Solar Power Prediction

1st Kartikeya Sehgal
kartikeya22244@iiitd.ac.in
2022244

2nd Krishna Shukla
krishna22254@iiitd.ac.in
2022254

***Abstract*—This paper explores various techniques for accurately predicting AC power output in solar plants based on data obtained from solar panels and temperature sensors. This project aims to arrive at the most suitable model for the above task by training models of Linear Regression, Random Forests, AdaBoost, XGBoost, and Artificial Neural Networks on the obtained dataset.**

I. Introduction

1) Problem Statement: This project aims to develop machine learning models capable of accurately predicting the solar power generated by two different solar power plants using data from their respective solar panels and weather sensors (temperature reports). By employing various regression techniques, our goal is to create models that effectively capture the relationship between input parameters such as irradiation, temperature, and DC power, and the corresponding solar power outputs. This project will enable us to forecast the power output of the plants over the next few days, facilitating better planning and management of solar energy resources.

2) Motivation: India's green energy efforts focus on solar power, benefiting from abundant sunshine and technology. However, solar energy is daytime-limited, requiring substantial space and upkeep. Solar plant performance depends on sunlight, wind, and temperature, as well as panel efficiency. The goal is to understand how these factors interplay to predict solar energy outcomes.

II. Literature Review

"Solar Power Prediction Using Machine Learning" [1] By E. Subramanian et al.: This paper employs Support Vector Machines, Gradient Boosting, and Random Forest as machine learning algorithms to predict the generated solar power. It uses metrics such as *Area Under Curve (AUC)* and F1-score to evaluate the models.

"A Survey of Solar Power Forecasting Machine Learning Techniques" [2] By Debojyoti Chakraborty et al.: This piece of literature discusses, trains, and applies a variety of ensemble ML algorithms such as Bagging (Random Forest, Extra Trees), Boosting (AdaBoost, XGBoost, etc.), Voting, and stacking to the meteorological data from East India in order to forecast solar power generation. Their findings encouraged the use of stacking and voting ensemble machine learning methods for this purpose.

"Short-term Solar Power Forecasting Using XGBoost with Numerical Weather Prediction" [3] By Quoc-Thang Phan et al.: This review paper advocates for the use of Kernel Principal Component Analysis as the feature selection technique along with XGBOOST (Extreme Gradient Boosting) as the main regression technique for obtaining accurate prediction of 'one-hour ahead solar power forecasts.'

"Solar Energy Prediction Model Based on Artificial Neural Networks and Open Data" by Jose Manuel Barrera [4]: This research on solar energy prediction employs various neural network models, highlighting their impact on forecasting solar power output with improved accuracy. Leveraging open data sources, the study showcases the effectiveness of artificial neural networks in capturing complex patterns, thereby enhancing renewable energy forecasting techniques.

III. Description of Dataset

The datasets were obtained from a public contest on Kaggle. Each dataset belonged to a particular solar power plant located in India.

The dataset includes both solar panel data and weather sensor data for the two plants. The observations were recorded at 15-minute intervals over a period of 34 days. It's important to note that the data was collected at the inverter level, where each inverter is reported to contain multiple solar panels. This comprehensive dataset provides us with valuable insights into the performance of solar power plants and the factors influencing their energy generation.

A. Generation Data

Type 1: Generation Data

- `Date_and_time`: Timestamp for each observation. Observations were recorded at 15-minute intervals.
- `Plant_id`: Unique identifier for the specific solar power plant.
- `Source_key`: Identifier for the specific inverter (SOURCE_KEY).
- `DC_power`: Amount of DC power generated by the inverter (SOURCE_KEY) in this 15-minute interval. Units - kW.
- `AC_power`: Amount of AC power generated by the inverter (SOURCE_KEY) in this 15-minute interval. Units - kW.
- `Daily_yield`: Cumulative sum of power generated on that day until that point in time.

- `Total_yield`: Total yield for the inverter until that point in time.

B. Weather Sensor Data

Type 2: Weather Sensor Data

- `Date_and_time`: Timestamp for each observation. Observations were recorded at 15-minute intervals.
- `Plant_id`: Unique identifier for the specific solar power plant.
- `Source_key`: Identifier for the specific weather sensor (e.g., temperature sensor, irradiance sensor).
- `Ambient_temperature`: Ambient temperature at the plant.
- `Module_temperature`: Temperature reading for the solar panel module attached to the sensor panel.
- `Irradiation`: Amount of irradiation for the 15-minute interval.

IV. Proposed Architecture

A. Data Analysis

The dataset was initially spread across multiple files. The dataset of each plant was first compiled to form a single file. This was done by concatenating the two files on the basis of the common DATE_TIME column. Then we merged the dataset and plotted the correlation of AC_Power with the PLANT_ID. This plot, in fact, confirmed the fact that the PLANT_ID had little to no effect on the total AC_POWER that was being produced. We partitioned our dataset into training, validation, and test sets, allocating 75%, 15%, and 10% respectively. Each dataset was further split into input features (x_{train}, x_{val}, x_{test}) and output labels (y_{train}, y_{val}, y_{test}).

B. Feature Analysis and Selection

We conducted a thorough analysis of the relationships of the various parameters with AC power. The results are as follows:

- Inverter ID: No direct correlation observed.
- DC Power: Strong positive correlation, implying a direct relationship between DC power input and AC power output.
- Daily Yield and Total Yield: Indirect and complex relationships due to dependency on sunlight exposure and peak AC power generation.
- Ambient Temperature and Module Temperature: Indirect impact on AC power output, affecting panel and inverter efficiencies.
- Irradiation: Strong positive correlation with AC power output, indicating its importance in determining DC power generation.

C. Correlation Analysis

We validated our findings by plotting correlations between various parameters in the training dataset, confirming the observed relationships. The above findings were further supported by using Lasso Regression to determine the coefficients of each feature. Lasso stands for Least Absolute Shrinkage and Selection Operator and it plays a vital role in determining feature importance. We applied 5-fold cross-validation to determine the optimum value of λ (the penalizing factor) to be used as the learning rate. The reason behind choosing Lasso regression as a method for feature selection instead of Principal Component analysis was to be able to clearly identify the features that did not participate much in determining AC Power. Since PCA would have reduced the current dimensions by returning a linear combination of the original dimensions, making such a conclusion would have been difficult (See Figure 9).

D. Model Selection and Evaluation

We experimented with various regression algorithms, including Linear Regression, XGBoost, AdaBoost.R2, and Random Forest, to forecast solar power plant output. The datasets at hand were trained using multiple models.

1) Linear Regression
2) Random Forest Regression
3) ADABoost.R2 (Adaptive Boosting Regression)
4) XGBoost (Extreme Gradient Boosting)
5) Artificial Neural Networks (ANN)
    a) With Optimizer as Root Mean Squared Propagation (RMSprop)
    b) With Optimizer as Adaptive Moment Estimation (Adam)

Linear Regression in this scenario is the basic Least Squares Regression. Trivially, it failed to capture the complexities of the data due to its 'simple' nature. Subsequently, we explored ensemble methods such as AdaBoost.R2 and Random Forest. Adaptive Boosting Regression is an ensemble method in which the 'subsequent regressors focus on the more difficult areas' which previous iterations failed to explain. Random Forest is a more robust version of Bagging in which we pick $\sqrt{d}$ parameters at random from the input instead of the entire d dimensions (Bagging). These regression techniques performed much better than linear regression, with Random Forest performing slightly better than AdaBoost.

Upon further research, we discovered the machine learning model based on Extreme Gradient boosting and analyzed its performance over the others. XGBoost not only incorporated built-in cross-validation but also performed regularization. This algorithm was chosen over standard gradient boosting because it did not opt for a greedy approach while pruning trees. Instead, it made splits until the specified {max_depth} and then started pruning the trees backward.

XGBoost emerged as the top-performing model among the four models tested tilll now due to its advanced nature. Its superior performance on validation metrics, such as Mean Squared Error (MSE) and R^2 score, positioned it as the most effective choice.

Based on research and examples presented in [4], we decided our next model of choice to be Artificial Neural Network. For this purpose, we utilized the
tensorflow and *keras* library. We developed two ANN Models, incorporating three layers in each. We set the input dimension

at 4. We selected mean squared error as the loss function. For the activation function of the first two layers, we chose the ReLU function, and for the third layer, we applied a linear function.

$$\text{ReLU}(x) = \max(0, x)$$

The difference between the two ANN Models was because of the selected **optimizer.**

- First Model - Root Mean Squared Propagation as Optimizer: *RMSprop* adjusts the learning rate for each parameter by dividing it by the square root of the exponentially decaying average of squared gradients. This scaling operation reduces the learning rate for parameters with large gradients and increases it for parameters with small gradients.
- Second Model - Adaptive Moment Estimation (Adam) as Optimizer: Adam computes individual adaptive learning rates for different parameters from estimates of the first and second moments of the gradients. Adam and RMSprop were chosen over standard stochastic gradient descent because they are more robust to noisy data and tend to converge faster.

ANN-ADAM proved to be the best-performing model on the validation Set.

E. Evaluation Metrics:

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

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

MSE: Calculates the average squared difference between the actual values (y_i) and the predicted values ($\hat{y}_i$) for all samples in your data. Lower MSE indicates a better fit between the model and the data.

R2-Score: Represents the proportion of variance in the dependent variable (what you're trying to predict) explained by the independent variables (features used in your model). It ranges from 0 to 1, with a higher value indicating a better fit.

F. Final Model Testing

Evaluating the trained ANN-ADAM model on the test dataset yielded promising results, confirming its efficacy in accurately predicting solar power output.

By following this structured approach, we were able to develop an architecture for predicting AC power output in solar power plants, leveraging feature analysis, model selection, and validation techniques to ensure reliable and accurate predictions.

V. VISUALIZATIONS

We plot the plant-wise features on a scatter plot to determine their relationship with AC POWER. This allows us to theoretically determine the features which will impact the solar power output. In order to confirm the above analysis, we plot the correlation matrix using a heat map. Note that in each of the below visualizations, the plots on the left are for Plant - 1, and the plots on the right are for Plant - 2.

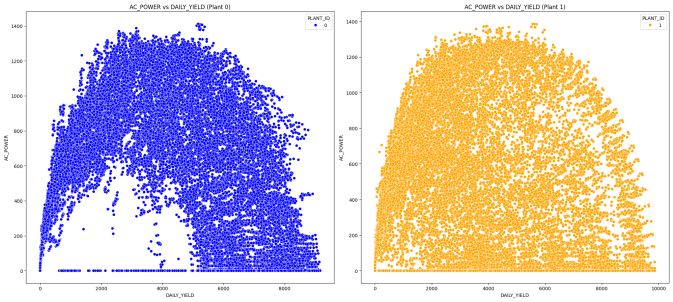

Fig. 1. AC Power Vs Daily Yield

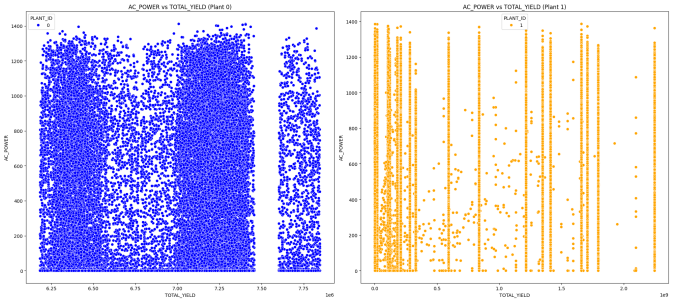

Fig. 2. AC Power Vs Total Yield

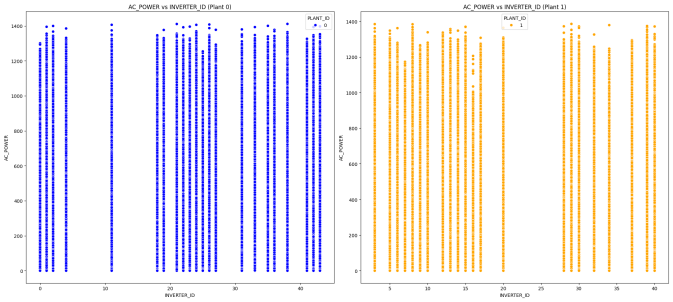

Fig. 3. AC Power Vs Inverter ID

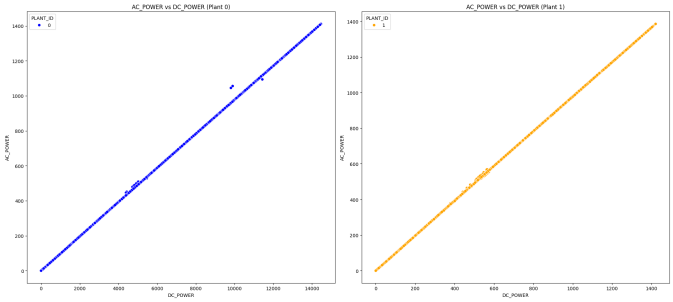

Fig. 4. AC Power Vs DC Power

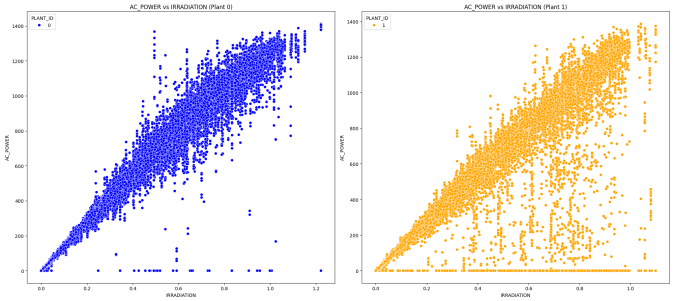

Fig. 5. AC Power Vs Irradiation

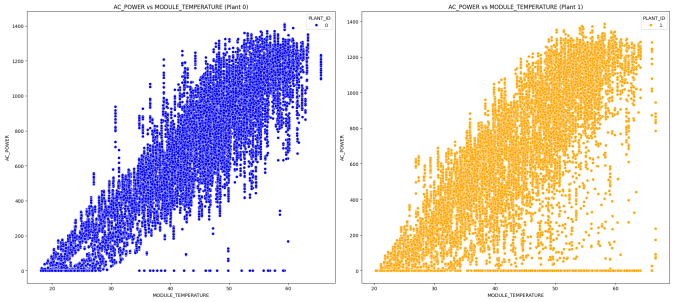

Fig. 6. AC Power Vs Module Temperature

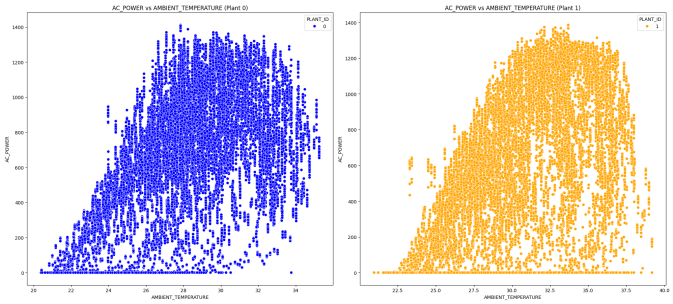

Fig. 7. AC Power Vs Ambient Temperature

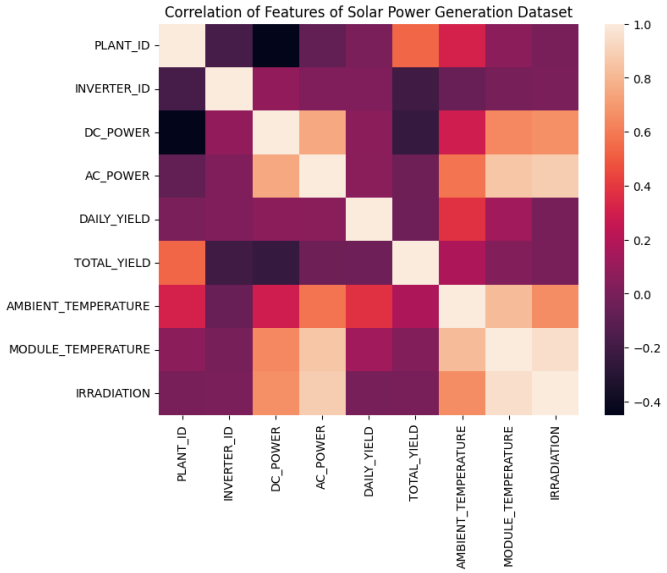

Fig. 8. Feature Correlation Heat Map

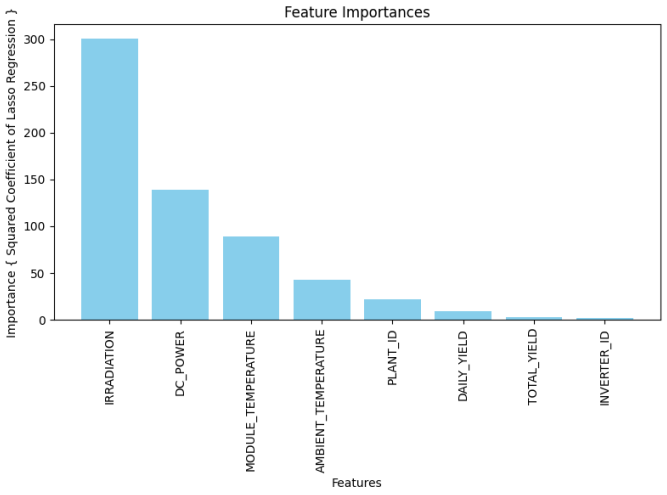

Fig. 9. Lasso Coefficients

From the above plots, we were able to drop the following features from any further consideration:

- `Daily_Yield`
- `Total_Yield`
- `Inverter_ID`

VI. Results

A. Analysis of Results

Given that R-squared (R2) measures the proportion of the variance in the dependent variable that is predictable from the independent variables, the fact that ANN-ADAM achieves the highest R2 score among the models indicates its superior accuracy in predicting AC power output in solar plants

TABLE I
MODEL PERFORMANCE

Model	MSE	R2-Score
Linear Regression	23702.03	0.8348
AdaBoost	9673.88	0.9326
Random Forest	1567.58	0.9891
XGBoost	97.26	0.9992
ANN-RMSprop	6.39	0.99996
ANN-Adam	3.59	0.99997

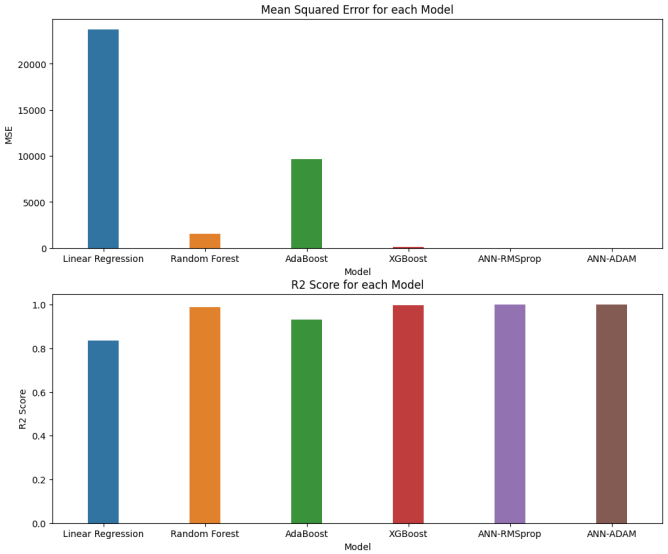

Fig. 10. Visualization of Model Performance

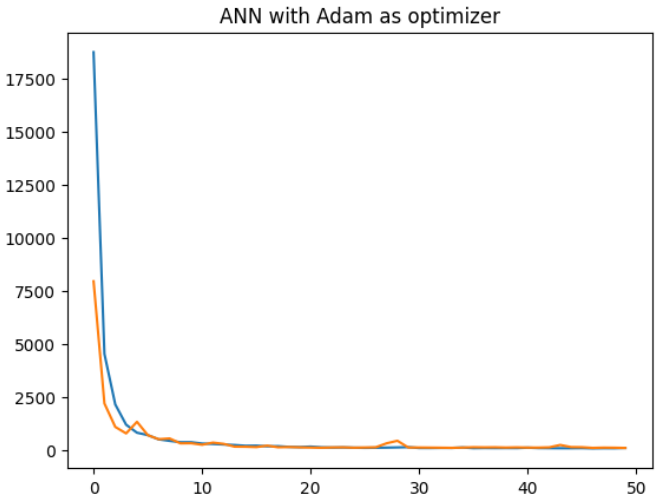

Fig. 11. ANN_ADAM: Number of Epochs Vs MSE

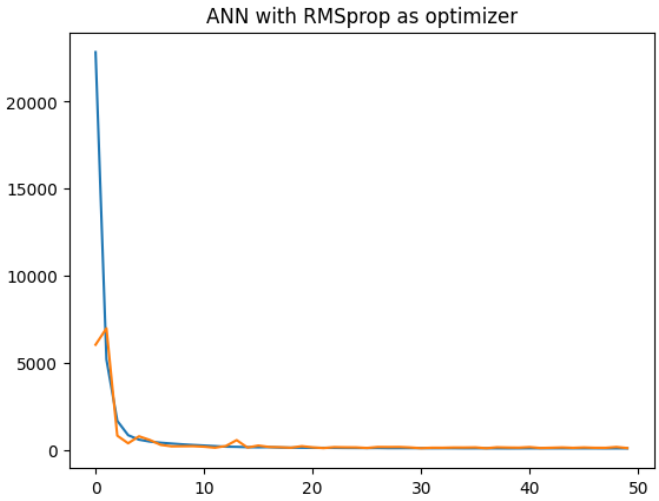

Fig. 12. ANN_RMSprop: Number of Epochs Vs MSE

The orange line-graph represents the validation set created by splitting the train set into further 20% and 80%.

TABLE II
ANN_ADAM PERFORMANCE ON TEST SET

Metric	Value
Test MSE	52.41
Test R2-Score	0.9996

VII. INFERENCES AND CONCLUSIONS

Several vital insights emerge in analyzing various machine learning algorithms for predicting AC power output in solar power plants. Firstly, XGBoost surpasses linear regression in capturing intricate non-linear correlations, making it more adept at representing complex patterns. While ensemble learning methods like AdaBoost and Random Forest lessen bias and variance, they need help in the case of noisy and high-dimensional datasets, which are typical of solar power prediction. XGBoost's focus on refining predictions through boosting strategies enables it to adapt better to dataset complexities. Additionally, XGBoost's incorporation of regularization techniques prevents overfitting and enhances model generalization, contrasting with Linear Regression's susceptibility to overfitting with high-dimensional datasets. Additionally, the remarkable performance of Artificial Neural Networks (ANNs) is due to their adeptness in handling non-linearity, feature representation, scalability, and regularization techniques. The ADAM optimizer is extremely robust since it adjusts the learning rate for each parameter individually and thus helps in early convergence. The ANN captures relationships within the data, allowing them to effectively model the complex dynamics of solar power generation.

A. Further Improvement

We performed a 5-fold cross-validation on the ANN-ADAM Model to determine the optimum number of epochs, after

which the MSE will reduce. We tested for epochs between 10 and 100 for this process. The results showed that the MSE in the case of 100 Epochs reduced significantly.

VIII. Individual Contributions

Both group members contributed equally to all aspects of the project. They collaborated closely on data analysis, preprocessing, model development, result analysis, documentation, and final presentation.

References

[1] Subramanian, E., Karthik, M. M., Krishna, G. P., Kumar, V. S., & Prasath, D. V. (2022). Solar Power Prediction Using Machine Learning. Department of Computer Science, Sri Shakthi Institute of Engineering and Technology, Coimbatore, India. Retrieved from https://arxiv.org/ftp/arxiv/papers/2303/2303.07875.pdf

[2] Chakraborty, D., Mondal, J., Barua, H. B., & Bhattacharjee, A. (2023). Computational solar energy – Ensemble learning methods for prediction of solar power generation based on meteorological parameters in Eastern India. Renewable Energy Focus, 44, 277-294. https://doi.org/10.1016/j.ref.2023.01.006

[3] Phan, Q.-T., Wu, Y.-K., & Phan, Q.-D. (2021). Short-term Solar Power Forecasting Using XGBoost with Numerical Weather Prediction. In 2021 IEEE International Future Energy Electronics Conference (IFEEC) (pp. 1-6). doi:10.1109/IFEEC53238.2021.9661874

[4] Barrera, J. M. (2020). Solar Energy Prediction Model Based on Artificial Neural Networks and Open Data. Sustainability, 12(17), 6915. Retrieved from https://www.mdpi.com/2071-1050/12/17/6915

[5] McGovern, A., Gagne, D. J., II, Basara, J., Hamill, T. M., & Margolin, D. (2015). Solar Energy Prediction: An International Contest to Initiate Interdisciplinary Research on Compelling Meteorological Problems. *Bulletin of the American Meteorological Society*, *96*(8), 1388-1395. https://doi.org/10.1175/BAMS-D-14-00006.1