Reliable solar irradiance prediction using ensemble learning-based models: A comparative study

Junho Lee[a], Wu Wang[a], Fouzi Harrou[a], Ying Sun[a]

[a]*King Abdullah University of Science and Technology (KAUST)*
Computer, Electrical and Mathematical Sciences and Engineering (CEMSE) Division, Thuwal 23955-6900, Saudi Arabia
E-mail: fouzi.harrou@kaust.edu.sa

Abstract

Accurately predicting solar irradiance is important in designing and efficiently managing photovoltaic systems. This paper aims to provide a reliable short-term prediction of solar irradiance based on various meteorological factors using ensemble learning-based models that take into account the time-dependent nature of the solar irradiance data. The use of ensemble learning models is motivated by their desirable characteristics in combining several weak regressors to achieve an improved prediction quality relative to conventional single learners. Furthermore, they reduce the overall prediction error and have the ability to combine different models. In this paper, we first investigate the prediction performance of the well-known ensemble methods, Boosted Trees, Bagged Trees, Random Forest, and Generalized Random Forest in short-term prediction of solar irradiance. The performance of these ensemble methods has been compared to two commonly known prediction methods namely Gaussian process regression, and Support Vector Regression. Typical Meteorological Year data are used to verify the prediction performance of the considered models. Results showed that ensemble methods offer superior prediction performance compared to the individual regressors. Furthermore, the results showed that the ensemble models have a consistent and reliable prediction when applied to data from different locations. Lastly, variables contribution assessment showed that the lagged solar irradiance variables contribute significantly to the ensemble models, which help in designing more parsimonious models.

Keywords: Solar irradiance, prediction, ensemble learning, TMY data.

1. Introduction

The main difficulty in solar energy production is the volatility intermittent of photovoltaic system power generation due mainly to weather conditions. Essentially, the variation of the temperature and irradiance can have a profound impact on the quality of electric power production [1]. As solar irradiance is highly related to solar power harvesting, thus its prediction can be a good indicator of power production [2]. For the large-scale solar farms, the power imbalance of the photovoltaic system may cause a significant loss in their economical profit. Thus, accurate prediction

of solar irradiance is becoming vital to reduce the impact of uncertainty and energy costs and enable suitable integration of photovoltaic systems in a smart grid. There have been many studies for models and algorithms to predict solar irradiance based on various meteorological factors that are routinely measured, such as temperature or humidity [3, 4].

Accurately predicting solar irradiance becomes the backbone of smart grids due to the increased photovoltaic systems installation. Much work has been done in the literature to develop improved solar power prediction techniques. Fu and Cheng proposed the solar irradiance prediction with the clearness index and several features extracted from the all-sky images [5]. They utilized the regression technique to predict the clearness index and then calculated the solar irradiance for 5–15 minutes ahead from the predicted clearness index. Cheng and Yu conducted a classification of cloud types based on all-sky images to improve solar irradiance prediction[6]. They showed that multiple regression models according to cloud types could yield more accurate prediction results on short-term irradiance prediction. Torregrossa et al. proposed an ultra-short-term dynamic interval predictor (DIP) of solar irradiance, which relies on experimentally observed correlations between forecasting errors in the next time step and derivative of the solar irradiance[7]. Bae et al. considered the support vector machine (SVM), combined with k-means clustering for weather type, for one-hour ahead forecasting of the solar irradiance from various meteorological drivers including the cloud cover [8]. They highlighted that the SVM regression model performed better than the artificial neural network (ANN) and the nonlinear autoregressive (NAR) schemes. Further, Kim et al. predicted the direct normal irradiance (DNI) from the global horizontal irradiance (GHI) at minute scale in the Korean Peninsula via Monte-Carlo simulation [9]. The method in [10] integrates the benefits of SVM model, particle swarm optimization, and discrete wavelet transformation to predict solar radiation based on satellite data as inputs. However, this approach cannot be implemented in real-time due to the use of a discrete wavelet transformation that needs the availability of batch data. In [11], kerne-based methods have been applied to predict daily global solar radiation in humid regions.

McCandless et al. proposed a decision tree-based model to predict both the temporal and spatial variability of solar irradiance [12]. They included weather data such as cloud cover and temperature as predictors. They achieved better prediction precision compared to a climatology model. Hirata and Aihara used a time series model called infinite-dimensional delay coordinates for predicting solar irradiance [13]. They demonstrated that the proposed method is effective especially for the period after sunrise. Frimane et al. proposed a Dirichlet process Gaussian mixture model to produce synthetic solar global horizontal irradiance (GHI) time-series data at up to 1-minute resolution from input data of resolution higher than 10 minutes [14]. David et al. focused on the short term solar forecasting for a single location [15]. They adapted time series models such as ARMA and GARCH to produce both point predictions and interval predictions. Kakimoto et al. presented a method for probabilistic forecasting of solar irradiance based on

the conditional density of the observed solar irradiance conditional on the irradiance predicted by numerical weather prediction [16].

In [17], an advanced machine learning-based model has been applied to predict solar irradiation from weather measurements. This model is based on long short-term memory (LSTM) networks, which has an extended capacity to describe time dependencies in time series data. It showed good prediction performance compared to the backpropagation neural networks. The prediction of solar irradiation is provided hourly. Recently in [18], a deep learning method based on conventional neural networks is applied to predict solar irradiance on thirty stations in Turkey. Several input variables are used by the deep learning model for predication including extraterrestrial radiation and climatic variables, sunshine duration, cloud cover, minimum temperature, and maximum temperature. Watanabe and Nohara proposed an approach to predict time series of the surface solar irradiance (SSI) based on one-granule cloud features computed from satellite observation [19]. However, the cloud features are required for implementing this approach and the absence of information related to clouds movement can affect the prediction quality of this approach. A neural network approach is employed in [20] to improve the prediction of near-future global solar irradiance based on sky image data. Specifically, color information in the images is used as input for prediction. However, the prediction performance of this approach could be affected by the quality of the input images. The method in [21] used a hybrid genetic algorithm and ANN model to predict daily solar irradiance in Northwest China. Similarly, in [3] the ANN model has been combined with a mind evolutionary algorithm to improve daily prediction of solar irradiance. In [22], simplified vector-based modeling is proposed for predicting solar irradiance in urban areas. This approach showed good performance compared to RADIANCE by achieving predictions with the average differences of 3%, 4%, and 6%, for the low-, medium-, and high-density areas, respectively. Verbois et al. introduced an integrated model merging the advantages of both the weather research and forecasting model and multivariate statistical learning for enhancing prediction of hourly solar irradiance [23]. Results showed the outperformance of this model compared to smart persistence, a climatological prediction, and Global Forecasting System.

The major aim of this study is to investigate the capability of ensemble learning methods in improving solar irradiance prediction based on other meteorological factors. In fact, ensemble learning models emerged as a way to improve the prediction capacity of weak models by using several learners [24, 25]. Essentially, a key reason for choosing ensemble techniques over other traditional techniques, such as SVM and GPR, is due to their capability to get reduced variance with low bias [26]. Boosting and bagging are two popular ensemble learning methods. The contributions of this study are as follows:

- First, we introduce a dynamic ensemble learning models to improve solar irradiance prediction. In fact, the use

of ensemble learning models is motivated by their desirable characteristics in combining several weak regressors to achieve an improved prediction quality relative to conventional single learners. Furthermore, they reduce the overall prediction error and have the ability to combine different models. However, data from environmental processes (e.g. solar irradiance) are frequently correlated in time due to process dynamics. Accounting for the dynamic nature of data can also reflect the efficiency of the designed prediction models. In this paper, this dynamic propriety of the solar irradiance data is considered by using lagged data when designing the prediction models to capture the time evolution of the data.

- Second, four ensemble learning models including Boosted Trees (BS), Bagged Trees (BG), Random Forest (RF), and Generalized Random Forest (GRF) are constructed for hourly global solar irradiance based on meteorological factors in six locations in the United States. Since ensemble learning methods can reduce the prediction errors, these methods are expected to yield an elegant and more flexible of predicting solar irradiance over the traditional single methods. For this aim, the performance of the ensemble learning models are compared to support vector machine regression (SVM) and Gaussian process regression (GPR). Overall, we provide a comparison of six models and we considered different parameters of each model, which lead in comparing 13 prediction models. Three evaluation metrics are used to compare prediction performance, R^2 (called coefficient of determination), Root mean square prediction error (RMSPE), and Mean absolute prediction error (MAPE). Performance of ensemble learning models was particularly remarkable compared to the single models.

- Lastly, to show the importance of lagged data in solar irradiance prediction, we assessed variable importance or importance for the predictive model using RF, GRF, and BS. As expected, the lagged solar irradiance data has the largest contribution in the three models compared to the other variables. This clearly confirms the importance of incorporating lagged data in the ensemble learning models.

The rest of the paper is arranged as follows. Section 2 will briefly review the used ensemble learning methods namely BS, BG, RF, and GRF. Section 3 presents the used TMY datasets, and the results of the comparative study are illustrated in Section 4. Lastly, Section 5 concludes the paper and provides some future lines of research.

2. Ensemble learning methods

Ensemble learning is a very efficient tool that can boost the efficiency of a weak model for achieving accurate predictions. Ensemble methods based on boosting and bagging (i.e., BS, BG, RF, and GRF) are briefed in this section.

2.1. Bagged regression trees

Historically, bootstrap aggregating (bagging) methods have been originally introduced by Breiman [26] as a concatenation of several similar independent learners and computing the final prediction by averaging the outputs of all learners. Bagging methods are highly useful in practice because they have the capability to reduce the variance error of prediction [27]. Bagging tries method uses multiple trees (learners) to improve the prediction performance of the model (Figure 1). A key reason for the popularity of the BGs methods because they possess such a great capacity and flexibility to reduce the variance of regression trees and overcome the overfitting issue in the single tree. Figure 1 presents the conceptual schema of BGs predictive model. The BG method begins by randomly generating N new datasets with the same size n (bootstrap samples) by the replacement from input training data. Then, each generated dataset is used for training one tree in the ensemble. For example, in this study, the BG model contains 30 trees. At last, the BG prediction is computed by averaging all predictions as follows:

$$\widehat{\mathbf{y}} = \frac{1}{N}\sum_{i=1}^{N} f_i(\mathbf{X}), \tag{1}$$

where each tree model f_i is trained on bootstrap data i.

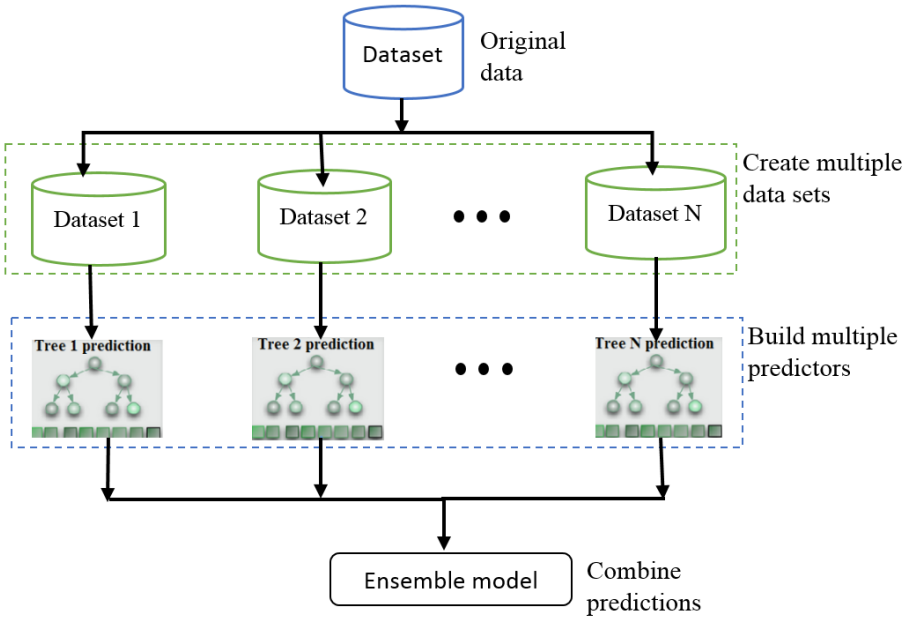

Figure 1: Conceptual presentation of BS method.

Theoretically, it is proved that the variance of prediction could be decreased by $1/N$ in comparison to the variance of a standalone learner. Therefore, the use of a large number of regressors will help improve the prediction accuracy of solar irradiance. To show the basic concept of using the bagging to improve prediction, we consider a prediction

problem with base regressors $r_1(x), \ldots, r_n(x)$. Assume that an ideal target function of true responses $y(x)$ resulting from a given set of inputs and that the distribution $p(x)$ is defined. Then, the error of every regressor is calculated as

$$e_i(x) = r_i(x) - y(x), \quad i = 1, \ldots, n \tag{2}$$

The mean squared error can be expressed as:

$$\mathbb{E}_x \left[(r_i(x) - y(x))^2 \right] = \mathbb{E}_x \left[e_i^2(x) \right]. \tag{3}$$

Thus, the mean overall error of all regressors is given by

$$\mathbb{E}_1 = \frac{1}{n} \mathbb{E}_x \left[e_i^2(x) \right] \tag{4}$$

Let us assume that that the errors are unbiased and uncorrelated, i.e. $\mathbb{E}_x [e_i(x)] = 0$ and $\mathbb{E}_x [e_i(x) e_j(x)] = 0, \quad i \neq j$. Essentially, the regression function is obtained by computing the average of the individual functions as

$$a(x) = \frac{1}{n} \sum_{i=1}^{n} r_i(x). \tag{5}$$

Then, its MSE is expressed as,

$$\begin{aligned} \mathbb{E}_n &= \mathbb{E}_x \left[\tfrac{1}{n} \textstyle\sum_{i=1}^{n} r_i(x) - y(x) \right]^2, \\ &= \mathbb{E}_x \left[\tfrac{1}{n} \textstyle\sum_{i=1}^{n} e_i \right]^2 = \tfrac{1}{n^2} \mathbb{E}_x \left[\textstyle\sum_{i=1}^{n} e_i^2(x) + \sum_{i \neq j} e_i(x) e_j(x) \right], \\ &= \tfrac{1}{n} \mathbb{E}_1. \end{aligned} \tag{6}$$

As mentioned above, it is necessary to employ bagging models other than only single regressors to learn relevant features of data and improve prediction accuracy. In fact, the usage of the averaged model permits reducing the MSE by a factor of n. The bagged models have the advantage of decreasing the prediction error by training the model based on different datasets. As numerous trees are combined in bagged trees, this makes the interpretation of the model difficult. In summary, bagged trees model is effective in enhancing the prediction performance, but lacks interpretability capabilities.

2.2. Boosted Trees

Boosting algorithm is an ensemble algorithm which combines multiple predictions from base learners [28, 29]. Originally, Boosting is proposed to boost the performance of a weaker classifier, which is only slightly better than random guessing. Gradually, it attracts attention from both the machine learning community [30] and the statistics community [31]. Boosting algorithm shows comparable prediction results on benchmark problems to the state-of-the-art methods, such as support vector machines, and it is used in many winning solutions in data mining challenges [32]. [33, 34] observed that Boosting could be viewed as a gradient descent algorithm in some function space. [35, 36, 37] made further contributions in discovering the connection between the Boosting algorithm and the framework of statistical estimation. They laid out a general framework in which Boosting algorithm is considered as numerical optimization using steepest descent minimization, and opened the door for applications other than classification.

To describe the boosting algorithm, we briefly describe regression trees. Let $\mathbf{y} \in \mathbb{R}$ denote the response variable that we are going to predict. In this paper, $\mathbf{y}$ is the solar irradiance. Let $\mathbf{X} \in \mathscr{D} \subset \mathbb{R}^d$ be the input features, where $\mathscr{D}$ is the feature space and d is the number of input features. Regression trees split the feature space $\mathscr{D}$ into distinct and non-overlapping regions which are called leaves. We denote the leaves as $\mathscr{D}_1, \ldots, \mathscr{D}_T$. The leaves are formulated such that $\mathscr{D}_i \cap \mathscr{D}_j$ is empty when $i \neq j$ and $\bigcup_{i=1}^{T} \mathscr{D}_i = \mathscr{D}$. Each leaf $\mathscr{D}_i$ is associated with a weight w_i. Usually, the weights w_i is set as the mean of response variable in the training data where the corresponding input features are in $\mathscr{D}_i$. To do predictions with a given tree, if the input feature $\mathbf{X} \in \mathscr{D}_i$, then the response is predicted as w_i.

To build a regression tree with a given training data $\{(\mathbf{X}_1, \mathbf{y}_1), \ldots, (\mathbf{X}_n, \mathbf{y}_n)\}$, we recursively split the feature space $\mathscr{D}$ into sub-regions such that the residual sum of squares (RSS) is minimized. In the first step, we select a input feature X_j from the feature set $\mathbf{X} = \{X_1, \ldots, X_d\}$ and a cut-point $s \in \mathbb{R}$ and define $\mathscr{D}_1(j,s) = \{\mathbf{X} \in \mathscr{D} | X_j \leq s\}$ and $\mathscr{D}_2(j,s) = \{\mathbf{X} \in \mathscr{D} | X_j > s\}$. The feature X_j and the cut-point s are selected to minimize the RSS,

$$\sum_{i:\mathbf{X}_i \in \mathscr{D}_1(j,s)} (\mathbf{y}_i - \bar{\mathbf{y}}_{\mathscr{D}_1(j,s)})^2 + \sum_{i:\mathbf{X}_i \in \mathscr{D}_2(j,s)} (\mathbf{y}_i - \bar{\mathbf{y}}_{\mathscr{D}_2(j,s)})^2, \tag{7}$$

where $\bar{\mathbf{y}}_{\mathscr{D}_1(j,s)} = \sum_{i:\mathbf{X}_i \in \mathscr{D}_1(j,s)} \mathbf{y}_i / n_1$, where n_1 is number of samples such that the input feature $\mathbf{X}_i \in \mathscr{D}_1(j,s)$, and $\bar{\mathbf{y}}_{\mathscr{D}_2(j,s)}$ is defined similarly. The algorithm proceeds by splitting $\mathscr{D}_1(j,s)$ and $\mathscr{D}_2(j,s)$ using the same idea of splitting $\mathscr{D}$ until stopping conditions are satisfied. See [38] for a thorough description of regression trees.

To state the boosting algorithm in the most general form, consider predicting the response $\mathbf{y}$ by a function $f^*(\mathbf{X})$ of input features $\mathbf{X}$ such that the risk is minimized,

$$f^*(\mathbf{X}) = \arg\min_{f(.)} \mathbb{E}[\rho(\mathbf{y} - f(\mathbf{X}))], \tag{8}$$

where $\rho(.,.)$ is a loss function, such as the squared error loss $\rho(y,f) = (y-f)^2$, and arg min stands for argument of the minimum. The boosting algorithm approximates the solution $f^*(\mathbf{X})$ by an additive model of the form

$$f(\mathbf{X}) = \sum_{i=1}^{m} f_i(\mathbf{X}), \tag{9}$$

where $f_i(\mathbf{X}), i = 1,\ldots,m$ are base learners. In this paper, the base leaner is the regression tree. The number of additive functions m is the primary parameter in the algorithm. The boosting algorithm is summarized in Algorithm 1. Essentially, when the loss function is the squared error loss, the boosting algorithm iteratively fits regression trees to the residuals of previous fits. Gradually, the errors made in previous fits are corrected by latter fits. The bias and variance of the boosting algorithm depend on the number of boosting trees m, and the step length ν. The parameters m and ν can be tuned by cross-validation. In this paper, we use the R [39] package `xgboost` [40] and apply the boosting algorithm to the solar irradiance prediction.

Algorithm 1: The boosting algorithm

Input: Data: $\{(\mathbf{X}_1,\mathbf{y}_1),\ldots,(\mathbf{X}_n,\mathbf{y}_n)\}$; The number of additive functions: m; A step length: ν.

Output: The predictor $f(\mathbf{X}) = \sum_{i=1}^{m} f_i(\mathbf{X})$

Initialize: Set $f_0(\mathbf{X}) = \arg\min_{f\in\mathbb{R}} \sum_{i=1}^{n} \rho(\mathbf{y}_i, f)$

for $i = 1 : m$ **do**

1. Compute the negative gradient:

$$\mathbf{U}_i = -\frac{\partial}{\partial f}\rho(\mathbf{y}_i, f)\,\Big|_{f=\sum_{j=1}^{i-1} f_j(\mathbf{X}_i)}$$

2. Fit the negative gradient $\mathbf{U}_i$ by the inputs $\mathbf{X}_i$ using regression trees:

$$(\mathbf{U}_i, \mathbf{X}_i)_{i=1}^{n} \xrightarrow{trees} g_i(.)$$

3. Update $f_i(.) = \nu g_i(.)$

end

2.3. Random Forest and Generalized Random Forest

Random forest [41] is a variation of the bagging algorithm. Random forest is versatile enough to be applied to data with missing values, mixed type observations and large scale problems. Random forest is recommended as the best off-the-shelf leaner for small to moderate problems by [42] after comparing 179 classifiers using 121 data sets.

According to the analysis in [41], the generalization error of the bagging algorithm depends on the strength of individual trees and the correlation between the trees. To decrease correlation between trees, [41] suggested using

randomly selected features at each node when fitting trees. As the bagging algorithm, the random forest algorithm repeatedly fits trees to the bootstrap samples of the training set. Different from the bagging algorithm, at each candidate split in the tree fitting process, the candidates of the splitting features are restricted to be a random subset of all the input features. The number of candidate features is an important parameter of the random forest algorithm, and we tune it by cross validation in this paper. Algorithm 2 is a schematic illustration of a random forest algorithm. In this paper, we use the R [39] package `ranger` [43] and apply the random forest to the solar irradiance prediction.

Algorithm 2: The random forest

Input: Data: $\{(\mathbf{X}_1,\mathbf{y}_1),\ldots,(\mathbf{X}_n,\mathbf{y}_n)\}$; The number of trees: m; The number of candidate features for splitting the nodes of each tree: k.

Output: The predictor $f(\mathbf{X}) = m^{-1}\sum_{i=1}^{m} f_i(\mathbf{X})$

for $i = 1 : m$ **do**

1. Randomly select n samples $\{(\mathbf{X}_1^*,\mathbf{y}_1^*),\ldots,(\mathbf{X}_n^*,\mathbf{y}_n^*)\}$ from the data $\{(\mathbf{X}_1,\mathbf{y}_1),\ldots,(\mathbf{X}_n,\mathbf{y}_n)\}$ with replacement.

2. Construct a regression tree $f_i(\mathbf{X})$ using $\{(\mathbf{X}_1^*,\mathbf{y}_1^*),\ldots,(\mathbf{X}_n^*,\mathbf{y}_n^*)\}$. At each node of the tree, use k randomly selceted features as the candidate splitting variable.

end

Recently, [44] proposed the generalized random forest (GRF) which can be used to fit statistical models identified as local estimation equations. The GRF has theoretical guarantees and can be applied to a broad range of statistical learning problems. The GRF is a generalization of the random forest in that it fits a forest of regression trees to predict the response. Different from the random forest, the GRF fits trees to random splitting of the training data, and use the out-of-sample data to estimate the weights w_i of the trees. Specifically, for a given splitting rate $0 < s < 1$, to train a tree the GRF first selects $[ns]$ sub-sample from the training data $\{(\mathbf{X}_1,\mathbf{y}_1),\ldots,(\mathbf{X}_n,\mathbf{y}_n)\}$ without replacement. The selected sub-sample is used to train the structure of a regression tree, and the weights on the leaves of tree are determined by the un-selected samples. The splitting rate s is determined by cross-validation in this paper. We use the R [39] package `grf` [45] and apply the GRF to the solar irradiance prediction.

3. TMY3 dataset

Typical Meteorological Year (TMY) datasets consist of hourly average weather conditions for the twelve months considered to be typical for a specific location. TMY data are commonly used to facilitate proper performance comparisons of energy systems whose performance depends on weather conditions such as the solar photovoltaic (PV) systems or wind turbines. For the United States, the Typical Meteorological Year version 3 (TMY3) data are available for download at http://rredc.nrel.gov/solar/old_data/nsrdb/1991-2005/tmy3 [46]. There are a total of 1,020

TMY3 ground stations in the United States, and we pick six locations from them for our study, as shown in Figure 2. To predict the solar irradiance (global horizontal irradiance; GHI), we select six meteorological drivers based on information from previous similar studies [8, 17], as shown in Table 1.

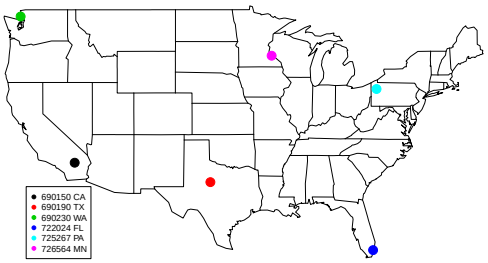

Figure 2: TMY3 stations with the United States Air Force (USAF) codes used by the National Climatic Data Center for station identification.

Table 1: Solar irradiance (GHI) and Meteorological drivers

Name	Description (Unit)
GHI	Global horizontal irradiance (Wh/m^2)
TotalSkyCover	Amount of sky dome covered by clouds or obscuring phenomena at the time indicated (Tenth*)
Dry-bulb	Dry-bulb temperature at the time indicated ($^\circ$C)
Dew-point	Dew-point temperature at the time indicated ($^\circ$C)
RHumidity	Relative humidity at the time indicated (%)
WindSpeed	Wind speed at the time indicated (m/s)
Visibility	Distance to discernable remote objects at the time indicated (m)

$*$ Tenth is 1/10 (e.g., 1 Tenth is equivalent to 10 %).

Further, we only consider the data during the daytime, which has non-zero irradiation. Figure 3 shows the pairwise correlation coefficients between the GHI and six weather variables during the daytime for each station. It can be shown that both temperature variables (Dry-bulb and Dew-point) are positively correlated with GHI, whereas the sky cover (TotalSkyCover) and the humidity (RHumidity) are negatively correlated with the solar irradiance over all stations. However, WindSpeed and Visibility show different patterns for each location.

Besides the weather variables, we also consider the time-domain relations of the hourly data. Figures 4–5 illustrate the monthly and daily distributions of the irradiance at noon, respectively. As expected, Figure 4 shows that the GHI is relatively high during the summer season, while the less irradiance is measured during the winter over all stations. However, the variances of the irradiance in each month show different patterns for each location. While the CA station shows relatively stable GHI, the FL, PA, or MN stations show relatively large variations in GHI in each month. Figure 5 shows that the daily irradiance at noon is almost equally distributed. Figure 6 shows the distributions of the hourly solar irradiance for each location. As expected, the hourly mean shows a bell-shaped distribution with a peak around noon in every station. Besides the weather variables in Table 1, we also consider month, day of the month, and hour of the day as the predictors for the GHI based on the exploratory data analysis above.

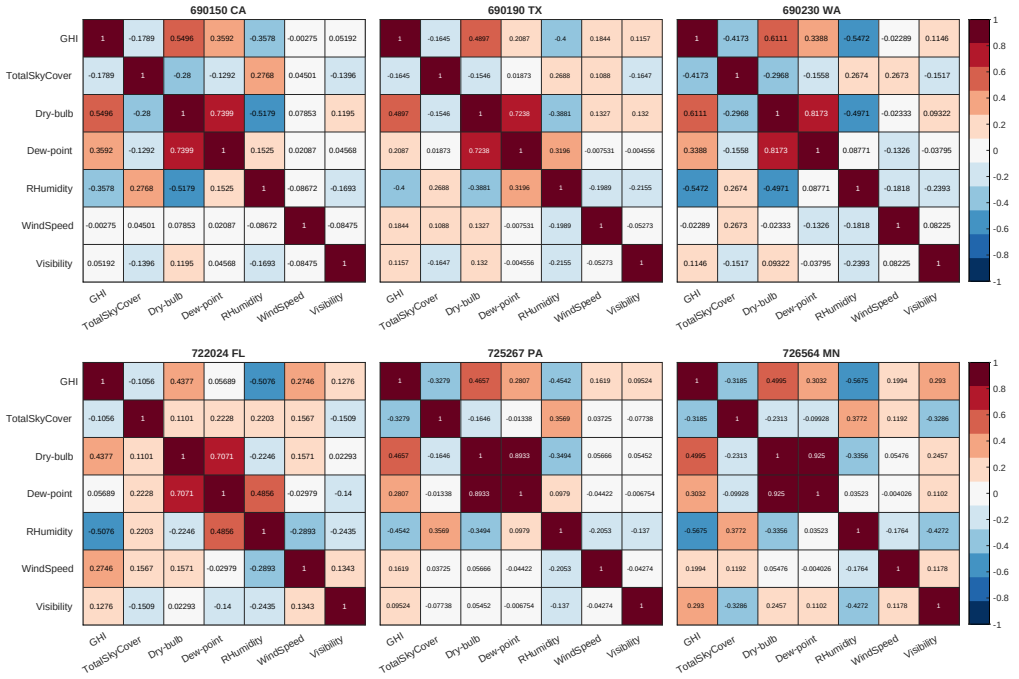

Figure 3: Correlation coefficients between Solar irradiance and Meteorological drivers

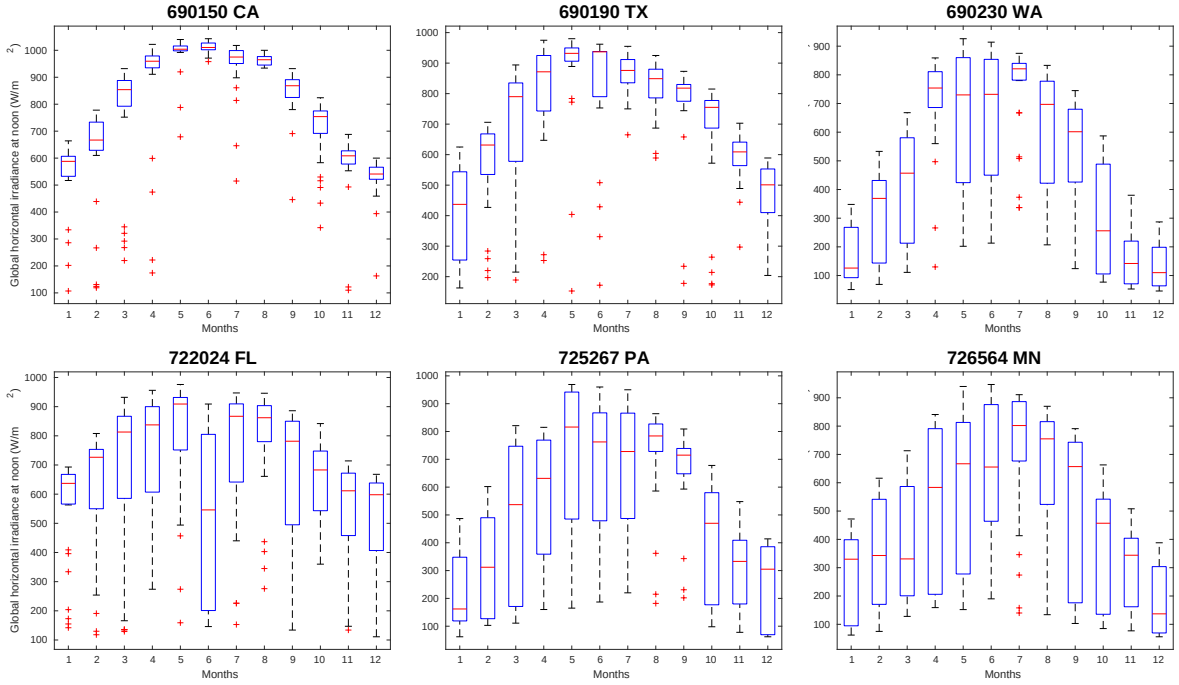

Figure 4: Distribution of Irradiance at Noon

Furthermore, we add a one-hour lagged GHI value as an explanatory variable in the model, similar to a time series analysis. Thus, we consider a total of ten variables to predict the GHI, as shown in Table 2. We choose the first six

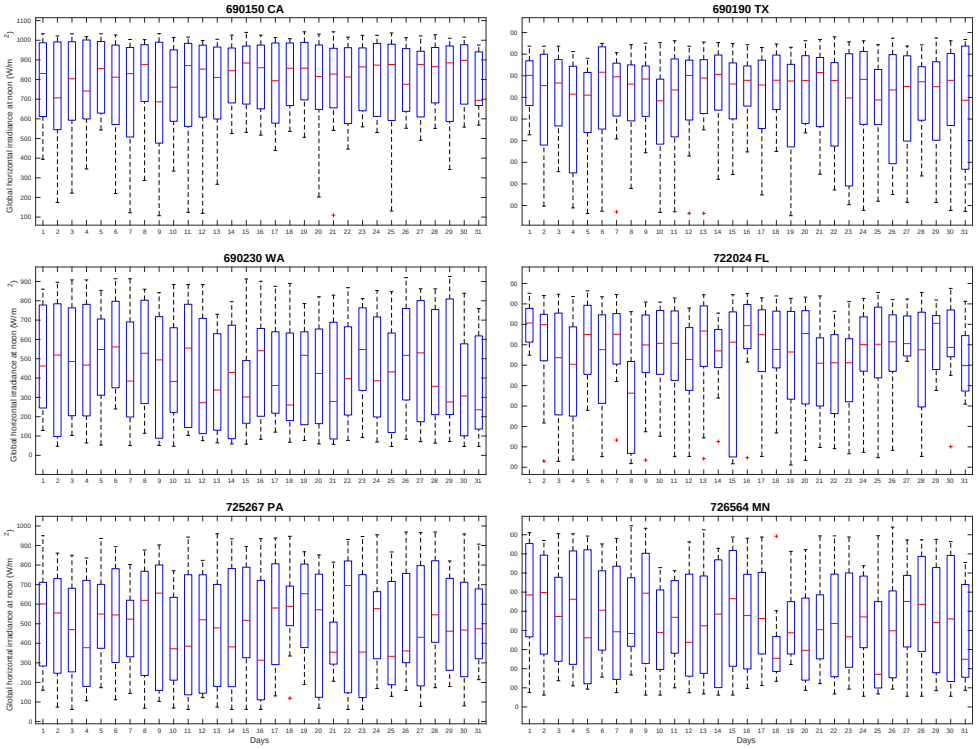

Figure 5: Daily distribution of Irradiance at Noon

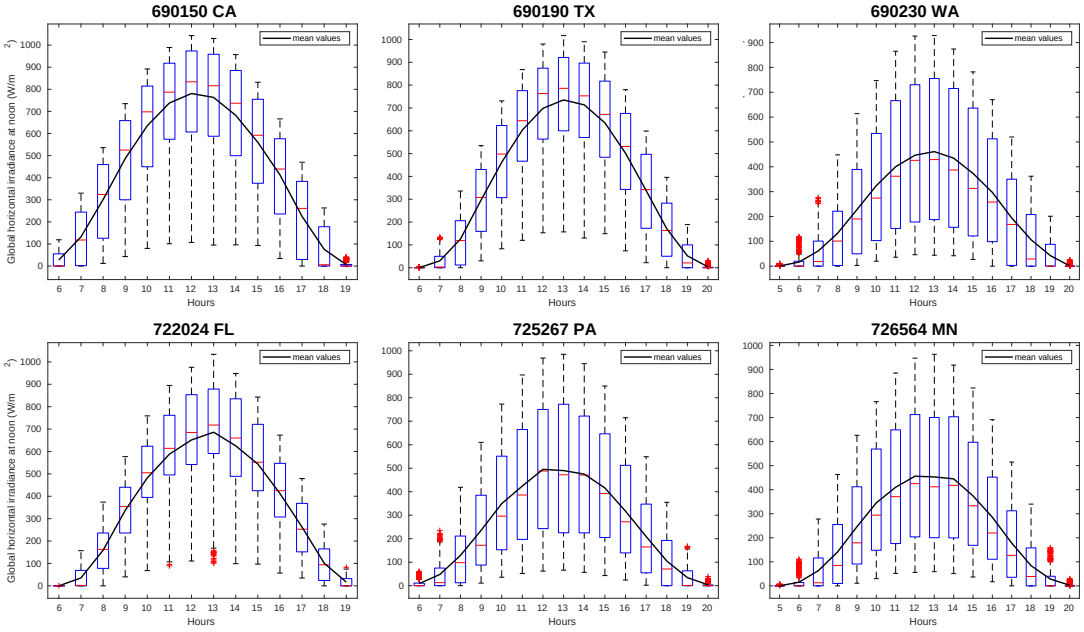

Figure 6: Distribution of Irradiance in the Daytime

months, January to June, for the training dataset. The testing dataset is defined with the next three days from the last day of the training dataset.

Table 2: Variables in the Models

Variable Name	Description
GHI	Response variable. Global horizontal irradiance (Wh/m^2)
Date-MM	Month
Date-DD	Day of the month
Time-HH	Hour of the day
TotCldtenths	Total sky cover (Tenth*)
DrybulbC	Dry-bulb temperature ($^\circ$C)
DewpointC	Dew-point temperature ($^\circ$C)
RHum	Relative humidity (%)
Wspdms	Wind speed (m/s)
HVism	Visibility (m)
GHI-Lag1h	One-hour lagged GHI value

$*$ Tenth is 1/10 (e.g., 1 Tenth is equivalent to 10 %).

4. Data analysis

This section presents a statistical analysis of the TMY data. The performance of four ensemble learning methods are demonstrated in predicting solar irradiance from different stations. Here, also the results from ensemble learning methods are compared to those obtained by GPR and SVM models. More details about SVM and GPR can be found in [47] and [48], respectively. In addition, identification variables importance in RF, GRF, and BS models has assessed.

4.1. Model performance evaluation

To assess the prediction performance, we compare some statistical indicators as follows:

- Coefficient of determination (R^2)

$$R^2 = \frac{\sum_{i=1}^{n_{test}} [(y_{i,test} - \bar{y}_{test}) \cdot (\hat{y}_{i,test} - \bar{\hat{y}}_{test})]^2}{\sqrt{\sum_{i=1}^{n_{test}} (y_{i,test} - \bar{y}_{test})^2} \cdot \sqrt{\sum_{i=1}^{n_{test}} (\hat{y}_{i,test} - \bar{\hat{y}}_{test})^2}}, \tag{10}$$

- Root mean square prediction error (RMSPE)

$$\text{RMSPE} = \sqrt{\frac{\sum_{i=1}^{n_{test}} (y_{i,test} - \hat{y}_{i,test})^2}{n_{test}}}, \tag{11}$$

- Mean absolute prediction error (MAPE)

$$\text{MAPE} = \frac{\sum_{i=1}^{n_{test}} |y_{i,test} - \hat{y}_{i,test}|}{n_{test}}, \tag{12}$$

where $y_{i,test}$ and $\hat{y}_{i,test}$ are the ith GHI value and its predicted value, respectively, in the testing dataset, for $i = 1,\ldots,n_{test}$, where n_{test} is the total number of the testing dataset. While the higher R^2 means the better prediction, the lower RMSPE or MAPE is more accurate in the prediction. Furthermore, for the model comparison, we also predict the solar irradiance using the SVM with five different kernels and the GPR with four different kernels, respectively. Thus, for the prediction, we conduct a total of thirteen algorithms, as shown in Table 3.

Table 3: Models

Name	Description	Kernel function
SVM-L	Support Vector with the Linear kernel	$\mathbf{x}_i^\top \mathbf{x}_j$
SVM-Q	Support Vector with the Quadratic kernel	$(1+\mathbf{x}_i^\top \mathbf{x}_j)^2$
SVM-C	Support Vector with the Cubic kernel	$(1+\mathbf{x}_i^\top \mathbf{x}_j)^3$
SVM-MG	Support Vector with the Medium Gaussian kernel	$\exp(-\sqrt{p}\|\mathbf{x}_i-\mathbf{x}_j\|^2)$
SVM-CG	Support Vector with the Coarse Gaussian kernel	$\exp(-4\sqrt{p}\|\mathbf{x}_i-\mathbf{x}_j\|^2)$
GPR-RQ	Gaussian Process Regression with the Rational Quadratic kernel	$\sigma_f^2\left(1+\frac{r^2}{2\alpha\sigma_l^2}\right)^{-\alpha}$
GPR-SE	Gaussian Process Regression with the Squared Exponential kernel	$\sigma_f^2 \exp\left(\frac{1}{2}\frac{r^2}{\sigma_l^2}\right)$
GPR-M52	Gaussian Process Regression with the Matern 5/2 kernel	$\sigma_f^2\left(1+\frac{\sqrt{5}r}{\sigma_l}+\frac{5r^2}{3\sigma_l^2}\right)\exp\left(\frac{\sqrt{5}r}{\sigma_l}\right)$
GPR-Exp	Gaussian Process Regression with the Exponential kernel	$\sigma_f^2 \exp\left(\frac{r}{\sigma_l}\right)$
ETR-BS	Ensemble of Tree: Boosted Trees	
ETR-BG	Ensemble of Tree: Bagged Trees	
ETR-RF	Ensemble of Tree: Random Forest	
ETR-GRF	Ensemble of Tree: Generalized Random Forest	

where $r = \sqrt{(\mathbf{x}_i-\mathbf{x}_j)^\top(\mathbf{x}_i-\mathbf{x}_j)}$ in the GPR kernel function.

4.2. Results

We train each model in Table 3 with the training dataset for each TMY3 station in Figure 2. Then, we predict the solar irradiance (GHI) for the testing dataset from each trained model. Those predicted solar irradiances are illustrated in Figures 7–9 for each location. To evaluate the prediction performance, we compute the coefficient of determination (R^2), root-mean-square prediction error (RMSPE) and mean absolute prediction error (MAPE) for each model and location, respectively. These statistical indicators are provided in Table 4. For the California (CA) station, ETR-GRF is uniformly the best. For the Texas (TX) station, ETR-BS is the best in terms of R^2, while ETR-GRF is the best in terms of RMSPE and MAPE. For the Washington (WA) and Pennsylvania (PA) stations, ETR-RF is uniformly the best. For the Florida (FL) station, ETR-GRF is the best in terms of R^2, while SVM-Q is the best in terms of RMSPE and MAPE. For the Minnesota (MN) station, ETR-RF, ETR-BG, and ETR-GRF are the best in terms of R^2, RMSPE, and MAPE, respectively. Thus, among the thirteen models in Table 3, which we considered, there is not a single model that is uniformly superior to others. However, in a total of eighteen evaluations (three indicators for six locations) in Table 4, the Ensemble learning model takes the sixteen first places (89%: sixteen out of eighteen): two of single first place by ETR-BS and ETR-BG, and two of seven first places by ETR-RF and ETR-GRF, respectively.

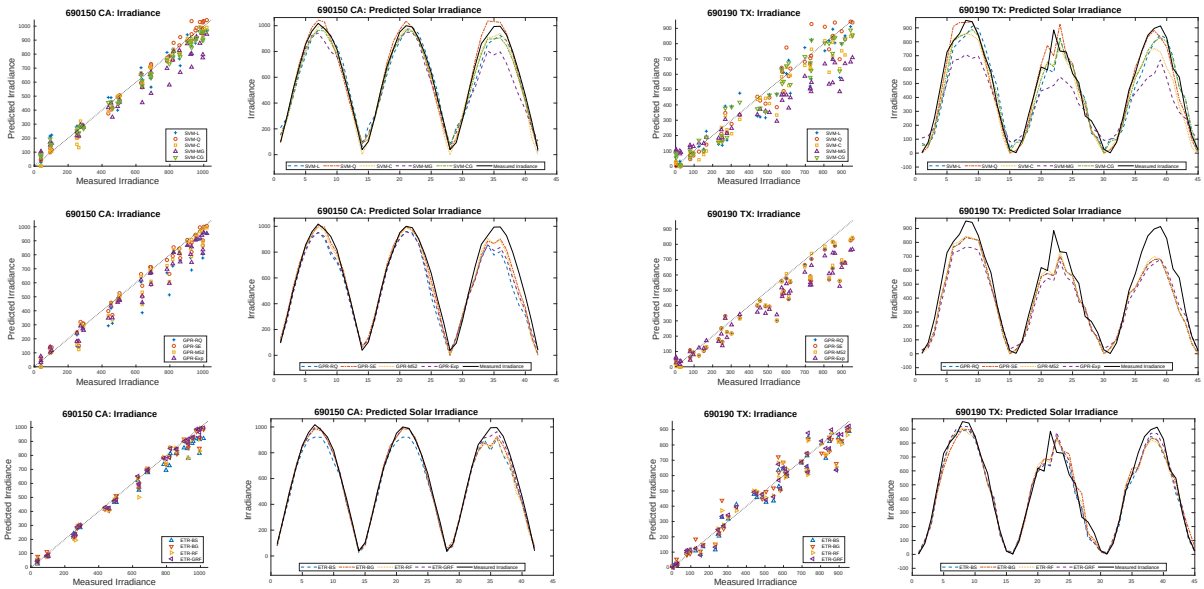

Figure 7: Predicted Solar Irradiance (CA and TX)

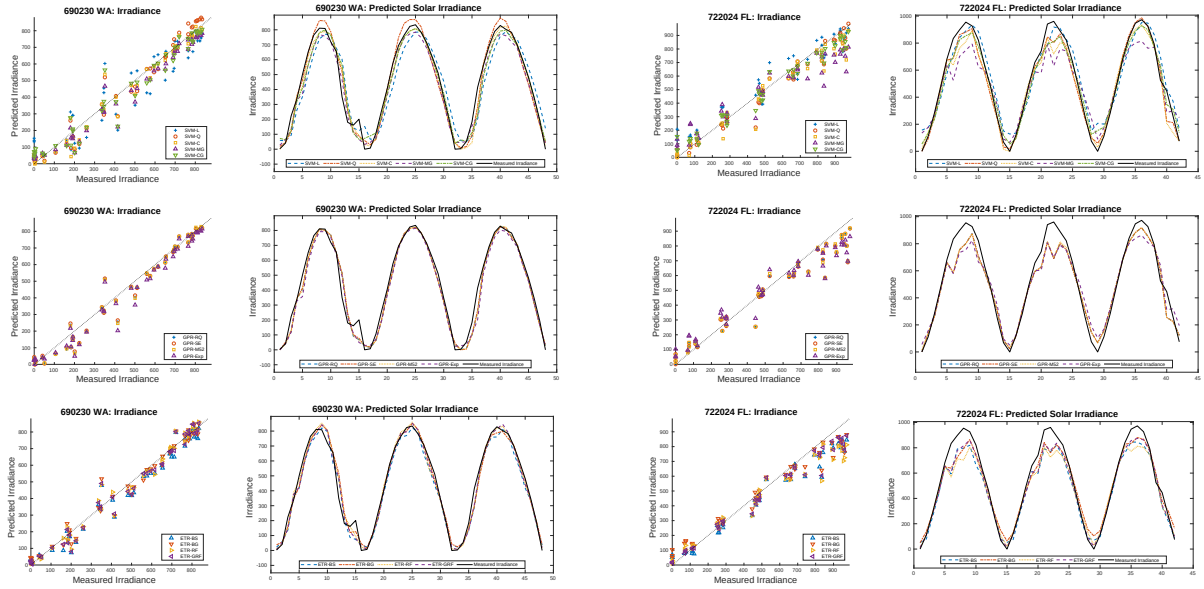

Figure 8: Predicted Solar Irradiance (WA and FL)

Thus, we could say that Ensemble learning models are better than SVM or GPR overall, especially Random Forest (ETR-RF) or Generalized Random Forest (ETR-GRF): each of them takes seven first places out of eighteen (39%), respectively.

We also investigate the prediction errors of each model from the testing datasets, where the prediction error is defined as the difference between the observed value and the predicted value ($y_{i,test} - \hat{y}_{i,test}$ in the Equations 11 and

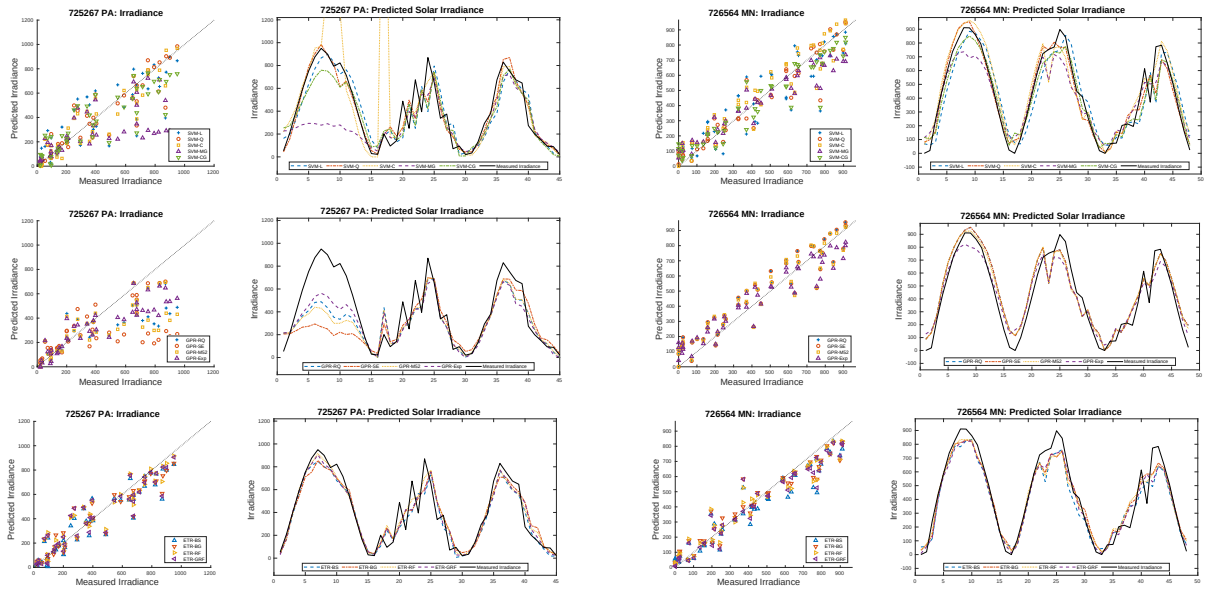

Figure 9: Predicted Solar Irradiance (PA and MN)

Table 4: R^2s, RMSPEs and MAPEs are from the testing datasets.

	CA			TX			WA		
Model	R^2	RMSPE	MAPE	R^2	RMSPE	MAPE	R^2	RMSPE	MAPE
SVM-L	97.04	61.67	48.67	90.82	100.24	81.32	88.20	101.03	85.53
SVM-Q	98.77	38.47	30.54	95.51	70.23	50.58	96.91	55.75	44.28
SVM-C	98.81	43.14	34.32	95.51	87.80	65.13	96.96	53.11	36.58
SVM-MG	95.14	97.89	73.77	94.24	166.16	135.85	97.14	60.97	50.29
SVM-CG	99.15	47.38	40.62	94.50	79.29	64.56	95.17	65.53	46.92
GPR-RQ	94.07	106.96	77.09	94.14	115.14	85.21	97.26	52.66	36.12
GPR-SE	97.40	58.53	44.46	94.14	115.14	85.21	97.33	51.15	35.73
GPR-M52	96.83	70.62	47.90	94.25	110.73	80.63	97.27	52.82	35.35
GPR-Exp	97.02	84.09	63.91	95.43	129.75	101.05	96.67	60.17	42.65
ETR-BS	98.95	59.27	45.01	96.25	65.71	47.58	97.31	52.69	42.09
ETR-BG	98.96	39.27	28.79	95.52	67.75	48.72	97.64	44.65	31.50
ETR-RF	98.34	48.70	31.43	95.95	65.48	46.36	98.05	41.59	28.73
ETR-GRF	99.63	24.07	19.76	96.12	62.35	42.27	97.82	44.32	31.79

	FL			PA			MN		
Model	R^2	RMSPE	MAPE	R^2	RMSPE	MAPE	R^2	RMSPE	MAPE
SVM-L	93.97	89.31	75.90	72.16	158.84	115.07	83.03	123.48	101.31
SVM-Q	97.19	63.13	42.99	84.35	118.90	76.02	92.19	83.91	59.32
SVM-C	95.57	82.23	56.34	22.91	511.95	180.89	91.38	93.87	81.46
SVM-MG	92.19	127.09	101.24	41.84	256.24	174.31	93.78	106.04	83.11
SVM-CG	97.50	68.50	57.20	75.94	151.53	114.30	91.05	96.20	71.12
GPR-RQ	95.63	88.13	63.27	72.43	196.54	135.89	90.93	99.05	86.26
GPR-SE	95.63	88.13	63.27	39.03	259.46	169.39	90.93	99.05	86.26
GPR-M52	95.45	87.65	62.12	65.51	212.19	143.12	91.28	96.61	82.78
GPR-Exp	94.49	103.16	81.20	82.74	171.28	124.34	92.73	95.71	78.66
ETR-BS	97.61	88.51	70.31	89.34	103.20	75.00	93.60	91.47	69.30
ETR-BG	97.53	78.30	60.77	88.50	104.80	79.68	94.70	79.09	59.15
ETR-RF	96.27	100.19	71.60	91.87	86.79	65.02	94.99	79.38	60.09
ETR-GRF	98.00	72.99	52.36	88.58	102.72	72.64	94.83	80.41	58.58

12). The boxplots of the prediction errors from each model are illustrated in Figure 10 for each location. From these boxplots, we can assess the distribution of the prediction errors for each model, and it supports what we found from Table 4: Ensemble learning models show not uniformly but overall better performance than SVMs or GPRs. Ensemble learning models (ETRs) provide shorter IQRs (interquartile range; height of the box in the boxplot) and

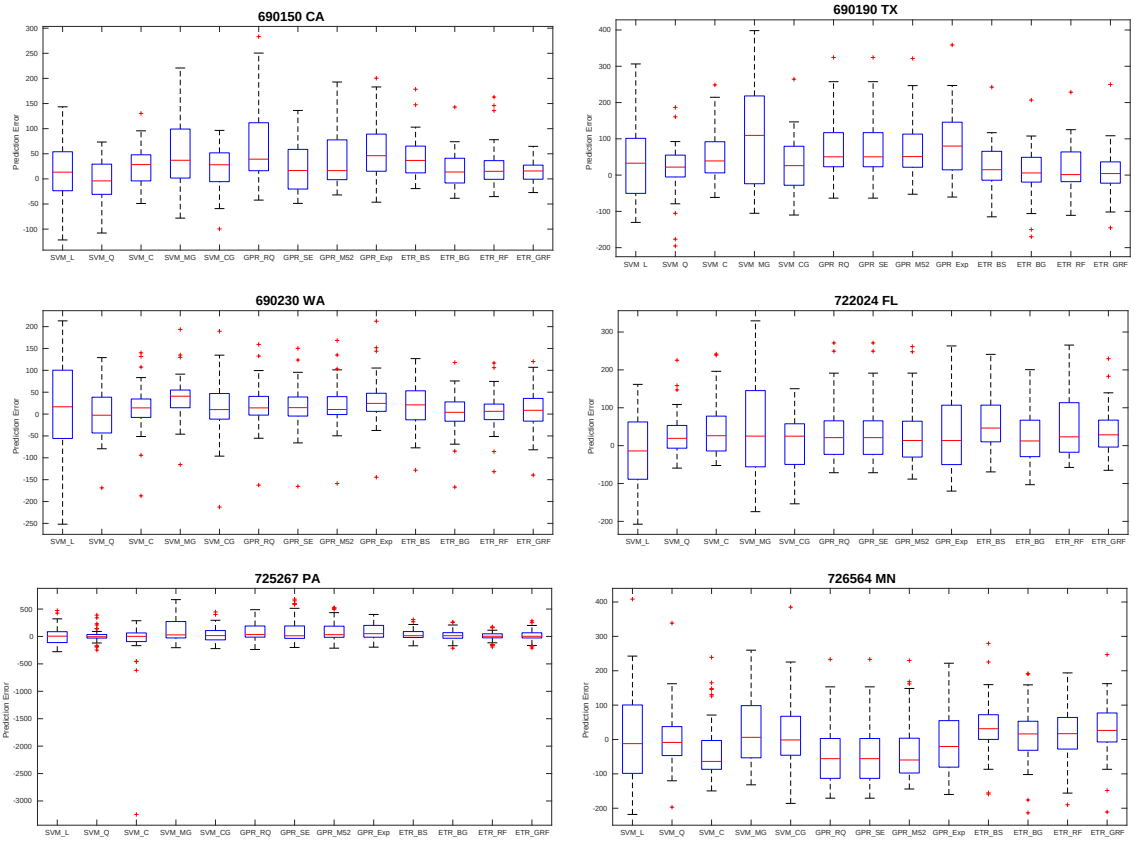

Figure 10: Prediction errors from the testing datasets.

whiskers compared to SVMs and GPRs, and their medians (red line in the middle inside of the box) are close to zero. This means that the ETR produces the prediction errors which are distributed around zero. When we focus on each model, SVM-Q also provides comparable results in some stations (TX, FL, PA, and MN). But, in CA and WA, it shows heavy-tailed and skewed distributions of the prediction error: skewed to the left in CA and skewed to the right in WA. ETR-RF looks the best in WA and PA, but it also provides heavy-tailed and skewed (to the left in TX and to the right in FL) distributions. However, ETR-GRF provides the symmetric distribution around zero with relatively short whiskers (i.e., non-heavy-tailed) in every station. Thus, among all of the thirteen models, we could say that Generalized Random Forest (ETR-GRF) provides robust performance in terms of the prediction error distribution.

4.3. Variable importance identification using RF, GRF and BS

One important step to help understand the results obtained by the RF, GRF, and BS models is to assess variable contribution or importance for predictive model. It should be not that some variables do not have relevant contributions to the model and thus they can be ignored because they make the model more complex. Accordingly, the identification of important variables are necessary for eliminating variables with the smallest contribution. To do so, the RF, GRF,

and BS models include a built-in function that allows evaluating the variables importance. Figure 11(a-c) displays the variable importance bar chart using the RF, GRF, and BS models by stations. The larger a variable importance score, the more important a feature is. The sum of the variable importance score for all the features is 100 in a model. Figure 11(a-c) indicates that GHI-Lag1h has the largest contribution in the three models compared to the other variables. This result is clearly confirming the importance of considering a time-dependent nature of data when designing the ensemble models.

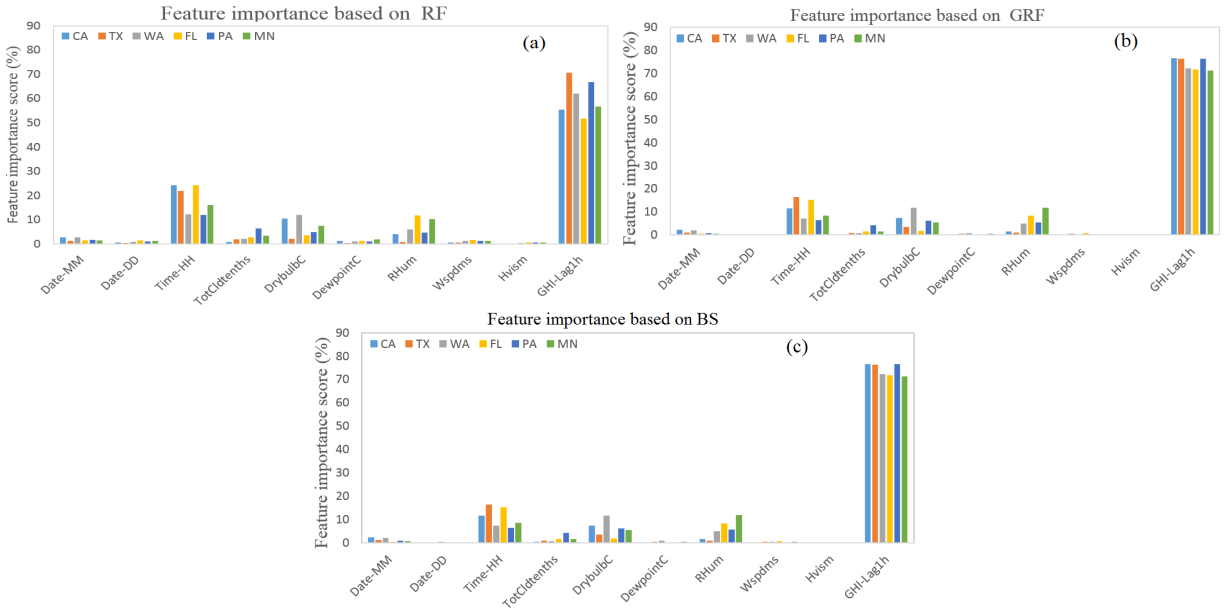

Figure 11: Feature importance identification based on (a) RF, (b) GRF and (c) BS by station.

5. Conclusion

In this paper, we present four ensemble learning-based models (Boosted Trees, Bagged Trees, Random Forest, and Generalized Random Forest) to provide a reliable short-term prediction of solar irradiance based on meteorological drivers. For the verification of the prediction quality, we use the Typical Meteorological Year version 3 (TMY3) data corresponding to six locations we choose. Based on information from previous similar studies, we select six weather variables to predict solar irradiance. We also add the month, day of the month, and hour of the day in the models as the predictors based on the exploratory data analysis of the time-domain relations of the hourly data.

We evaluate the prediction performances via R^2, RMSPE, and MAPE, and compare to the results from Support Vector Machine regressions and Gaussian process regressions. There is not a single model that is the best uniformly for all six locations and in terms of all three statistical indicators. However, results support that Ensemble learning

models offer superior prediction performance overall by taking sixteen first places out of a total of eighteen evaluations (three indicators for six locations).

In many practical data, such as solar irradiance and meteorological data, multiple times series data are collected from different spatial locations with necessary spatiotemporal dependence, which is not considered in ensemble learning models and machine learning models. To further improve solar irradiance prediction, in future works, it is planned to develop ensemble learning methods that consider spatiotemporal information.

Acknowledgement

The research reported in this publication was supported by funding from King Abdullah University of Science and Technology (KAUST), Office of Sponsored Research (OSR) under Award No: OSR-2019-CRG7-3800.

References

[1] Y. Feng, W. Hao, H. Li, N. Cui, D. Gong, and L. Gao, "Machine learning models to quantify and map daily global solar radiation and photovoltaic power," *Renewable and Sustainable Energy Reviews*, vol. 118, p. 109393, 2020.

[2] Y. El Mghouchi, E. Chham, M. Krikiz, T. Ajzoul, and A. El Bouardi, "On the prediction of the daily global solar radiation intensity on south-facing plane surfaces inclined at varying angles," *Energy conversion and management*, vol. 120, pp. 397–411, 2016.

[3] Y. Feng, D. Gong, Q. Zhang, S. Jiang, L. Zhao, and N. Cui, "Evaluation of temperature-based machine learning and empirical models for predicting daily global solar radiation," *Energy Conversion and Management*, vol. 198, p. 111780, 2019.

[4] E. F. Alsina, M. Bortolini, M. Gamberi, and A. Regattieri, "Artificial neural network optimisation for monthly average daily global solar radiation prediction," *Energy conversion and management*, vol. 120, pp. 320–329, 2016.

[5] C.-L. Fu and H.-Y. Cheng, "Predicting solar irradiance with all-sky image features via regression," *Solar Energy*, vol. 97, pp. 537–550, 2013.

[6] H.-Y. Cheng and C.-C. Yu, "Multi-model solar irradiance prediction based on automatic cloud classification," *Energy*, vol. 91, pp. 579–587, 2015.

[7] D. Torregrossa, J.-Y. Le Boudec, and M. Paolone, "Model-free computation of ultra-short-term prediction intervals of solar irradiance," *Solar Energy*, vol. 124, pp. 57–67, 2016.

[8] K. Y. Bae, H. S. Jang, and D. K. Sung, "Hourly solar irradiance prediction based on support vector machine and its error analysis," *IEEE Transactions on Power Systems*, vol. 32, no. 2, pp. 935–945, 2017.

[9] C. K. Kim, H.-G. Kim, Y.-H. Kang, C.-Y. Yun, and S. Y. Kim, "Probabilistic prediction of direct normal irradiance derived from global horizontal irradiance over the korean peninsula by using monte-carlo simulation," *Solar Energy*, vol. 180, pp. 63–74, 2019.

[10] S. Ghimire, R. C. Deo, N. Raj, and J. Mi, "Wavelet-based 3-phase hybrid svr model trained with satellite-derived predictors, particle swarm optimization and maximum overlap discrete wavelet transform for solar radiation prediction," *Renewable and Sustainable Energy Reviews*, vol. 113, p. 109247, 2019.

[11] L. Wu, G. Huang, J. Fan, F. Zhang, X. Wang, and W. Zeng, "Potential of kernel-based nonlinear extension of arps decline model and gradient boosting with categorical features support for predicting daily global solar radiation in humid regions," *Energy conversion and management*, vol. 183, pp. 280–295, 2019.

[12] T. McCandless, S. Haupt, and G. S. Young, "A model tree approach to forecasting solar irradiance variability," *Solar Energy*, vol. 120, pp. 514–524, 2015.

[13] Y. Hirata and K. Aihara, "Improving time series prediction of solar irradiance after sunrise: Comparison among three methods for time series prediction," *Solar Energy*, vol. 149, pp. 294–301, 2017.

[14] Â. Frimane, T. Soubdhan, J. M. Bright, and M. Aggour, "Nonparametric bayesian-based recognition of solar irradiance conditions: Application to the generation of high temporal resolution synthetic solar irradiance data," *Solar Energy*, vol. 182, pp. 462–479, 2019.

[15] M. David, F. Ramahatana, P.-J. Trombe, and P. Lauret, "Probabilistic forecasting of the solar irradiance with recursive arma and garch models," *Solar Energy*, vol. 133, pp. 55–72, 2016.

[16] M. Kakimoto, Y. Endoh, H. Shin, R. Ikeda, and H. Kusaka, "Probabilistic solar irradiance forecasting by conditioning joint probability method and its application to electric power trading," *IEEE Transactions on Sustainable Energy*, vol. 10, no. 2, pp. 983–993, 2018.

[17] X. Qing and Y. Niu, "Hourly day-ahead solar irradiance prediction using weather forecasts by lstm," *Energy*, vol. 148, pp. 461–468, 2018.

[18] K. Kaba, M. Sarıgül, M. Avcı, and H. M. Kandırmaz, "Estimation of daily global solar radiation using deep learning model," *Energy*, vol. 162, pp. 126–135, 2018.

[19] T. Watanabe and D. Nohara, "Prediction of time series for several hours of surface solar irradiance using one-granule cloud property data from satellite observations," *Solar Energy*, vol. 186, pp. 113–125, 2019.

[20] J. O. Kamadinata, T. L. Ken, and T. Suwa, "Sky image-based solar irradiance prediction methodologies using artificial neural networks," *Renewable Energy*, vol. 134, pp. 837–845, 2019.

[21] Y. Feng, N. Cui, Y. Chen, D. Gong, and X. Hu, "Development of data-driven models for prediction of daily global horizontal irradiance in northwest china," *Journal of Cleaner Production*, vol. 223, pp. 136–146, 2019.

[22] W. Liao, Y. Heo, and S. Xu, "Simplified vector-based model tailored for urban-scale prediction of solar irradiance," *Solar Energy*, vol. 183, pp. 566–586, 2019.

[23] H. Verbois, R. Huva, A. Rusydi, and W. Walsh, "Solar irradiance forecasting in the tropics using numerical weather prediction and statistical learning," *Solar Energy*, vol. 162, pp. 265–277, 2018.

[24] Z. Li, K. Goebel, and D. Wu, "Degradation modeling and remaining useful life prediction of aircraft engines using ensemble learning," *Journal of Engineering for Gas Turbines and Power*, vol. 141, no. 4, p. 041008, 2019.

[25] F. Harrou, A. Saidi, and Y. Sun, "Wind power prediction using bootstrap aggregating trees approach to enabling sustainable wind power integration in a smart grid," *Energy Conversion and Management*, vol. 201, p. 112077, 2019.

[26] L. Breiman, "Bagging predictors," *Machine learning*, vol. 24, no. 2, pp. 123–140, 1996.

[27] C. D. Sutton, "Classification and regression trees, bagging, and boosting," *Handbook of statistics*, vol. 24, pp. 303–329, 2005.

[28] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of computer and system sciences*, vol. 55, no. 1, pp. 119–139, 1997.

[29] R. E. Schapire, Y. Freund, P. Bartlett, W. S. Lee *et al.*, "Boosting the margin: A new explanation for the effectiveness of voting methods," *The annals of statistics*, vol. 26, no. 5, pp. 1651–1686, 1998.

[30] R. E. Schapire, "The boosting approach to machine learning: An overview," in *Nonlinear estimation and classification*. Springer, 2003, pp. 149–171.

[31] P. Bühlmann, T. Hothorn *et al.*, "Boosting algorithms: Regularization, prediction and model fitting," *Statistical Science*, vol. 22, no. 4, pp. 477–505, 2007.

[32] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*. ACM, 2016, pp. 785–794.

[33] L. Breiman, "Arcing classifiers," *Annals of Statistics*, vol. 26, pp. 123–40, 1996.

[34] ——, "Prediction games and arcing algorithms," *Neural computation*, vol. 11, no. 7, pp. 1493–1517, 1999.

[35] J. Friedman, T. Hastie, R. Tibshirani *et al.*, "Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors)," *The annals of statistics*, vol. 28, no. 2, pp. 337–407, 2000.

[36] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of statistics*, pp. 1189–1232, 2001.

[37] ——, "Stochastic gradient boosting," *Computational statistics & data analysis*, vol. 38, no. 4, pp. 367–378, 2002.

[38] L. Breiman, J. Friedman, R. Olshen, and C. Stone, *Classification and Regression Trees*. Chapman & Hall, 1984.

[39] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2019. [Online]. Available: https://www.R-project.org/

[40] T. Chen, T. He, M. Benesty, V. Khotilovich, Y. Tang, H. Cho, K. Chen, R. Mitchell, I. Cano, T. Zhou, M. Li, J. Xie, M. Lin, Y. Geng, and Y. Li, *xgboost: Extreme Gradient Boosting*, 2019, r package version 0.82.1. [Online]. Available: https://CRAN.R-project.org/package=xgboost

[41] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[42] M. Fernández-Delgado, E. Cernadas, S. Barro, and D. Amorim, "Do we need hundreds of classifiers to solve real world classification problems?" *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 3133–3181, 2014.

[43] M. N. Wright and A. Ziegler, "ranger: A fast implementation of random forests for high dimensional data in C++ and R," *Journal of Statistical Software*, vol. 77, no. 1, pp. 1–17, 2017.

[44] S. Athey, J. Tibshirani, S. Wager *et al.*, "Generalized random forests," *The Annals of Statistics*, vol. 47, no. 2, pp. 1148–1178, 2019.

[45] J. Tibshirani, S. Athey, R. Friedberg, V. Hadad, L. Miner, S. Wager, and M. Wright, *grf: Generalized Random Forests (Beta)*, 2019, r package version 0.10.3. [Online]. Available: https://CRAN.R-project.org/package=grf

[46] S. Wilcox and W. Marion, "Users manual for tmy3 data sets," *Technical Report NREL/TP-581-43156, NREL Lab., Golden, CO.*, 2008.

[47] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statistics and computing*, vol. 14, no. 3, pp. 199–222, 2004.

[48] T. Liu, H. Wei, and K. Zhang, "Wind power prediction with missing data using gaussian process regression and multiple imputation," *Applied Soft Computing*, vol. 71, pp. 905–916, 2018.