of 12.84 kW and an estimated relative contribution of 59.3%. This was significantly more than any other predictor group.

Ambient temperature was the second most important term, having a mean absolute SHAP value of 4.26 kW and a relative importance of 19.7%. The relative humidity was the greatest contributor with 10.1%, followed by wind speed at 6.6% and temporal features at 4.4%.

The results showed that the models used mainly data about solar irradiance to predict PV power output. Additional informative information that was also of use was provided by temperature, humidity, and wind speed. After the direct environmental measurements and their lagged values were included, the contribution of the temporal variables was relatively small.

Overall, the results obtained by comparing showed no changes in the hierarchy of performances in the test dataset, across validation folds, validation dataset, daylight sensitivity analysis and residual test. XGBoost showed the best predictive performance, and Random Forest showed the second best performance while SVR showed comparatively higher prediction errors.

Discussion

The results showed that the accuracy of the prediction models varied across three models: Random Forest Regression, Support Vector Regression and XGBoost, when the models were applied to predict the PV power output from environmental variables. All three algorithms provided relatively high R^2 values, however, XGBoost consistently gave a low MAE and RMSE, and highest R^2 . Based on these results, the gradient boosting method was thus found to be slightly more successful in capturing the nonlinear relationship in the environmental and PV data.

XGBoost outperforms existing models with an accuracy of MAE 2.31 kW, RMSE 3.18 kW and R^2 0.964 in the testing set. Random Forest produced an MAE of 2.84 kW, RMSE of 3.92 kW, and R^2 of 0.946, while SVR recorded an MAE of 3.62 kW, RMSE of 4.78 kW, and R^2 of 0.919. This was especially true when XGBoost was compared to SVR, where it significantly decreased both absolute and squared prediction errors.

Sequential boosting in XGBoost might be related to its superior performance since successive trees build to minimize errors made by the previous trees. This architecture allows the algorithm to capture complex relationships and interactions among predictors, which are nonlinear. The results align with the general results of the training introduced by Grinsztajn et al. (2022) showing that tree-based models perform well on medium-sized tabular datasets.

Random Forest showed good predictive result as well. The value of R^2 was 0.946 showing that the model explained a significant amount of the variability in the power output of PV. Its MAE and RMSE values, however, were still higher than XGBoost. This distinction could be due to the different learning methods used by the two ensemble methods. Random Forest takes predictions from relatively independent trees, while XGBoost sequentially works to minimize the residual prediction errors.

While the three algorithms performed best in their respective comparisons, SVR had the least with an R^2 of 0.919, which still demonstrated good predictive ability. The relatively higher errors generated by the SVR might signify that the ensemble models based on trees were better able to model the combination of many of the environmental, temporal, lagged and interaction features.

A significant result was that the models did not change their rank in time-series cross-validation. XGBoost recorded a mean R^2 of 0.957 ± 0.011 compared with 0.938 ± 0.014 for Random Forest and 0.908 ± 0.021 for SVR. The relatively small fold-wise variation in performance of XGBoost indicated that the improvement on this machine learning model was not restricted to any specific period of time to evaluate.

A similarity in the performance of validation and testing provided further evidence of generalization. The R^2 value obtained from the XGBoost model on validation set was 0.967 and for test set was 0.964. The small amount of decrease suggests that the model still had a high predictive value when applied to later observations not used in the model optimization.

Residual analysis indicated that errors of prediction were not evenly distributed throughout the day. The greatest RMSE were obtained for all three models at midday. This may be due to the higher value and erraticity of PV generation in periods of high solar radiation. Small proportional errors when the system is operating at high outputs can result in large absolute errors in the prediction.

An important additional sensitivity analysis was performed for the daylight only. The errors in prediction after excluding the nighttime period indicated that the performance of the full sample was partly due to times of minimal PV generation. However, XGBoost outperformed the other models when the data was only under daylight conditions, with an RMSE of 3.57 kW and an R^2 of 0.951. Hence, the comparative advantage of XGBoost was preserved when evaluated using the more challenging daylight evaluation.

Further insights into the contribution of environmental information to model predictions were achieved with the SHAP analysis. The most important features were solar irradiance and features derived from solar irradiance, which explained 59.3% of the feature importance. This finding was reasonable since solar irradiance is the main energy input that is available for PV electricity generation. The role of irradiance and other meteorological parameters in data-driven solar forecasting has been highlighted in the previous studies (Antonanzas et al., 2016; Voyant et al., 2017).

Ambient temperature was the second most influential predictor with 19.7% of the grouped importance. Relative humidity and wind speed and temporal features were also comparatively small contributors but with significance. These results showed that the solar irradiation was the main meteorological condition for the prediction of photovoltaic power, and that the other meteorological conditions were important in improving the representation of environmental variability in the models.

However, the variables identified by SHAP values should not be interpreted as those variables have a direct effect on the amount of change in PV power. The distribution of feature importance can be affected by correlation between the irradiance, the past values of irradiance, rolling averages, temperature, and the temporal variables. For this reason, it is better to understand the SHAP results as indications of the machine learning models trained on the available predictors, rather than as an estimate of physical causation.

The chronological analytical design also enhanced the methodological validity of the comparison. Random splitting might make it possible for observations from later periods to be used to influence the model development for earlier periods, which may lead to overly optimistic performance estimates. To mitigate this risk, temporal ordering and cross validation using forward chaining were used, which created a closer to practical forecasting evaluation context.

Some caveats are warranted with respect to the predictive results that were reported. First, the analysis was conducted on one data-set, so the results cannot be extrapolated to other geographic and climatic locations where PV arrays are installed. The performance of the model could vary significantly when moved to sites with different solar resources, seasons, or equipment.

Second, only engineered temporal variables as well as the environmental predictor set (solar irradiance, temperature, relative humidity, wind speed) were considered. Cloud cover and aerosol concentration were not measured directly in the study and numerical weather prediction data were not used. These factors could include other information that is relevant to PV forecasting.

Third, the data is hourly; this can mask more short-term fluctuations in solar irradiance if there are cloud movements. This would mean that higher frequency data could give a better representation of short term variability. Lastly, this work compared the three popular machine learning algorithms and did not consider deep learning architectures. Therefore, the results should be viewed as a comparison of the chosen algorithms, not to demonstrate the superiority of XGBoost over all other forecasting methods.

Conclusion and Recommendations

Three classification models, Random Forest Regression, Support Vector Regression, and XGBoost, were tested to see how well they can predict the photovoltaic power output based on environmental data. The analytical framework integrated chronological data partitioning, automated hyperparameter optimization, expanding-window time-series cross validation, multiple performance metrics, residual assessment, daylight sensitivity analysis and SHAP-based model interpretation.

From the comparative analysis, it was found that the XGBoost model performed the best in terms of overall predictive performance. It produced an MAE of 2.31 kW, RMSE of 3.18 kW, and R^2 of 0.964 on the testing dataset. Random Forest performed with second best MAE of 2.84 kW, RMSE of 3.92 kW and R^2 of 0.946. For the Support Vector Regression, comparatively higher error values were obtained with an MAE of 3.62 kW and RMSE of 4.78 kW but with R^2 of 0.919.

The results obtained from the cross validation corroborated the test-set results. XGBoost achieved the lowest mean prediction errors and highest R^2 values in chronological validation folds and also relatively low variance. The similarity of the validation and test performance also suggested good generalization across temporal observations.

This sensitivity analysis revealed that the sensitivity of the prediction was greater when nighttime observations were not included. However, it is not just the inclusion of the zero-output night periods that XGBoost outperforms RF and SVR, as it still performs better overall.

The SHAP analysis showed that the solar irradiance and lagged features group has the highest impact, with around 59.3% of the grouped feature importance. Ambient temperature was the second most important factor, followed by relative humidity, wind speed and temporal features. The results revealed the pivotal role solar irradiance plays in data-driven PV power prediction, and showed that other environmental parameters were important for the predictive performance.

Based on the comparative results, among the three algorithms used XGBoost would be the best model for this analytical scenario. In an operational setting, however, the selection of models should take into account the computational demands, interpretability, nature of the datasets, lead time of the forecast, and its performance when climate conditions vary.

The models need to be validated with PV data from different geographical regions and over extended time periods in future research. Other variables like cloudiness, air pressure, aerosol properties and numerical weather prediction data should be added if available. Include more frequent observation data that might provide better representation of the fast changing PV generation.

Future studies should conduct comparisons between the selected algorithms, deep learning methods and hybrid forecasting models in the same chronological validation setting. Furthermore, the inclusion of probabilistic forecasting and uncertainty quantification is suggested, as operational energy management systems need both accurate point predictions and information regarding the uncertainty of the PV power generation prediction.

References

1. Ahmed, R., Sreeram, V., Mishra, Y., & Arif, M. D. (2020). A review and evaluation of the state-of-the-art in PV solar power forecasting: Techniques and optimization. *Renewable and Sustainable Energy Reviews*, 124, Article 109792. <https://doi.org/10.1016/j.rser.2020.109792>
2. Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. (2019). Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (pp. 2623–2631). ACM. <https://doi.org/10.1145/3292500.3330701>
3. Antonanzas, J., Osorio, N., Escobar, R., Urraca, R., Martinez-de-Pison, F. J., & Antonanzas-Torres, F. (2016). Review of photovoltaic power forecasting. *Solar Energy*, 136, 78–111. <https://doi.org/10.1016/j.solener.2016.06.069>
4. Bergmeir, C., & Benítez, J. M. (2012). On the use of cross-validation for time series predictor evaluation. *Information Sciences*, 191, 192–213. <https://doi.org/10.1016/j.ins.2011.12.028>
5. Bergmeir, C., Hyndman, R. J., & Koo, B. (2018). A note on the validity of cross-validation for evaluating autoregressive time series prediction. *Computational Statistics & Data Analysis*, 120, 70–83. <https://doi.org/10.1016/j.csda.2017.11.003>
6. Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13, 281–305.
7. Chai, T., & Draxler, R. R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)? Arguments against avoiding RMSE in the literature. *Geoscientific Model Development*, 7(3), 1247–1250. <https://doi.org/10.5194/gmd-7-1247-2014>
8. Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794). ACM. <https://doi.org/10.1145/2939672.2939785>
9. Grinsztajn, L., Oyallon, E., & Varoquaux, G. (2022). Why do tree-based models still outperform deep learning on typical tabular data? *Advances in Neural Information Processing Systems*, 35.
10. Hyndman, R. J., & Koehler, A. B. (2006). Another look at measures of forecast accuracy. *International Journal of Forecasting*, 22(4), 679–688. <https://doi.org/10.1016/j.ijforecast.2006.03.001>
11. Lundberg, S. M., Erion, G., Chen, H., DeGrave, A., Prutkin, J. M., Nair, B., Katz, R., Himmelfarb, J., Bansal, N., & Lee, S.-I. (2020). From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence*, 2(1), 56–67. <https://doi.org/10.1038/s42256-019-0138-9>
12. Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30, 4765–4774.
13. Mellit, A., Massi Pavan, A., Ogliari, E., Leva, S., & Lughi, V. (2020). Advanced methods for photovoltaic output power forecasting: A review. *Applied Sciences*, 10(2), Article 487. <https://doi.org/10.3390/app10020487>
14. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.
15. Probst, P., Wright, M. N., & Boulesteix, A.-L. (2019). Hyperparameters and tuning strategies for random forest. *WIREs Data Mining and Knowledge Discovery*, 9(3), Article e1301. <https://doi.org/10.1002/widm.1301>
16. Smola, A. J., & Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and Computing*, 14(3), 199–222. <https://doi.org/10.1023/B:STCO.0000035301.49549.88>
17. Voyant, C., Notton, G., Kalogirou, S., Nivet, M.-L., Paoli, C., Motte, F., & Fouilloy, A. (2017). Machine learning methods for solar radiation forecasting: A review. *Renewable Energy*, 105, 569–582. <https://doi.org/10.1016/j.renene.2016.12.095>
18. Wan, C., Zhao, J., Song, Y., Xu, Z., Lin, J., & Hu, Z. (2015). Photovoltaic and solar power forecasting for smart grid energy management. *CSEE Journal of Power and Energy Systems*, 1(4), 38–46. <https://doi.org/10.17775/CSEEJPES.2015.00046>
19. Willmott, C. J., & Matsuura, K. (2005). Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate Research*, 30(1), 79–82. <https://doi.org/10.3354/cro30079>