# Machine Learning methods for Global Horizontal Irradiance prediction

Donato Francioso
University of Bari
d.francioso7@studenti.uniba.it

Francesco Didio
University of Bari
f.didio2@studenti.uniba.it

## Abstract

*Year after year, the production of Co2 in the world keeps increasing. To try to reduce Co2 emissions we could use the largest energy producer, that is the Sun. Solar energy is abundant, safe and it can be used for Photovoltaic Panels.*

*The focus of this study is to predict Global Horizontal Irradiance (GHI), that is the total amount of shortwave radiation received from above by the horizontal surface, the ground, using physical data like Temperature, Relative Humidity and Direct Normal Irradiance (DNI).*

*This study compared the results of various Machine Learning models such as Random Forest Regressor, Elastic Net, Linear Regression, XGBoost, K-Nearest Regressor and Artificial Neural Network.*

## 1. Introduction

Considering energy consumption, the use of renewable resources like photovoltaic models is the best choice, but it might represent a problem since the solar energy is highly unstable, due to the variations of weather conditions, and this issue can cause power instability and an increase of energy cost. [1] The devices used to measure the useful data can be expensive so it would be a good thing to predict such information. Machine learning models are tools that can help the decision process in many contexts and use cases. The aim of this work is to try to find a method to predict GHI, the principal value in the photovoltaic installation, using some physical data like Temperature, Relative Humidity and DNI. The cost of measurement would be reduced by having data in advance, taken with physical instruments. Moreover, there would be a value which is close to the real one.

## 2. Related works

The first phase for these research was dedicated to study of various metodology in GHI prediction. In all study that have been found the main target in the GHI prediction was to reduce the costs of manual prediction. Pratima Kumari and Durga Toshniwal of indian institute of technology Roorkee[3], used Elastic Net regression, Lasso and Ridge Regression, Multi layer perceptron Random forest and K-NN to predict GHI. Their work was focused on clustering of 21 Indian cities positioned in different climatic zones. Frank Vignola of Department of Physics in University of Oregon [4], study the correlation of cloudy and cloudless skies condition in GHI prediction. It found that is better to divide this two period to have a better results. In particular it says that in cloudy sky condition the errors are about five times as large. Ashis Kumar Mandal, Rikta Sen, Saptarsi Goswami and Basabi Chakraborty [5] have developed an alternative approach in GHI prediction. They used LSTM, because is the most powerful tool for modeling complex time series problems. In particular they have been used Univariate and Multivariate approach. In univariate approach they use only GHI as input for the prediction, instead in Multivariate they combined all the best correlated metereological variables (temperature, and humidity for the prediction). The study shows that the multivariate approach works better.

## 3. Materials

The dataset used in this work is the National Solar Radiation Database (NSRDB), that is a complete collection of hourly values of the solar irradiation components (DNI, DHI, GHI) and meteorological data using geostationary satellites. [2]

### 3.1. Data Collection

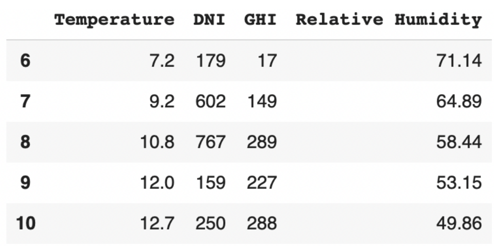
In this work, data of three years (2017-2019) were considered, with a temporal resolution of 1 hour useful to affirm the efficiency of selected machine learning models. The dataset of 3 cities, Bari, Torino and Roma located in different zones of Italy are used for training and testing the models.

### 3.2. Data Processing

To use the dataset, our data should be firstly cleaned and then transformed. This phase is the most important since, in case of regression, one must find which data work better for the predicted values.

First, the important features were selected and the other ones have been eliminated. The selected features are the most correlated with GHI, that are Temperature, Relative Humidity and DNI. Then the rows with GHI with 0 values are deleted, because in the night there is no solar irradiance and it is useless for the prediction. Then the Nan values are replaced with 0. Finally, the data was scaled to have a common scale with a robust scaler, that helps with the prediction. In particular it treats better the outliers.

In Figure 1 there is an example of our dataset after all the process.



| | Temperature | DNI | GHI | Relative Humidity |
|---|---|---|---|---|
| 6 | 7.2 | 179 | 17 | 71.14 |
| 7 | 9.2 | 602 | 149 | 64.89 |
| 8 | 10.8 | 767 | 289 | 58.44 |
| 9 | 12.0 | 159 | 227 | 53.15 |
| 10 | 12.7 | 250 | 288 | 49.86 |

Figure 1. Some rows of Bari dataset.

## 4. Methodology

For the prediction of GHI several methods were used.

### 4.1. Data split and hyperparameters

To fit the model, the data are split with the holdout method with 70% of data for the training phase and 30% of data for the test. For linear models like Linear Regression and Elastic Net the data were transformed into polynomial data with degrees equal to 3. This transformation help linear model to find a better correlation between data in case of multidimensional input. For tuning the hyperparameters, the GridSearch cross-validation, that combines all the possible parameters to find the best estimator using 10 folds, was used. For each methods is showed a table which contains the best hyperparameters for each city.

### 4.2. Linear Regression

Multidimensional Linear Regression is the simplest model for deriving a linear model to minimize the residual sum of squares between the observed targets in the dataset and the targets predicted by the linear approximation.

### 4.3. Elastic Net Regression

Elastic Net is a linear model with the introduction of penalties used to reduce the overfitting. The penalty is the combination of two traditional regularization penalties taken from Lasso and ridge regression. (aggiunta intro su lasso e ridge)

| | l1_ratio | alpha |
|---|---|---|
| Bari | 1 | 0.1 |
| Torino | 1 | 0.1 |
| Roma | 1 | 0.1 |

Table 1. Best Hyperparameters.

### 4.4. Random Forest Regressor

Random Forest Regressor is an ensemble method that combines many models (Decision tree) to provide a solution for complex problems. At each step, the tree produces a prediction and at the end it takes the mean of each tree as a result [3].

|  | max_depth | n_estimators |
|---|---|---|
| Bari | 6 | 140 |
| Torino | 6 | 120 |
| Roma | 6 | 100 |

Table 2. Best Hyperparameters.

### 4.5. XGBooster Regression

XGBoost is an efficient implementation of the gradient boosting algorithm. It is designed to be both computationally efficient and highly effective. Gradient boosting gives prediction models in the form of an ensemble of weak prediction models. In this case XGBooster uses decision trees.

|  | learning_rate | max_depth | n_estimators |
|---|---|---|---|
| Bari | 0.1 | 5 | 100 |
| Torino | 0.1 | 5 | 50 |
| Roma | 0.1 | 5 | 100 |

Table 3. Best Hyperparameters.

### 4.6. K-Neighbors Regressor

Generally used as a clustering method, it is also used to solve regression problems. The distance of each new test point to all training data point is calculated using several distances. [3]

|  | n_neighbors | metric |
|---|---|---|
| Bari | 9 | euclidean |
| Torino | 9 | manhattan |
| Roma | 9 | manhattan |

Table 4. Best Hyperparameters.

### 4.7. Artificial Neural Network

ANN can be used for supervised machine learning problems as well. Simple ANN architecture use sequential module from Keras. In table 5 is showed the architecture of used neural network and in table 6 the best hyperparameters for each city.

Adam optimizer was used with learning rate = 0.1 and Mean squared error as loss function.

| Level | Size | Parameters |
|---|---|---|
| Dense | 64 | input_dim=3,<br>initializer=”random_uniform”<br>activation = ”relu”,<br>regularizer = l1(0.1) |
| Dense | 32 | activation =”relu”,<br>regularizer = l1(0.1) |
| Dropout | 0.2 |  |
| Dense | 1 | activation=”linear” |

Table 5. Architecture.

|  | batch_size | epochs |
|---|---|---|
| Bari | 24 | 40 |
| Torino | 32 | 40 |
| Roma | 24 | 30 |

Table 6. Best Hyperparameters.

### 4.8. Metrics

To evaluate the models several metrics were used. In particular:

- **Root Mean-Squared-Error**: Root Mean-Square-Error (RMSE) is the square root of Mean-Squared-Error (MSE). MSE is the average of the square of the error. An error represents how close it is to a regression line. MSE is measured in units that are the square of the target variable, while RMSE is measured in the same units as the target variable.

$$RMSE = \sqrt{\sum \frac{(\widehat{y_i} - y_i)^2}{n}} \qquad (1)$$

- **R2**: Represents the quantity of variance of predicted value and real value. In particular, 100% means that the predicted value is perfectly correlated to the real value.

$$R^2 = 1 - \frac{\sum(\widehat{y_i} - y_i)^2}{\sum(y_i - \widetilde{y_i})^2} \qquad (2)$$

- **Mean-Absolute-Error**: Very similar to MSE, but in this case errors are calculated using absolute difference between the predicted value and real value.

$$MAE = \frac{1}{n} \sum |y_i - \widehat{y_i}| \qquad (3)$$

## 5. Results

In figure 2 are showed graphically the RMSE for each city. In table 7, 8 and 9 are showed the numeric results for all methods for each city.

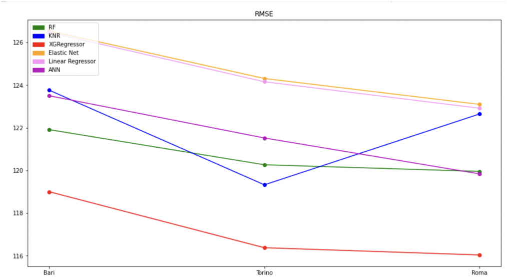


Figure 2. Some rows of Bari dataset.

| | RMSE | R2 | MAE |
|---|---|---|---|
| Linear | 126.47 | 0.805 | 94.46 |
| Elastic Net | 126.54 | 0.805 | 94.78 |
| XGBooster | 118.99 | 0.825 | 87.18 |
| Random Forest | 121.90 | 0.817 | 90.50 |
| KNR | 123.75 | 0.811 | 91.81 |
| ANN | 123.49 | 0.812 | 92.93 |

Table 7. Results for Bari.

| | RMSE | R2 | MAE |
|---|---|---|---|
| Linear | 124.15 | 0.805 | 93.44 |
| Elastic Net | 124.30 | 0.805 | 93.74 |
| XGBooster | 116.37 | 0.836 | 84.36 |
| Random Forest | 120.26 | 0.825 | 87.85 |
| KNR | 119.32 | 0.828 | 87.04 |
| ANN | 121.51 | 0.821 | 90.32 |

Table 8. Results for Torino.

| | RMSE | R2 | MAE |
|---|---|---|---|
| Linear | 122.91 | 0.814 | 91.52 |
| Elastic Net | 123.09 | 0.813 | 91.93 |
| XGBooster | 116.03 | 0.830 | 82.69 |
| Random Forest | 119.94 | 0.818 | 86.80 |
| KNR | 122.64 | 0.810 | 87.57 |
| ANN | 119.83 | 0.819 | 87.85 |

Table 9. Results for Roma.

### 5.1. Conclusion

Following the results in the previous tables it can be said:

- DNI, temperature and relative humidity are good features for predict the GHI.
- XGBooster is the best model for each city.
- A simple model like linear regression is not the best but it work good too.
- Random forest, KNR and neural network works similar, some city have better perform with one of this model than other.

In conclusion machine learning model can be used for GHI prediction if there is a small budget or if we need to predict this measue in advance.

## 6. Bibliography

## References

[1] Diagne, M., David, M., Lauret, P., Boland, J., and Schmutz, N. *Review of solar irradiance forecasting methods and a proposition for small-scale insular grids.* Renewable and Sustainable Energy Reviews 2013; 27: 65-76.

[2] Manajit Senguptaa, Yu Xiea, Anthony Lopezb, Aron Habtea, Galen Maclaurinb, James Shelbyc *The National Solar Radiation Data Base (NSRDB).* Renewable and Sutainable Energy Reviews 89 (2018) 51-60.

[3] Pratima Kumari , Durga Toshniwal *Machine learning techniques for hourly global horizontal irradiance prediction: A case study for smart cities of India.* International Conference on Applied Energy 2021

[4] Frank Vignola, Department of Physics Univeristy of Oregom, *Ghi correlations with Dhi and Dni and the effects of Cloudiness on one-minute Data.*

[5] Ashis Kumar Mandal, Rikta Sen, Saptarsi Goswami and Basabi Chakraborty, *Comparative Study of Univariate and Multivariate Long Short-Term Memory for Very Short-Term Forecasting of Global Horizontal Irradiance.*

[6] Pedro HT and Coimbra CF *Nearest-neighbor methodology for prediction of intra-hour global horizontal and direct normal irradiances.* Renewable energy 2015;80:770–782.