# **List of Tables**

**Chapter 3**

Table 1. Hardware Tools Table

Table 2. Software Tools Table

**Chapter 8**

Table 3: Initial Model Results for Windmill Power Production

Table 4. Optimized Model Results for Windmill Power Production

Table 5. Initial Model Results for Solar Power Production

Table 6. Optimized Model Results for Solar Power Production

Table 7. Model Performance on Hybrid Power Generation System

# **List of Figures**

**Chapter 3**

Fig 1. Software block diagram

**Chapter 6**

Fig 2. Distribution of Numerical Features

Fig 3. Triangle Heatmap of Feature Correlations

Fig 4. Top 10 Most Important Features Based on Correlation

Fig 5. Panel Area vs Generated Power

Fig 6. Irradiance vs Generated Power

Fig 7. Panel efficiency vs Generated Power

Fig 8. Model Training Performance

Fig 9. Univariate Analysis

Fig 10. Correlation Matrix

Fig 11. Top 10 Most Important Features Based on Correlation

Fig 12. Air Density vs Generated Power

Fig 13. Turbine Area vs Generated Power

Fig 14. Turbine Efficiency vs Generated Power

Fig 15. Wind Speed vs Generated Power

Fig 16. Model Performance

Fig 17. Cross-Validation Results

Fig 18. Average Production by Season

Fig 19. Annual Energy Production

Fig 20. Average Monthly Production for Each Source

Fig 21. Average Production by Day of the Week

Fig 22. Average Production by Hour of the Day

Fig 23. Average Hourly Energy Production (2020-2023)

Fig 24. Distribution of Produced Energy

**Chapter 8**

Fig 25. Random Forest CM

Fig 26. Linear Regression CM

Fig 27. Hyperparameter Tuning Results (Solar)

Fig 28. Hyperparameter Tuning Results (Wind)

Fig 29. Hyperparameter Tuning Results (Hybrid)

## **Optimization and Simulation of a Hybrid Solar and Wind-Powered Industrial Data Center**

# **Chapter 1**

Keep Blank

# **Chapter 2**

**2.1 Introduction**

As worldwide data processing and storage are rapidly growing, industry data centers are becoming the main infrastructure hubs. Nonetheless, these facilities require an enormous amount of energy, which is causing valid concerns about the environment and the consequently costly operations. [1] To mitigate these problems, the inclusion of the sun and the wind as two possible renewable energy sources has quickly become a realistic solution for data centers in their quest for sustainability and the utilization of power in a more efficient way. [2, 3] This particular study will delve into the modeling and simulation of an industrial data center that runs on both solar and wind energy to render it more advanced technology-wise and lessen its environmental impact.

When arguing about the use of solar-powered as opposed to wind-powered data centers, many factors are important, with each one contributing to the efficiency and sustainability of these energy sources. Solar energy, because it depends on photovoltaic panels, provides a predictable and relatively stable output of energy during the daylight hours. The conversion can be used to convert the unpredictable sunlight into electricity. [4] It can be an added advantage for data centers, which need a continuous flow of power so they run. Also, solar panels, by and large, present less of a problem with a space deficit in comparison with wind turbines, hence they are a possible solution for populated areas where space is limited. [5] However, solar panel efficiency is often directly related to geographic location and weather conditions, and therefore, it can produce insufficient or irregular amounts of energy in places with less sunlight, possibly necessitating the use of other power sources or large energy storage solutions.

It is the wind energy that captures the wind as the wind turbines generate electricity throughout the day and night, provided enough wind is available. [6] This quality can be very useful in places with strong and regular wind storms. However, wind turbines occupy a much larger area and may be visually and acoustically disturbing at times, thus it might be a hard one to apply in densely populated areas. The variation in wind speed, where it has both high and low periods, can lead to changes in energy output, thus needing robust energy storage systems or auxiliary power arrangements to guarantee a continuous power supply. Having said this, horizontal-axis wind turbines are mostly placed in suitable areas where the wind is favorable and thus the capacity factor is high. [7]

Primarily, the decision to choose solar and wind power for data centers depends on the surrounding geographic and manmade factors. Solar energy works well in predictable and sunny sites and is more efficient in populated areas with restricted space. On the other hand, Wind is more suitable for areas with high wind speeds. It can pose problems in those regions with space constraints and noise control. Nevertheless, in the long term, both of them that by the way of greening data center they can sustain the earth. But choosing the right options and identifying and dealing with the drawbacks of each source are the elements that need to be considered in a comprehensive way. [8]

The main objective of this work is to analyze the solar PV and wind turbine technologies and determine their interdependence in powering data centers. We plan to make energy demand analysis by making an in-depth study of the energy load behavior of the center so that the required capacity and the best combination of solar and wind power can be introduced or employed. We have designed the system by making a hybrid system that combines both solar and wind sources, which should come coupled with storage solutions that can not only absorb but also supply a stable and reliable power system. Finally, we will optimize the research through the use of advanced optimization tactics to heighten the productivity of the hybrid system. These tactics focus mainly on the choice of configuration and the right operational settings for both energy sources. [9]

We will be using different machine learning models to analyze the data from all the three types of energy sources: solar, wind and hybrid power stations. We will work out the best models for analyzing and break-down the best infrastructure to optimize the data centers’ power generation method.

2.2 Keep Blank

2.3 Keep Blank

2.4 Keep Blank

Develop Project Plan and Timeline & Gantt Chart: Keep Blank

# **Chapter 3**

**3.1 Resource List**

Table 1. Hardware Tools Table

|  |  |  |  |
| --- | --- | --- | --- |
| Component | Unit Price (in USD) | Quantity | Total Cost (in USD) |
| INTEL 13TH GEN CORE I5 13500 RAPTOR LAKE PROCESSOR 14 CORE 20 THREAD 2.5 GHZ ~ 4.8 GHZ 24 MB CACHE INTEL UHD 770 | 239.13 | 1 | 239.13 |
| MSI MAG B760M MORTAR DDR5 LGA1700 GAMING MOTHERBOARD | 183.62 | 1 | 183.62 |
| TEAM T-FORCE DELTA TUF GAMING RGB 16 GB 6000 MHZ DDR5 GAMING RAM | 67.47 | 2 | 134.94 |
| MAXGREEN 1200 VA OFFLINE UPS (PLASTIC BODY) | 52.10 | 1 | 52.10 |
| GIGABYTE C301 GLASS E-ATX GAMING CASE (BLACK) | 57.22 | 1 | 57.22 |
| MSI GEFORCE RTX 4060 TI VENTUS 3X 16 G OC GDDR6 GRAPHICS CARD | 555.13 | 1 | 555.13 |
| SAMSUNG 980 PRO 500 GB PCIE GEN4 M.2 NVME SSD | 73.45 | 1 | 73.45 |
| COOLERMASTER MWE 750-WATT GOLD V2 POWER SUPPLY | 999.23 | 1 | 999.23 |
| DEEPCOOL LE520 240 MM ALL-IN-ONE ARGB LIQUID CPU COOLER | 56.37 | 1 | 56.37 |
|  |  | **Total** | 2351.19 |

Table 1 represents the hardware components that one would focus on in a machine learning project are mainly those in obtaining stable performance and improving efficiency. The central processing unit, or CPU, does all the computations and tasks that machine learning algorithms have to do—in other words, getting data more suitable for training and the model itself. A gaming motherboard has all the necessary connections and is powerful enough to support high-performance components. All of this makes for a stable project of the CPU, RAM, and all peripherals. Gaming RAMs become key components in treating large data sets and complex computations by allowing fast data acquisition and storage necessary for model training and inference. UPS stands for uninterruptible power supply. Basically, this is a device for continuous power supply, normally connected to a computer to protect it from surges or failures that may damage data or hardware. The gaming case is where the components are put in and ordered. From this, smooth air circulation and cooling are possible to prevent overheating. The GPU provides much faster parallel computations needed while training the machine learning model, especially deep learning algorithms, and increases the speed manifold. Faster read and write speeds which are enabled by solid-state drives are primarily used in efficient data management and faster application running by removing the need for loading times and increasing system responsiveness. The power supply provides enough stable power to all the hardware parts allowing no overvoltage to occur. [10]

Table 2. Software Tools Table

|  |  |  |
| --- | --- | --- |
| **Tool** | **Functions** | **Why selected this tool** |
| Dataset | Collection of attributes and records for training and testing | Project relevance and authenticity to the target data centers |
| Pandas | Understanding dataset characteristics and informing preprocessing steps | Provides essential insights and visualizations for data understanding |
| Matplotlib, Seaborn | Builds static, and interactive visualizations | Provides essential visualizations for data analysis |
| Random Forest | Combines multiple decision trees | Improves predictive accuracy and controls overfitting |
| Linear Regression | Models the relationship between one or more independent variables and a dependent variable | Fits a linear equation to the observed data |
| Support Vector Regression | Models the relationship between one or more independent variables and a dependent variable | Finds a function that maximizes the margin of error while fitting the data within a specified tolerance |
| Gradient Boosting | Builds models sequentially, each correcting its predecessors’ errors | Improves predictive performance and reduce overfitting |
| K-Nearest Neighbors | Classifies data points based on the majority label of their K-nearest neighbors in the feature space | Predicts the value of a target variable by averaging the values of the K-nearest neighbors to a given data point |
| CatBoost Regressor | Builds models sequentially, each correcting its predecessors’ errors | Handles categorical features efficiently and robustly, and improves predictive accuracy |
| AdaBoost Classifier | Combines multiple weak classifiers to create a strong classifier | Focuses on the errors of previous models, and gives more weight to misclassified instances |
| Logistic Regression | Models the relationship between one or more independent variables and a dependent variable | Predicts the probability of a binary outcome |
| GridSearchCV Optimizer | Optimization of the model’s learning process | Efficient and adaptive learning rate |
| Mean Squared Error (MSE) | Measures the average squared difference between predicted and actual values | Quantifies the model’s prediction accuracy |
| Mean Absolute Error (MAE) | Measures the average absolute difference between predicted and actual values | Provides a straightforward indication of prediction accuracy |
| R2 Score | Quantifies the proportion of variance in the dependent variable that is predictable from the independent variables | Indicates the goodness of fit of a regression model |

**3.2 Specifications of Software Block Diagram using ML**

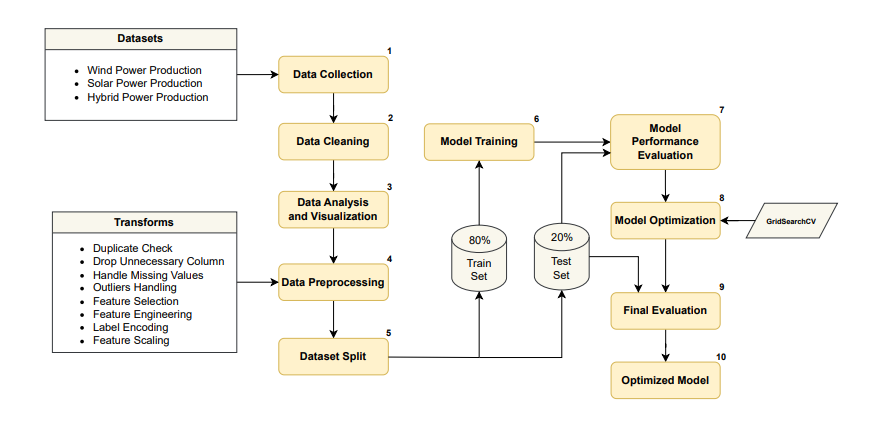


Fig 1. Software block diagram

Firstly, we have collected datasets from several sources which contained the data for solar powered stations, wind powered stations and hybrid stations. Secondly, we cleaned the datasets using different cleaning methods such removing duplicates, converting data types, filtering outliers, etc. Thirdly, we have analyzed the data and visualized them using different python libraries to make the data more readable. Fourthly, we have preprocessed the datasets using eight processing techniques. In the fifth section, we have split the data into two sets – train set containing 80% of the data and test set containing 20% of the data. In the sixth section, we have trained the machine learning models using the training set. In the seventh section, we have used the test set to evaluate the trained models’ performances. In the eight sections, we have optimized the models using the GridSearchCV algorithm. In the ninth section, the optimized models are evaluated for their performance using the test set again. Finally, in the tenth section, the optimized models with the best evaluative scores are finalized.

# **Chapter 5**

**5.1 Data Collection**

For this project, we have used three different datasets that contain the data about the performance and usage of both solar and wind powered generators, two common renewable energy sources.

The first dataset [11] that we have used, contained the data for the power generated by solar panels from a **Canadian power plant**, has the records of 4213 and 21 attributes namely:

* temperature\_2\_m\_above\_gnd: Represents the temperature 2m above the ground.
* relative\_humidity\_2\_m\_above\_gnd: Represents the relative humidity 2m above the ground.
* mean\_sea\_level\_pressure\_MSL: Represents the mean sea level pressure.
* total\_precipitation\_sfc: Represents the total precipitation.
* snowfall\_amount\_sfc: Represents the snowfall amount.
* total\_cloud\_cover\_sfc: Represents the total cloud coverage.
* high\_cloud\_cover\_high\_cld\_lay: Represents the high cloud coverage.
* medium\_cloud\_cover\_mid\_cld\_lay: Represents the medium cloud coverage.
* low\_cloud\_cover\_low\_cld\_lay: Represents the low cloud coverage.
* shortwave\_radiation\_backwards\_sfc: Represents the shortwave radiation backwards.
* wind\_speed\_10\_m\_above\_gnd: Represents the wind speed 10 m above the ground.
* wind\_direction\_10\_m\_above\_gnd: Represents the wind direction 10 m above the ground.
* wind\_speed\_80\_m\_above\_gnd: Represents the wind speed 80 m above the ground.
* wind\_direction\_80\_m\_above\_gnd: Represents the wind direction 80 m above the ground.
* wind\_speed\_900\_mb: Represents the wind speed at 900 mb.
* wind\_direction\_900\_mb: Represents the wind direction at 900 mb.
* wind\_gust\_10\_m\_above\_gnd: Represents the wind gust 10 m above the ground.
* angle\_of\_incidence: Represents the angle of incidence.
* Zenith: Represents the zenith.
* Azimuth: Represents the azimuth.
* generated\_power\_kw: Represents the power generated in kw.

The second group of datasets [12] containing the data for the power generated by windmills in **wind power stations in USA**, has records of 28200 for training and 12086 for testing. They have the following attributes:

* tracking\_id: Represents a unique identification number of a windmill.
* datetime: Represents the date and time of a record.
* wind\_speed(m/s): Represents the speed of wind (in meter per second).
* atmospheric\_temperature(Â°C): Represents the temperature (in degree Celsius) of a town or village that the windmill is present.
* shaft\_temperature(Â°C): Represents the temperature of the shaft (in degree Celsius).
* blades\_angle(Â°): Represents the angle of the blades of a wind turbine (in degrees).
* gearbox\_temperature(Â°C): Represents the temperature of a gearbox (in degree Celsius).
* engine\_temperature(Â°C): Represents the temperature of an engine (in degree Celsius).
* motor\_torque(N-m): Represents the torque of a motor (in Newton meter).
* generator\_temperature(Â°C): Represents the temperature of a generator (in degree Celsius).
* atmospheric\_pressure(Pascal): Represents the atmospheric pressure(in Pascals) in that area.
* area\_temperature(Â°C): Represents the temperature (in degree Celsius) of the area within a 100m radius of the windmill
* windmill\_body\_temperature(Â°C): Represents the temperature (in degree Celsius) of the body of a windmill.
* wind\_direction(Â°): Represents the direction of the wind (in degrees).
* resistance(ohm): Represents the resistance against the wind.
* rotor\_torque(N-m): Represents the torque of a rotator (in Newton meter).
* turbine\_status: Represents the status of the turbine (masked).
* cloud\_level: Represents the following levels of the cloud in the sky on a particular day: Extremely Low, Low, Medium.
* blade\_length(m): Represents the length of the blades of a windmill (in meter).
* blade\_breadth(m): Represents the breadth of the blades of a windmill (in meter).
* windmill\_height(m): Represents the height of the blades of a windmill (in meter).
* windmill\_generated\_power(kW/h): Represents the power generated (in Kilo Watt per hour).

The final dataset [13] contains the data of solar and wind energy production records on an hourly basis for the **French grid** since 2020. The dataset is well organized and it can be used in various fields of machine learning such as time series forecasting, anomaly detection, price signal analysis, etc. The dataset consists of 59806 records and 9 attributes namely:

* Date and Hour: Date and Hour range.
* Date: Day date granularity.
* StartHour: Recording Start Hour.
* EndHour: Recording End Hour.
* Source: Production Source (Wind or Solar).
* Production: Production (in MWh).
* dayOfYear: Day of the Year.
* dayName: Day Name.
* monthName: Month Name.

**5.2 Data Preprocessing**

We have used the following data preprocessing techniques to setup the dataset before training and testing the data with machine learning models. [14]

**Duplicate Rows Check:** Duplicate rows check is a key component of the data quality process: it involves finding and eliminating duplicate records that, if they are not removed, can cause the analysis or modeling to go wrong. It is possible to apply data duplication processes in traditional databases and flat files that are based on the basic SQL knowledge of comparison operators. Dignity of life is one of the main TU Delft's values, as is shown by numerous reasons: from keeping people's privacy in online and real-life teaching to ensuring values such as inclusion and respect. Once duplicates are identified, the obvious thing to do is try to find out the reason for the duplications; they could be due to common or rare errors. Consequently, having gone through the stages of detection and root cause identification, the next course of action after cleaning the datasets is the visualization and the distribution of the data to the main stakeholders.

**Dropping Unnecessary Columns:** The first thing that should be done in the process of dropping unnecessary columns is to evaluate the data and determine the existence of columns that are not and should not be in the dataset. Towards the goal of the column elimination, one can also use various filtering techniques such as correlation analysis to rank order the columns from which the least to the most important is to be removed. A complete list of stakeholders would include employees, customers, suppliers, partners, and the community. If everyone in a large company is considered a stakeholder, the list may be quite long but quite inclusive. Notwithstanding the fact that all employees could be labeled as stakeholders, they would still not feel equally important if kept at the same level and would probably become fierce competitors. What is suggested as the first step is for the data to be analyzed and the columns that cause troubles to be discovered and removed.

**Handle Missing or Null Values:** One of the most important things to do when working with datasets is to deal properly with missing or null values. If missing values were the cause of the issue, the wrong analysis or model would be produced. It all starts with the missing values detection, that is, finding the places where the values are absent, which can be done in different ways or with various tools. Thus, the first task is locating the missing values and making the required corrections. Different techniques can be uttered to deal with missing values, for example, imputing them with mean, median, or mode values, removing rows or columns with excessive missing data, or using algorithms that can handle missing values.

**Outliers Management:** Outliers Management is basically the practice of getting rid of outliers that heavily skew the results of statistical analysis, be it machine learning models or other analyses. An essential step in dealing with these outliers is detecting them through statistical methods, or even graphically by using visualizations. Evaluating the effect of these observations on the dataset is important to know the nature of their influence. From this we can decide to either transform, remove or cap outliers based on their effect on the data, or analysis.

**Feature Selection:** Feature Selection is an approach to identify the most important features, thus the process of accepting and excluding irrelevant variables for an analysis or model. This step is fundamental because this technique does not allow any occurring of "overfitting" a model and as a result, it raises its performance. It is a process that includes the examination of the significance of each feature by applying statistical tests, correlation analysis, and using model-based methods. When choosing a new feature, it is desirable to keep the ones that are highly correlated with the target variable or those that are the most fruitful for model simplification and reject the rest.

**Feature Engineering:** Feature Engineering is the process of creating new features or modifying existing ones to accommodate the improved performance of analysis or machine learning models. This step requires the derivation of new features by breaking down existing ones, thus the examples involved are probably calculating the ratio of a size or joining with other characteristics in a meaningful way. Additionally, it implies the conduction of operations on these characteristics like the logarithmic or polynomial transformation and the conversion of categorical data to numerical if it is needed. Feature engineering excellently done could give a great insight into the analysis and enhance the accuracy of the models.

**Label Encoding:** The process named Label Encoding is done with the help of the machine to transform the descriptions of the categories into source codes, which become suitable for machine learning. It means that we are to turn categorical data, which may be found in textual form, into a numerical representation where each category will be represented by a unique integer. Such a step is needed because many machine learning algorithms only accept numeric values as input.

**Feature Scaling:** Feature scaling is the method of adjusting feature values such as standardization or normalization to be in the same scale. This is important because different scales of characteristics can lead to changes in the performance and stability of machine learning algorithms. The scaling operation includes the selection of a method like normalization (scaling values to a specific range) or standardization (scaling values to have a mean of 0 and a standard deviation of 1). Of course, the right fit should be applied to all the available scaling methods to ensure that all features play an even role in the model. It will make the model work closer to a performance level of 100% as well as provide you with more reliable results.

# **Chapter 6: Design and Development**

**6.1 Solar Power Generation Model Development**

While developing the models for solar power generation we have used several python-based libraries to visualize the relationships between the independent and target variables. We have also implemented the formula: **Psolar = irradiance × panel area × panel efficiency**, in our calculations. A collection of figures describing the different relationships are listed below:

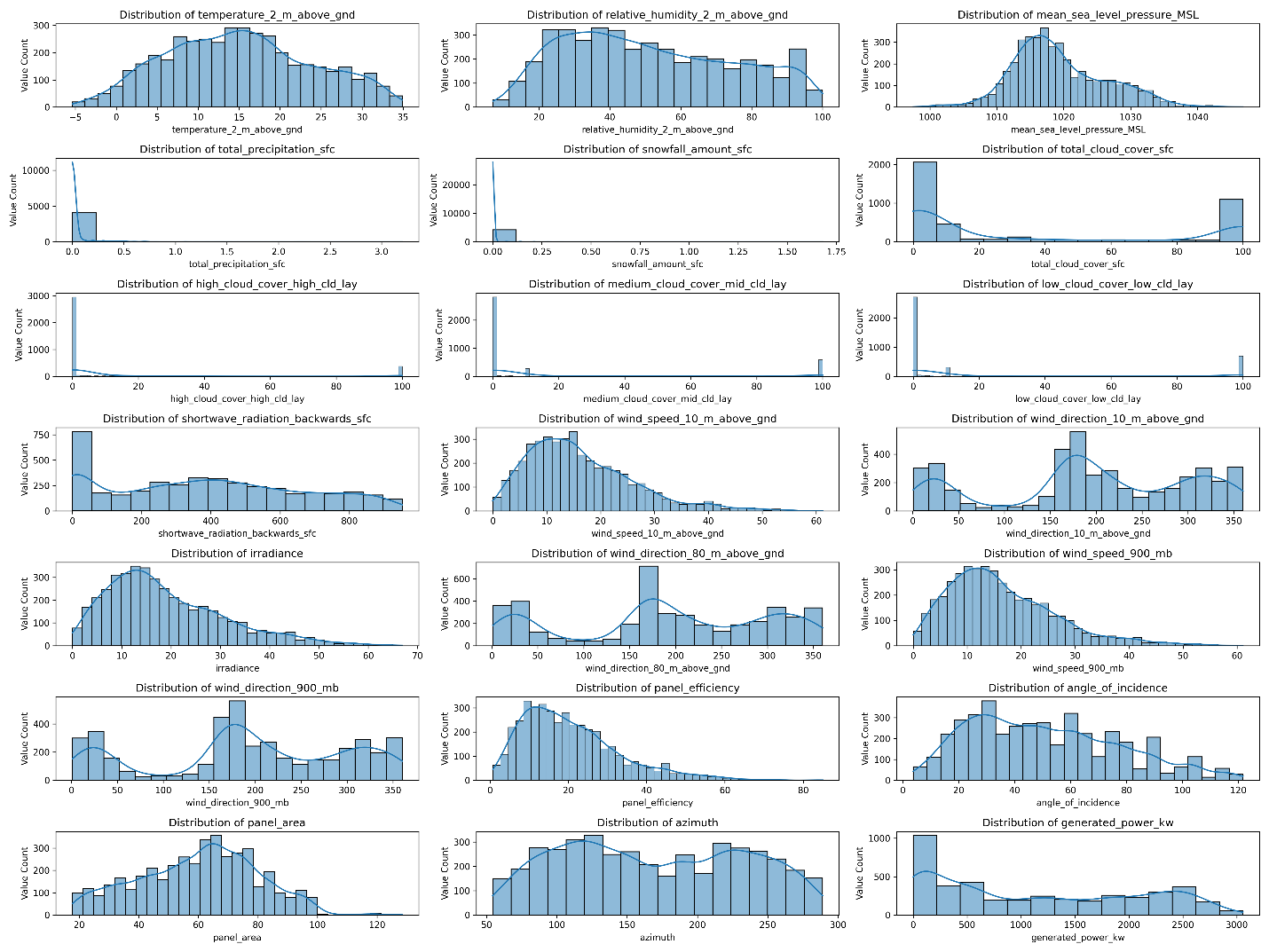


Fig 2. Distribution of Numerical Features

Figure 2 shows how the numerical features are distributed in their respective ranges and how the graphs are skewed, indicating the spreading or clustering of data.

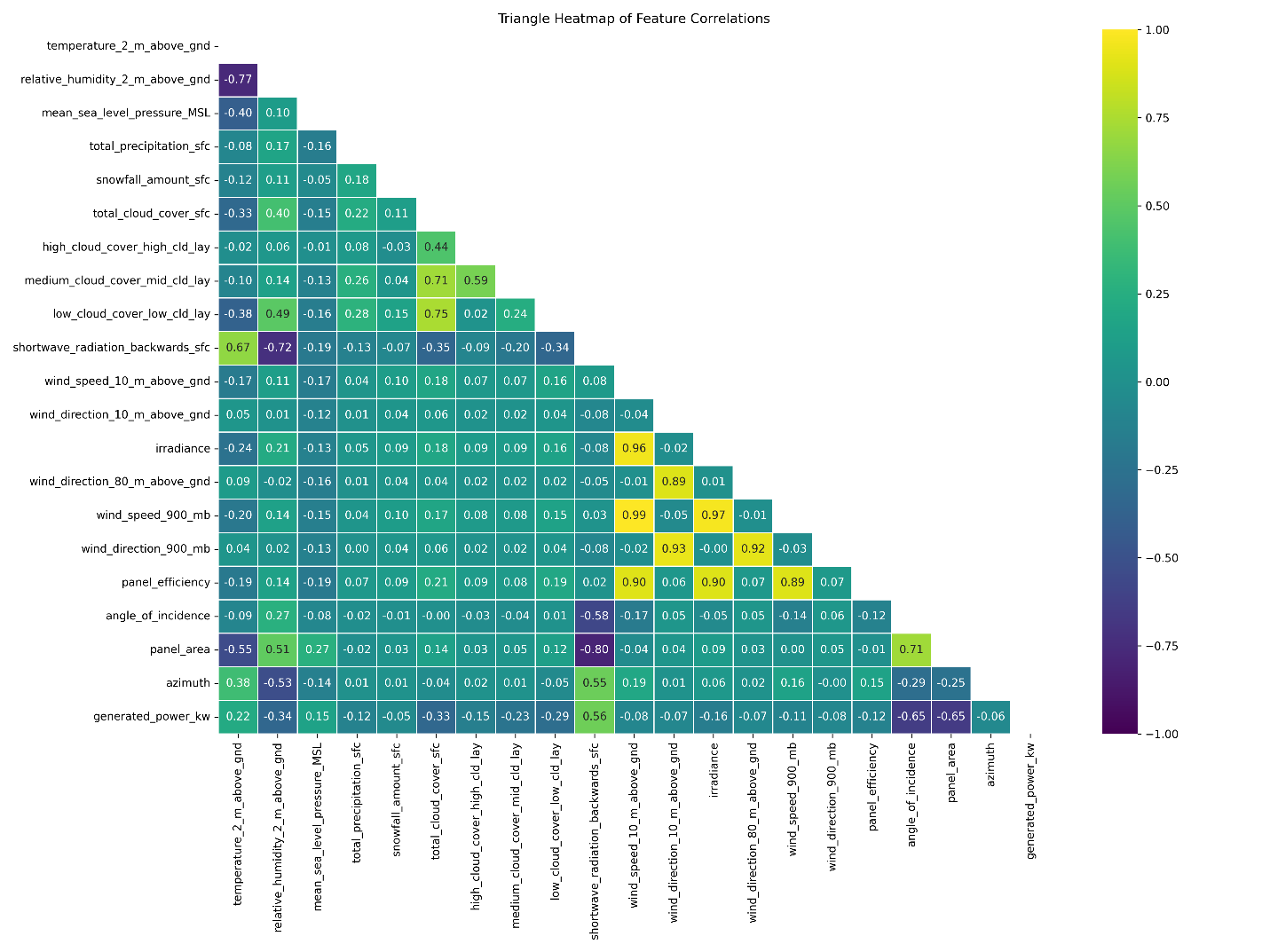


Fig 3. Triangle Heatmap of Feature Correlations

Figure 3 shows the heatmap of the feature correlations i.e. how strong the relation is between each feature. In this case, the darker the color is the more negatively correlated the features are and the lighter the color is the more positively correlated the features are.

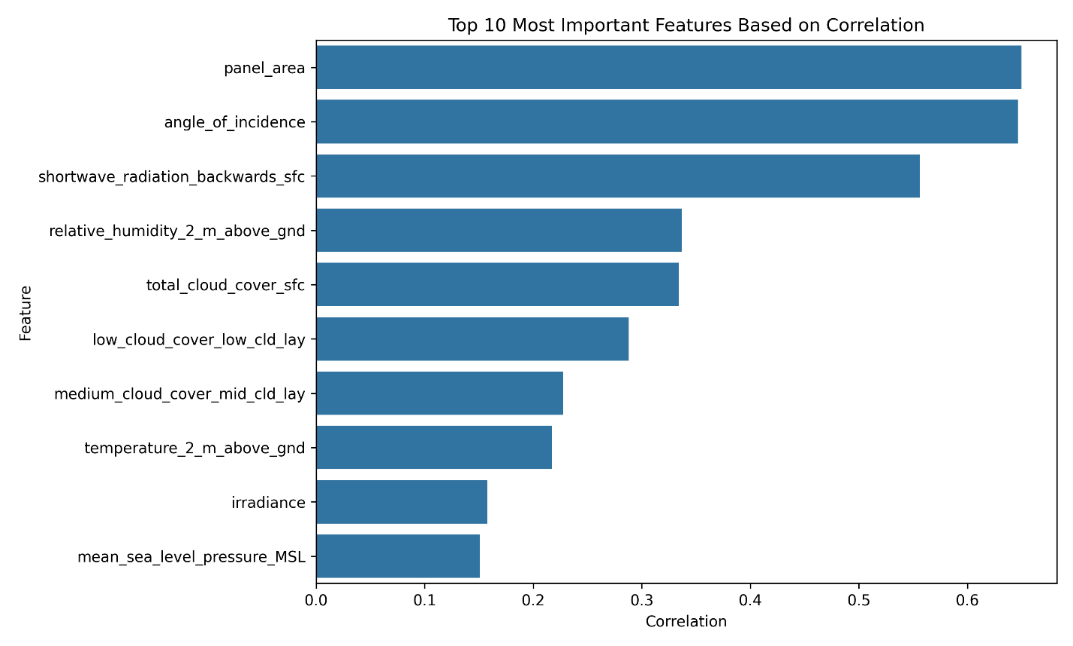


Fig 4. Top 10 Most Important Features Based on Correlation

Figure 4 represents the top 10 most important features based on correlation between the features and the target variable. It is found that ‘panel\_area’ had the highest correlation, followed by ‘angle\_of\_incidence’ and ‘shortwave\_radiation\_backwards\_sfc’.

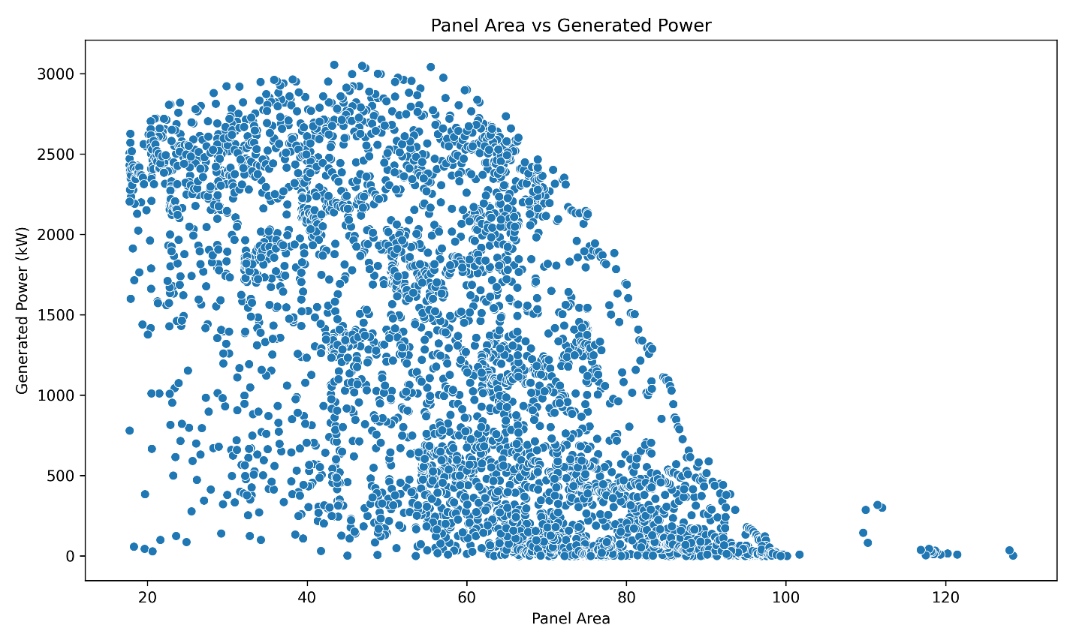


Fig 5. Panel Area vs Generated Power

Figure 5 shows the graph for panel area against generated power that indicates how much power is generated over a range of panel areas. It can be seen that panel area between 20 and 80 had generated the most amount of power and an approximation of 50 panel area had generated the highest amount of power that is, 3000 kw.

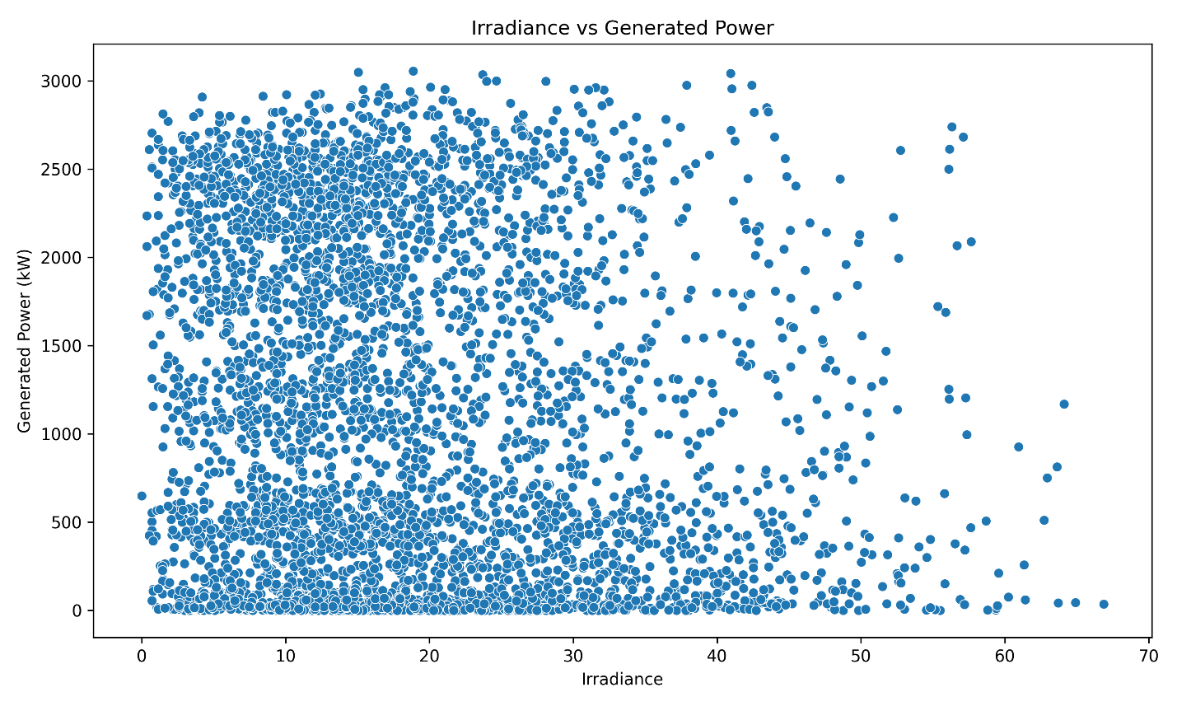


Fig 6. Irradiance vs Generated Power

Figure 6 illustrates the graph for irradiance against generated power that represents the change in power generation with respective to change in irradiance. It can be seen that irradiance within the range of 0 to 40 had the densest amount of power generation and an approximate irradiance of 20 had the highest power generation of 3000 kW.

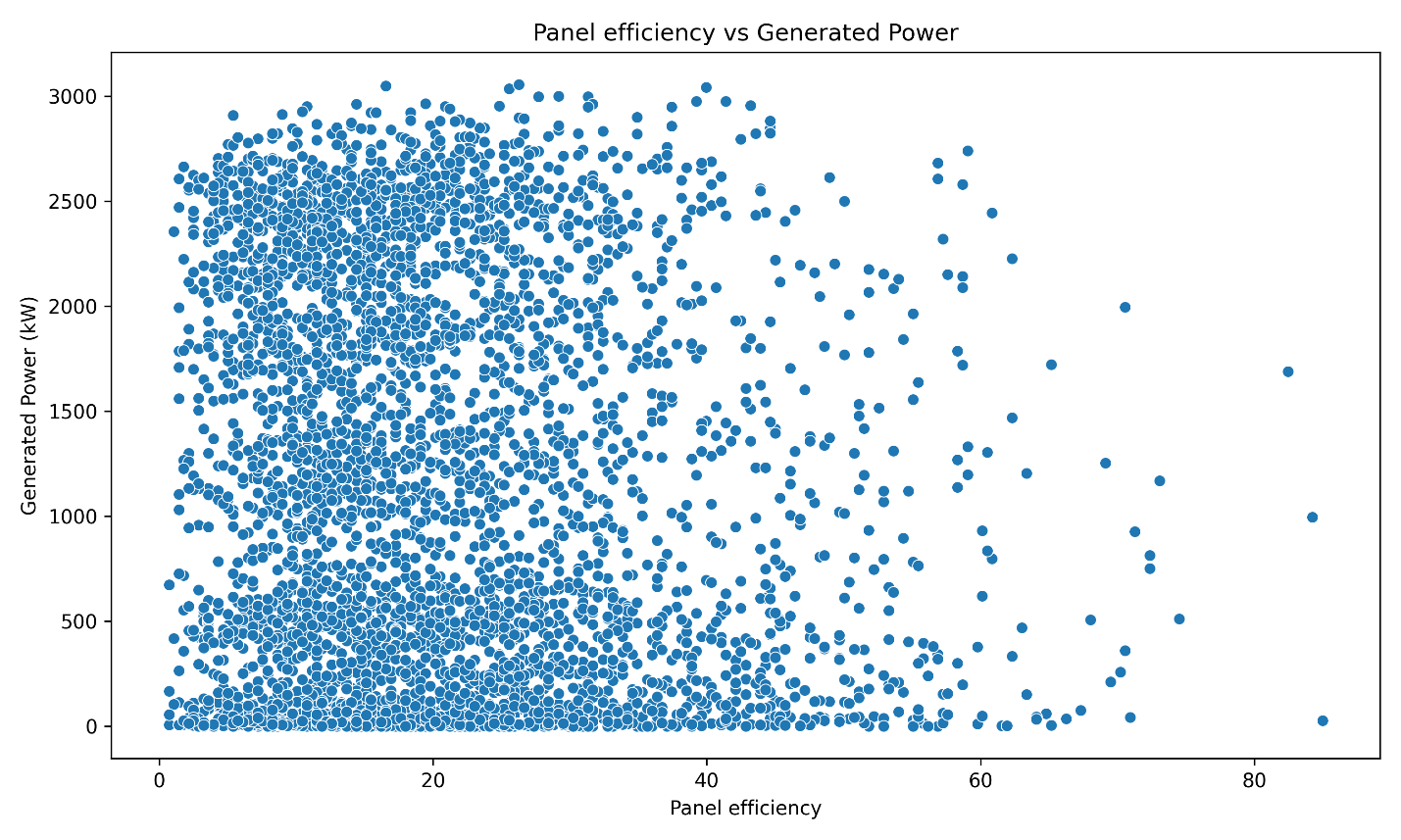


Fig 7. Panel efficiency vs Generated Power

Figure 7 represents the scatter diagram of panel efficiency against generated power that depicts how much power is generated for each change in panel efficiency. It can be seen that the graph is skewed to the left, meaning more variation of power generation occurred within the range of 0 to 40 panel efficiency, where an approximation of 20 panel efficiency generated the highest amount of power of 3000 kW.

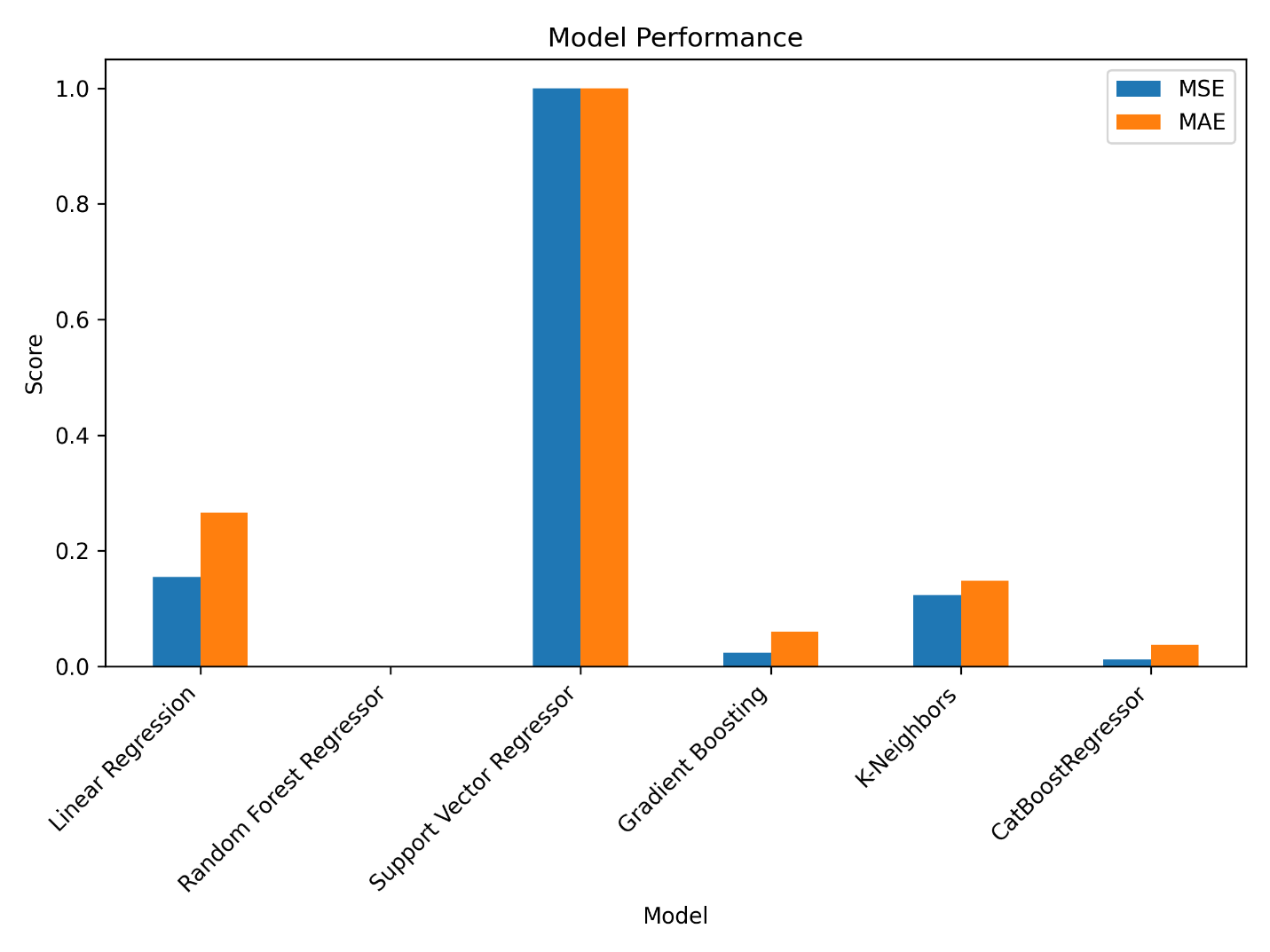


Fig 8. Model Training Performance

Figure 8 represents the training performances of every model we have used for the solar power generating plant. It can be seen that SVC had the highest MSE and MAE, meaning it had the most errors. In contrast, Random Forest Regressor had the lowest MSE and MAE, indicating that it is the best model to use.

**6.2 Wind Power Generation Model Development**

While developing the models for wind power generation we have used several python-based libraries to visualize the relationships between the independent and target variables. We have also implemented the formula: **Pwind = 0.5 × air density × turbine area × (wind speed3) × turbine efficiency**, in our calculations. A collection of figures describing the different relationships are listed below:

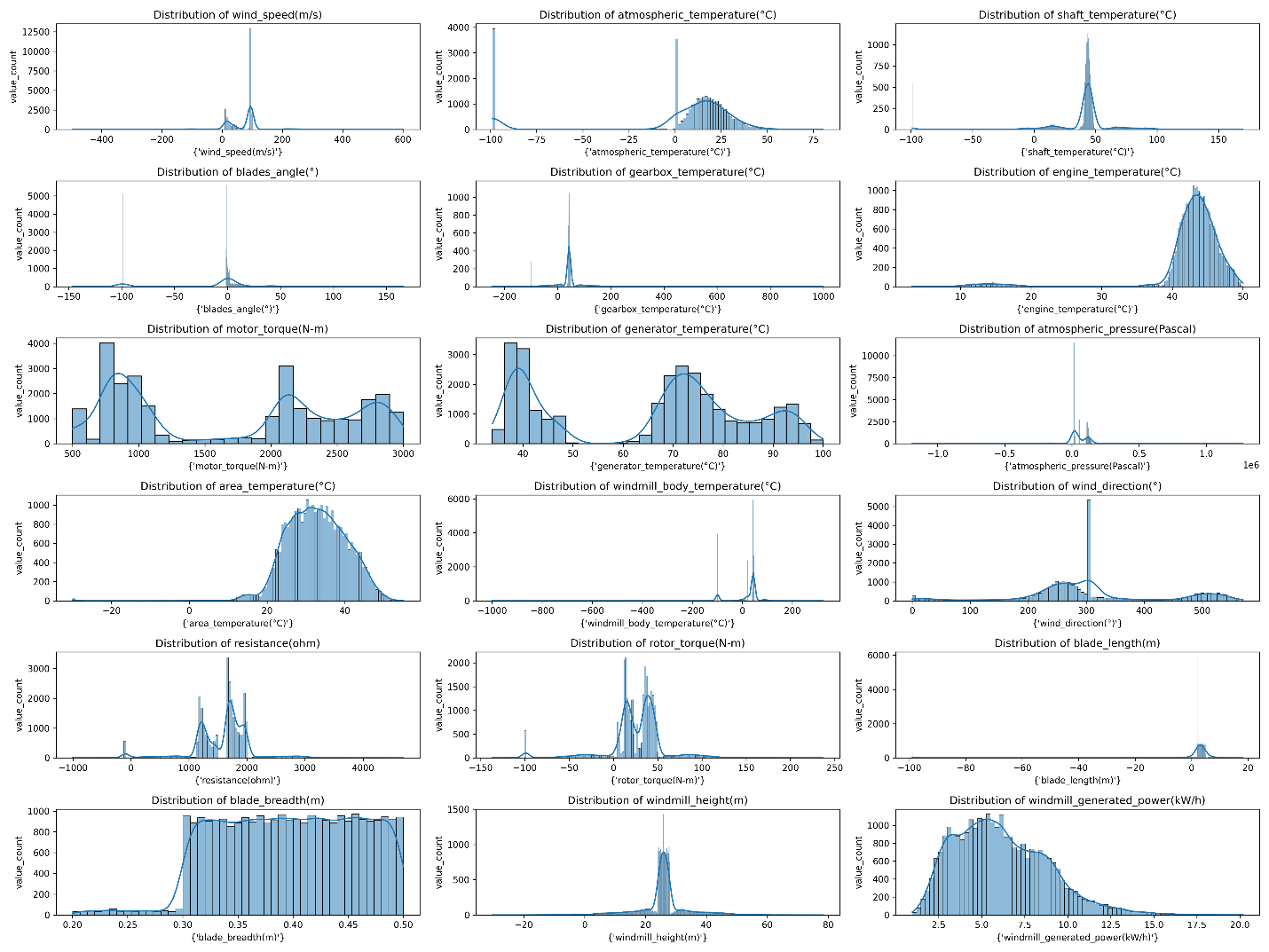


Fig 9. Univariate Analysis

Figure 9 represents the univariate analysis that indicates how each feature is distributed and how much the graph is skewed determining how the data is clustered.

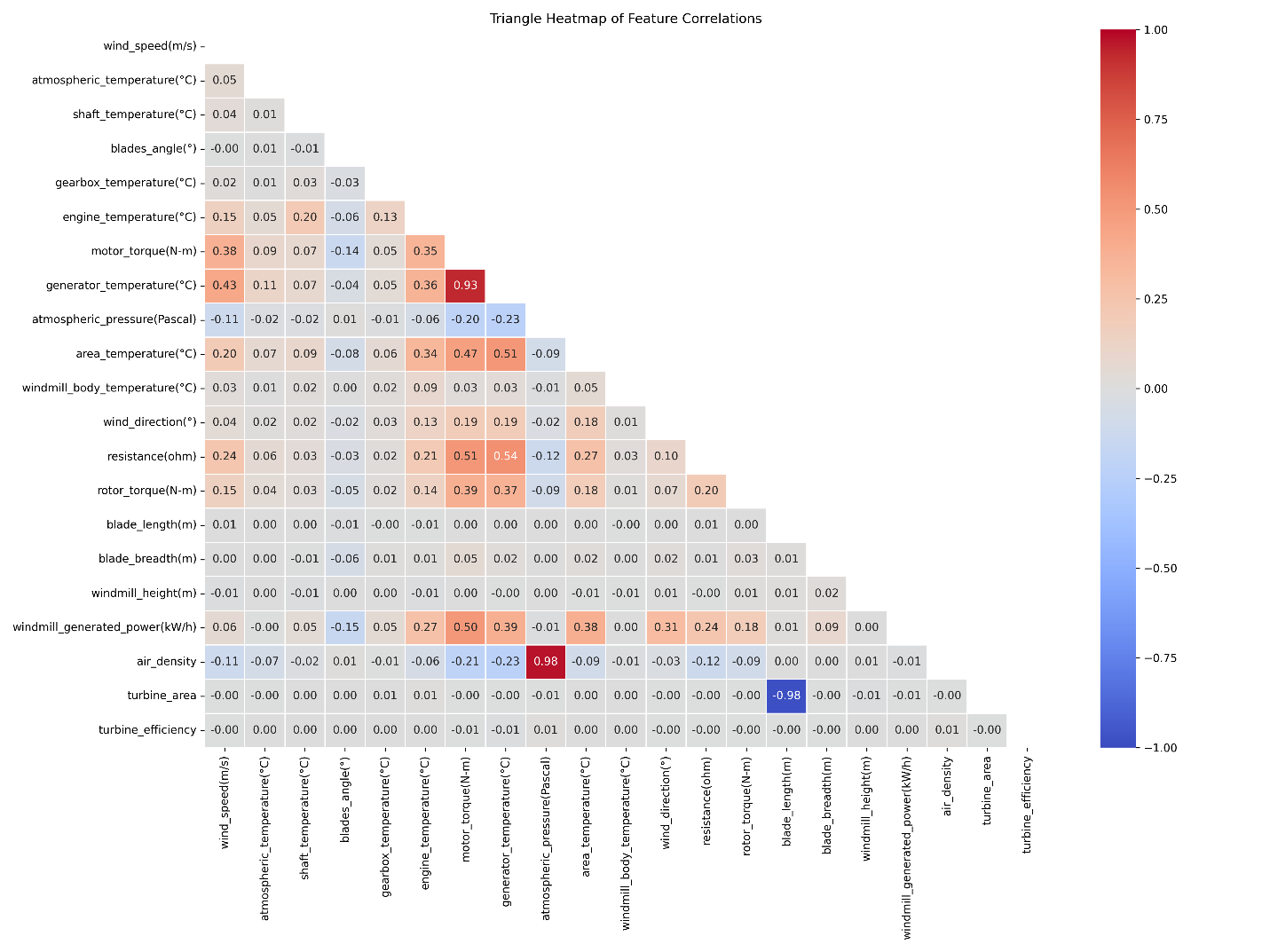


Fig 10. Correlation Matrix

Figure 10 represents the correlation matrix for the power generation by the wind. In this case, the bluer the color is, there is more negatively correlation and the redder the color is, there is more positively correlation between the features.

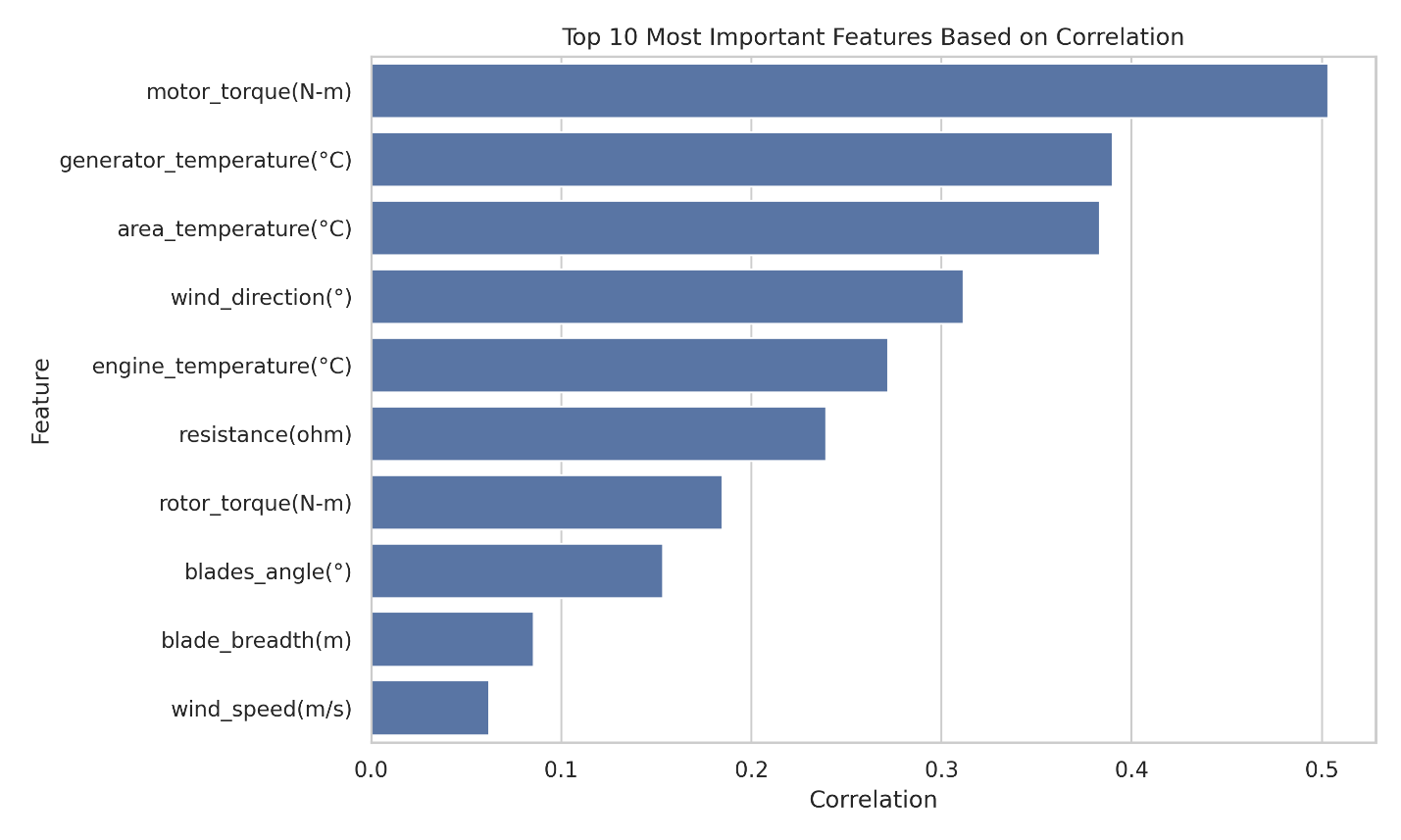


Fig 11. Top 10 Most Important Features Based on Correlation

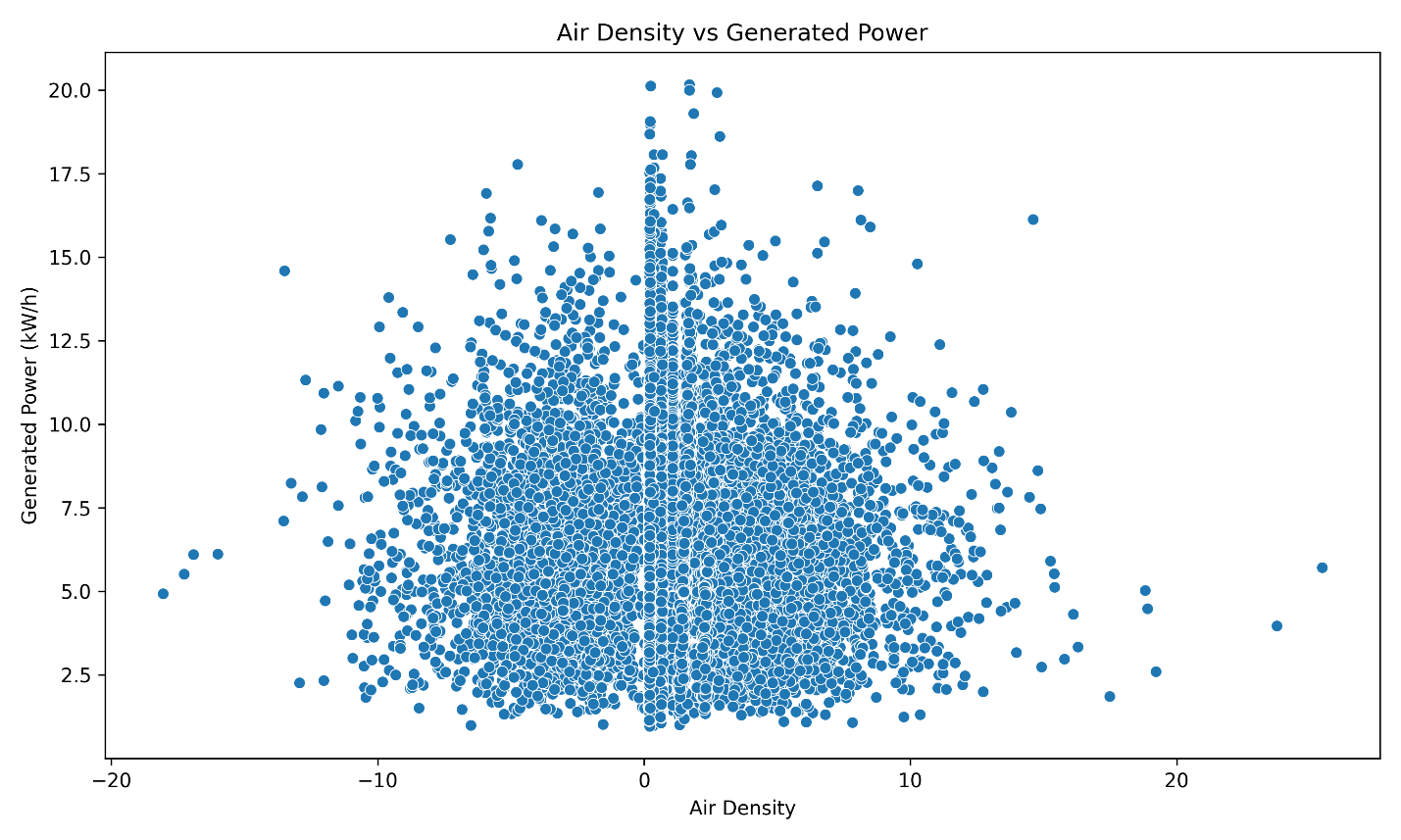


Fig 12. Air Density vs Generated Power

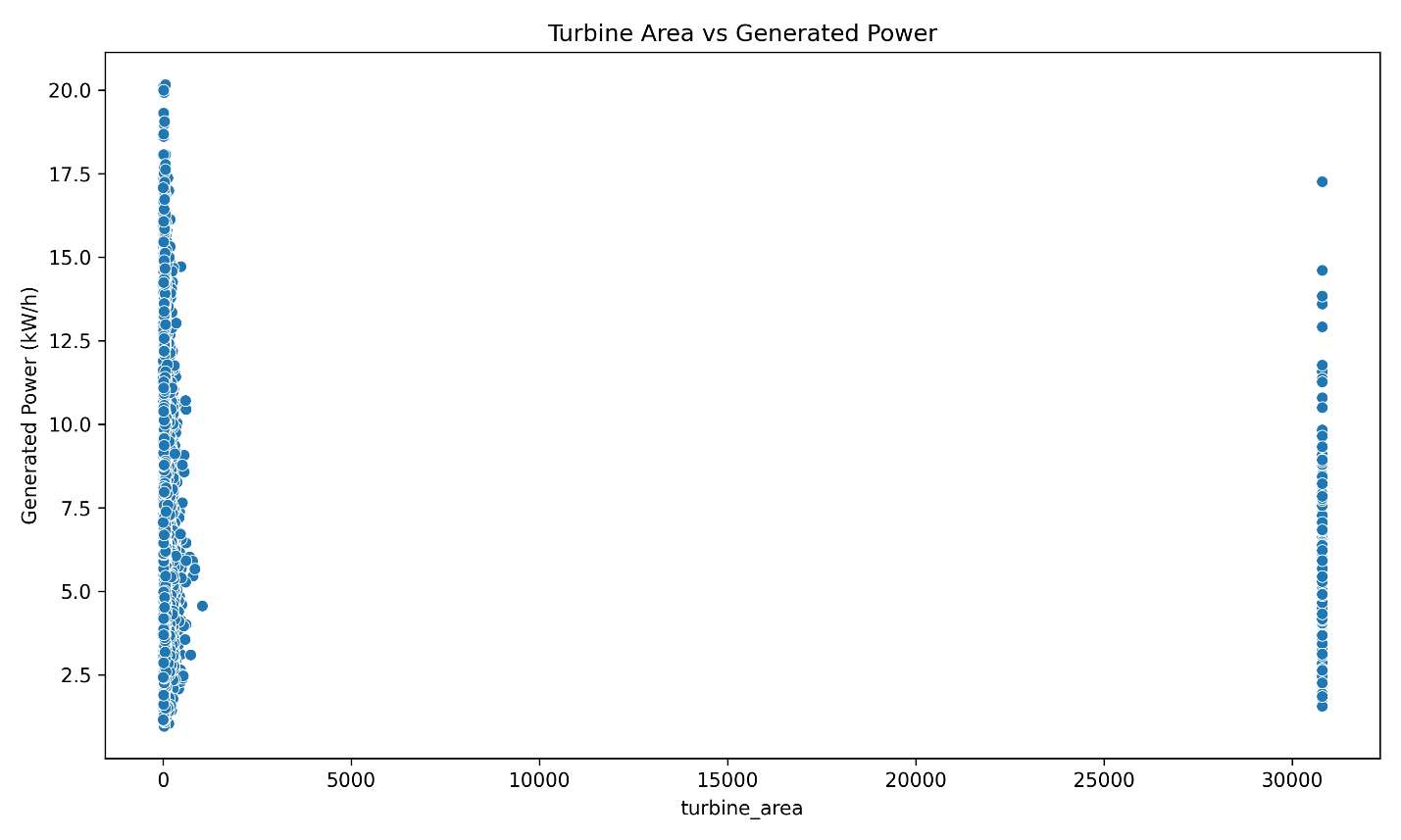


Fig 13. Turbine Area vs Generated Power

Figure 11 represents the top 10 most important features based on correlation, indicating that ‘motor\_torque(N-m)’ had the highest correlation followed by ‘generator\_temperature(°C)’ and ‘area\_temperature(°C)’.

Figure 12 represents the graph for air density against power generation, where it can be seen that the data is clustered in the center. Most of the power is generated when the air density was within the range of -10 to 10 and the maximum power of 20 kW/h was generated when the air density was 0.

Figure 13 represents the graph for turbine area against power generation which indicates that approximately 0 to 1000 area of turbine generated the most power and only a certain amount of power was generated for turbine area of 31000. The highest amount of power, 21.0 kW/h, was generated at turbine area of 0.

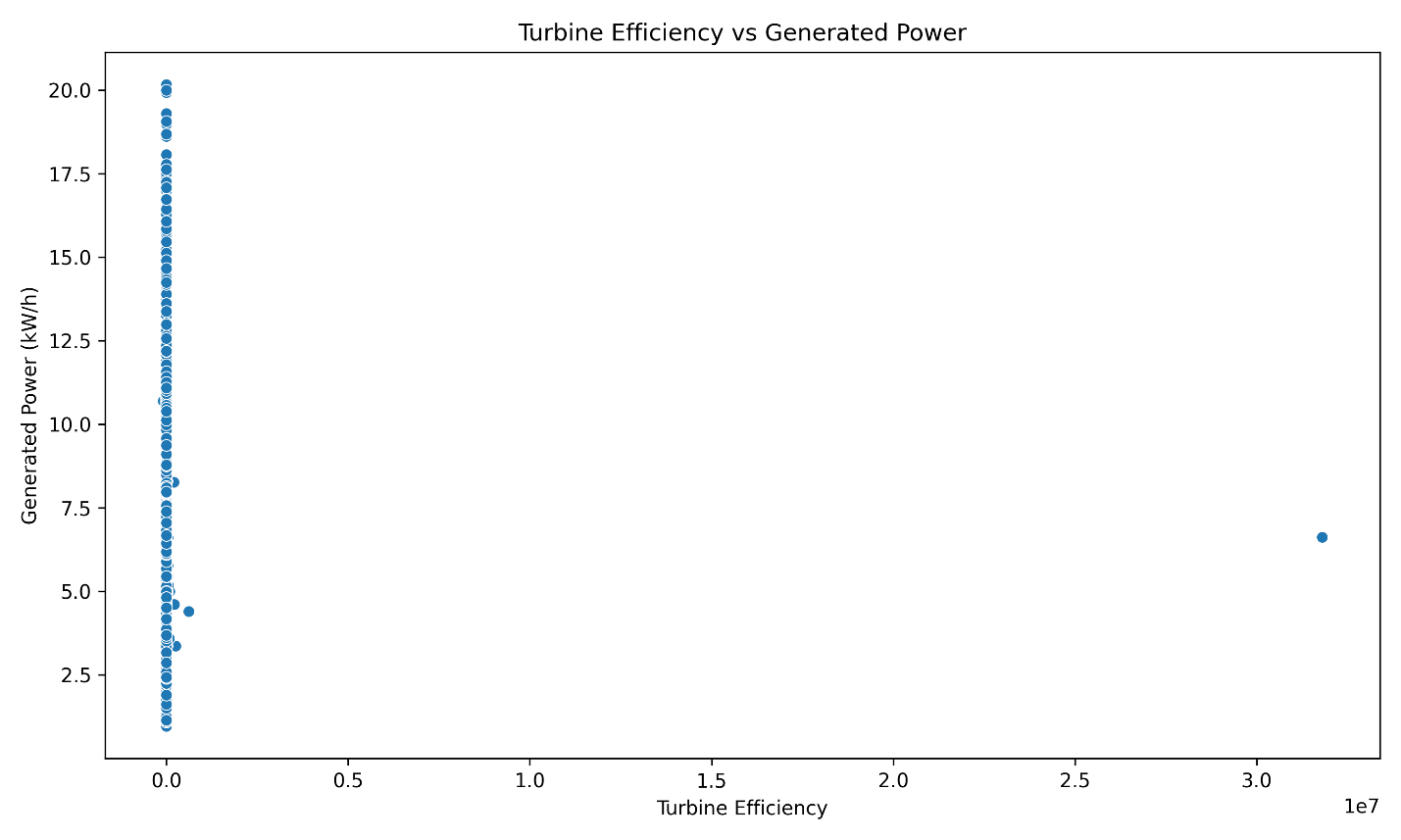


Fig 14. Turbine Efficiency vs Generated Power

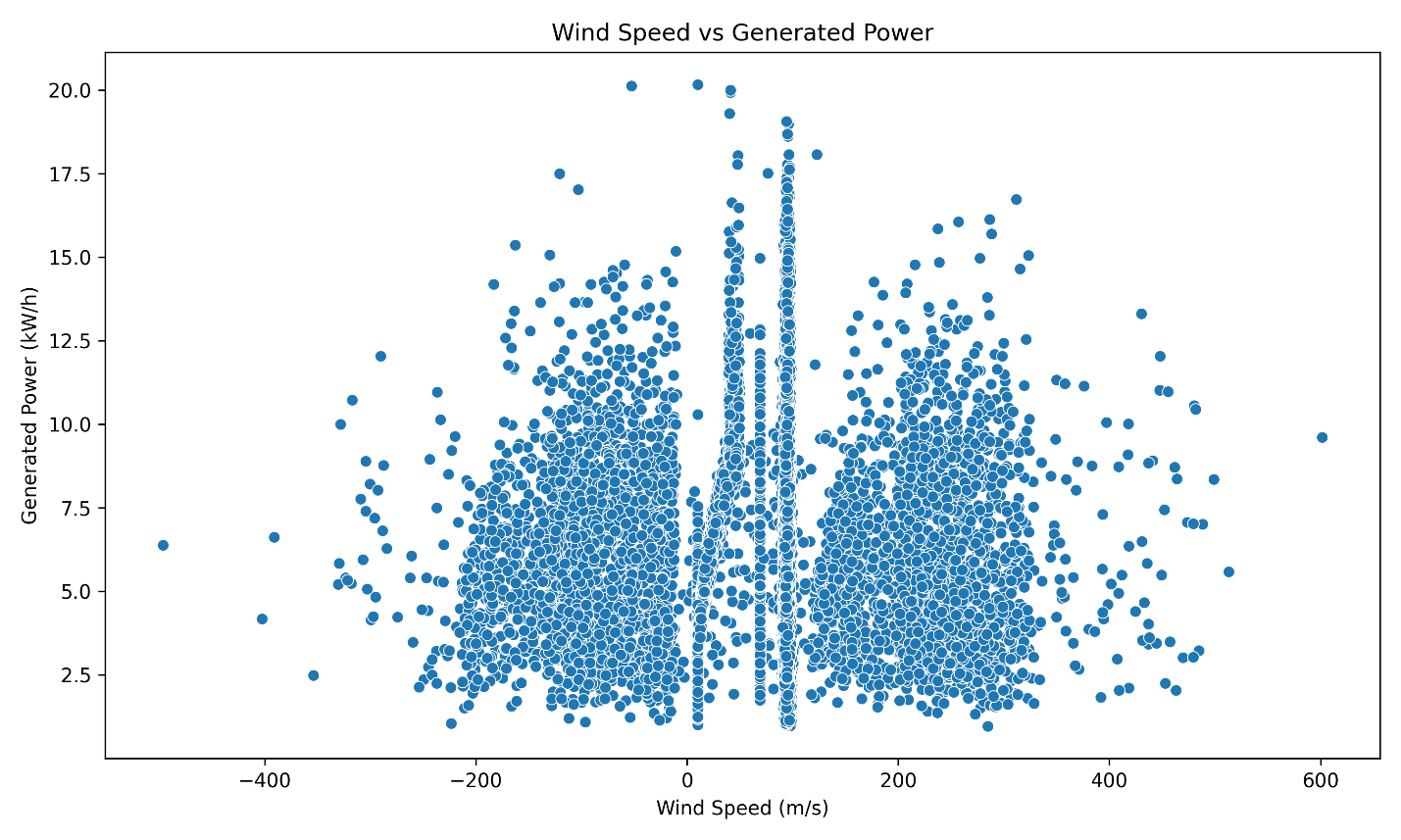


Fig 15. Wind Speed vs Generated Power

Figure 14 represents the graph for turbine efficiency against power generation indicating that almost all the power generations occurred at turbine efficiency of 0.0 and there is only one point at 3.4. The highest power 21.0 kW/h was generated at turbine efficiency of 0.0.

Figure 15 represents the graph for wind speed against power generation indicating the graph is almost symmetrical and most of the power generation data is collected in the range of -200 to 300 m/s. The highest power of 20.0 kW/h was generated at wind speed of 0 m/s.

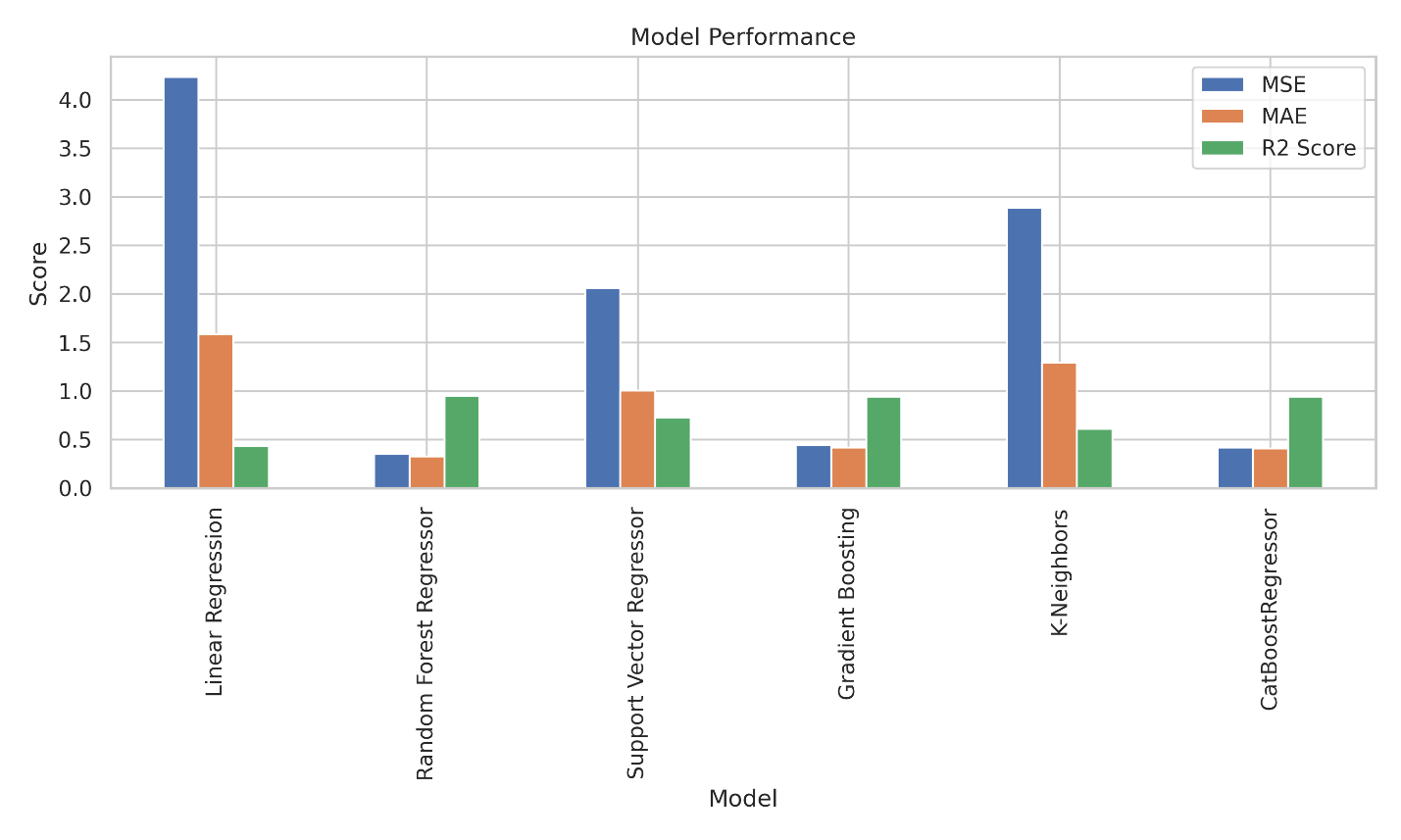


Fig 16. Model Performance

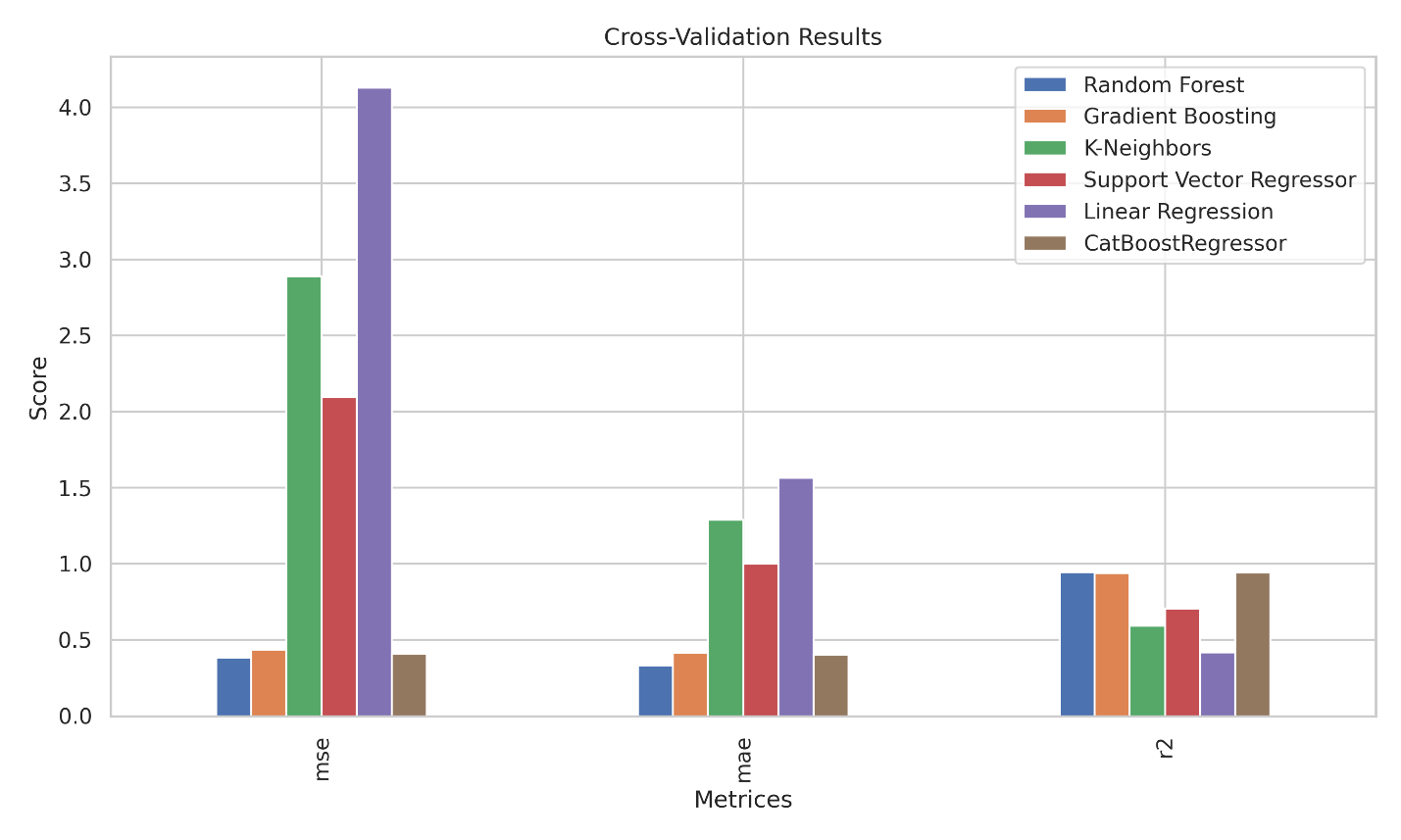


Fig 17. Cross-Validation Results

Figure 16 represents the models’ performances on the wind powered stations depicting that linear regression had the highest MSE and MAE and the lowest R2 score compared to others, meaning it will be the worst model to work on. On the contrary, random forest regressor had the lowest MSE, MAE and the highest R2 score which indicates that it is the best model to train the wind powered plant dataset.

Figure 17 represents the cross-validation results for all the models we have used the dataset to train on. It can be seen that linear regression still yielded the highest MSE and MAE and lowest R2 score indicating it is the least accurate model. On the contrary, random forest produced the lowest MSE and MAE and the highest R2 score, meaning it had the best cross validation scores.

**6.3 Hybrid Data Center Consumption Model Development**

While developing the models for hybrid data center we have used several python-based libraries to visualize the relationships between the independent and target variables. A collection of figures describing the different relationships are listed below:

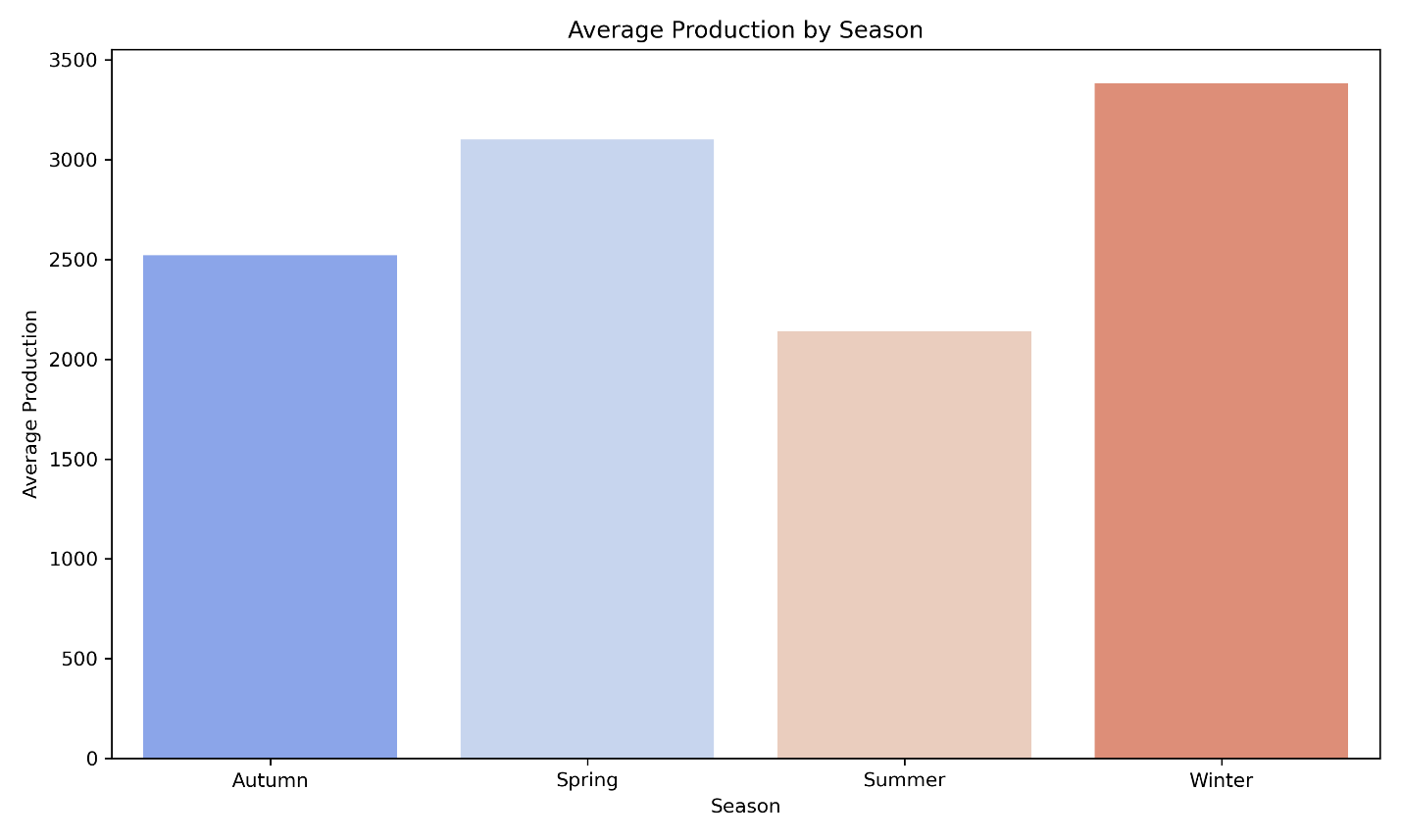


Fig 18. Average Production by Season

Figure 18 represents the average production of power in each season. It can be seen that in winter, maximum power is generated followed by spring, autumn and summer. During the winter, the sky tends to be clear and so more wind can flow, moreover, sunlight can easily fall on the solar panels. Hence, the most power generation takes place in winter altogether.

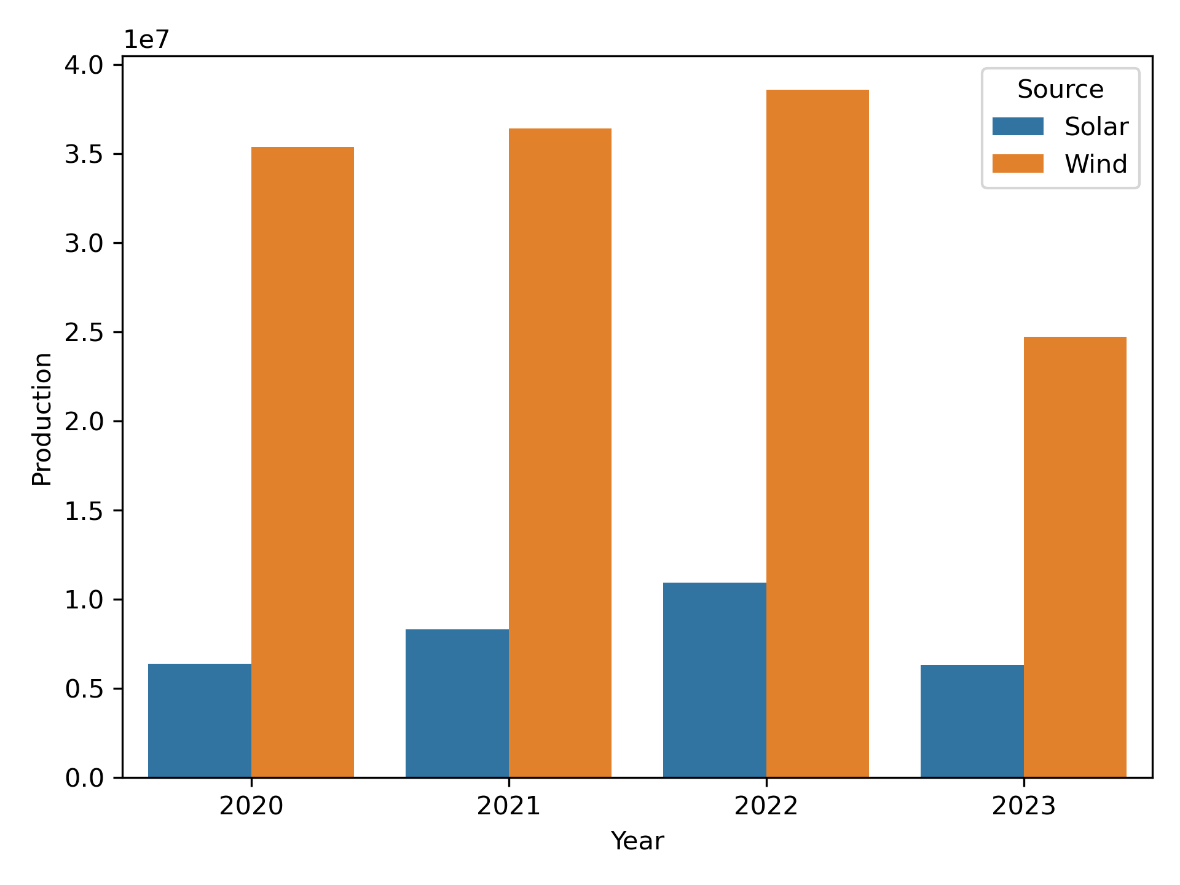


Fig 19. Annual Energy Production

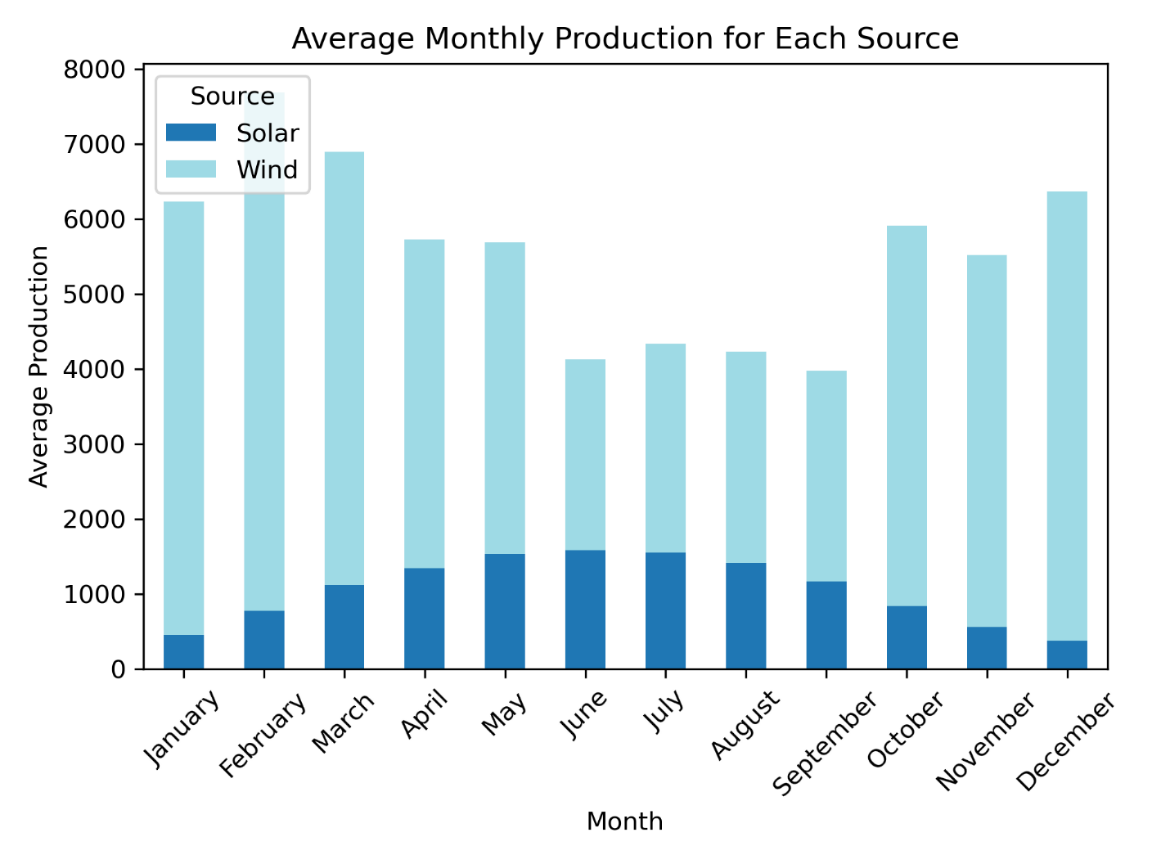


Fig 20. Average Monthly Production for Each Source

Figure 19 represents the annual energy production and it can be seen that the maximum energy was produced in the year 2022 by both solar and wind. Over the years, the production increased for both until 2022, after that, it fell on 2023.

Figure 20 shows the average monthly power production of both solar and wind resource. It can be seen that wind produced more power than solar. The highest amount of power produced was at February by wind and by solar at June. Overall, wind was the most efficient and power generative compared to solar power generation.

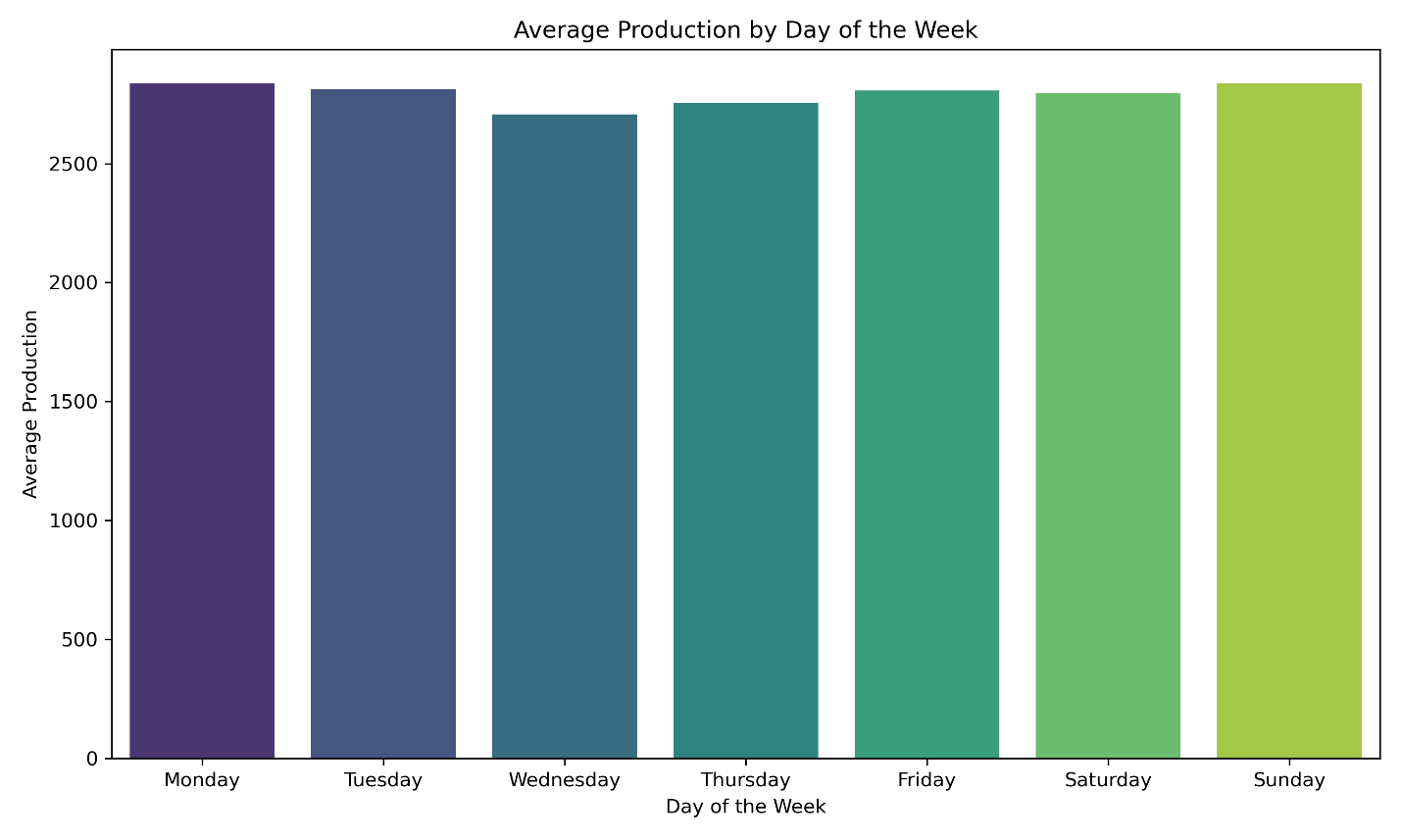


Fig 21. Average Production by Day of the Week

Figure 21 represents the average power production by each day of the week. As all the production level is pretty similar for all the days, it can be easily accepted that power production does not depend much on any specific day.

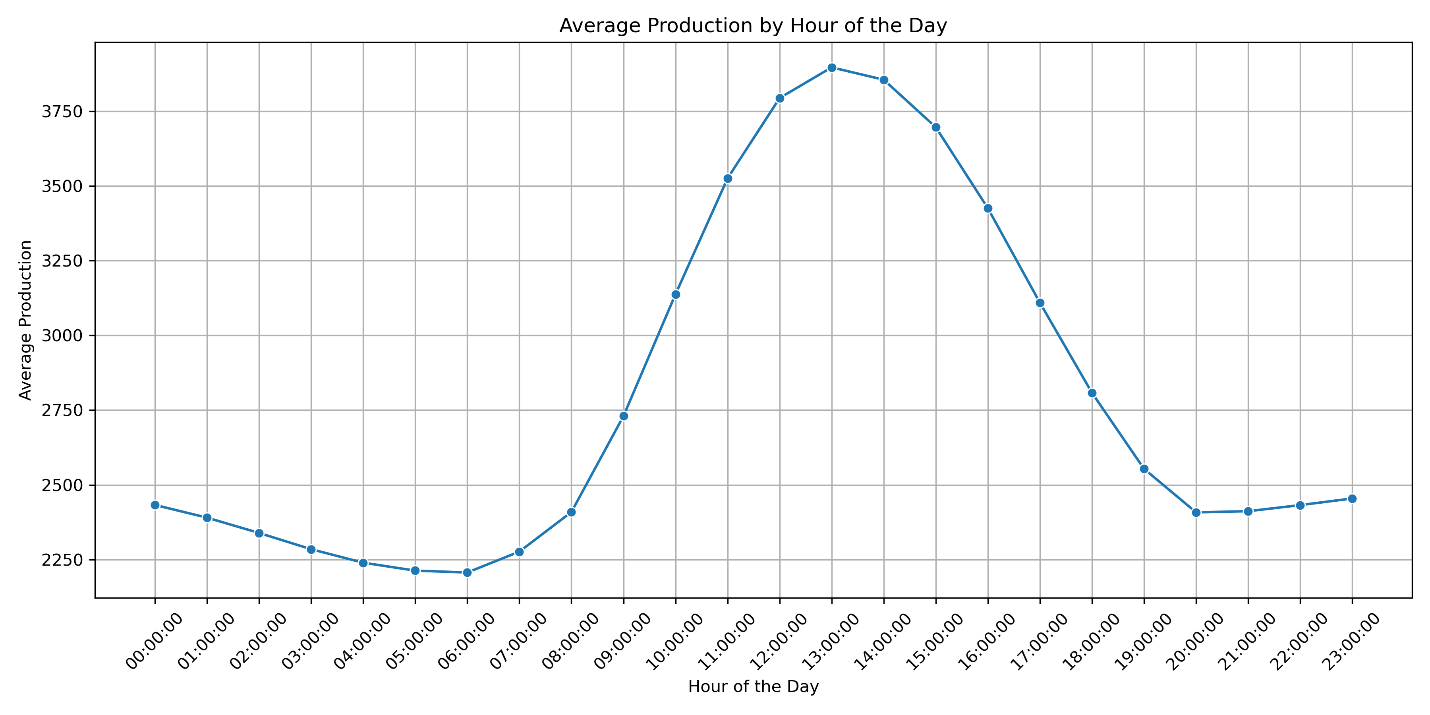


Fig 22. Average Production by Hour of the Day

Figure 22 represents the graph for average power production by hour of the day. It can be seen that most of the power is generated around the 9:00:00 to 18:00:00, for about 9 hours. This shows that it is more effective to generate power during the nights around these times.

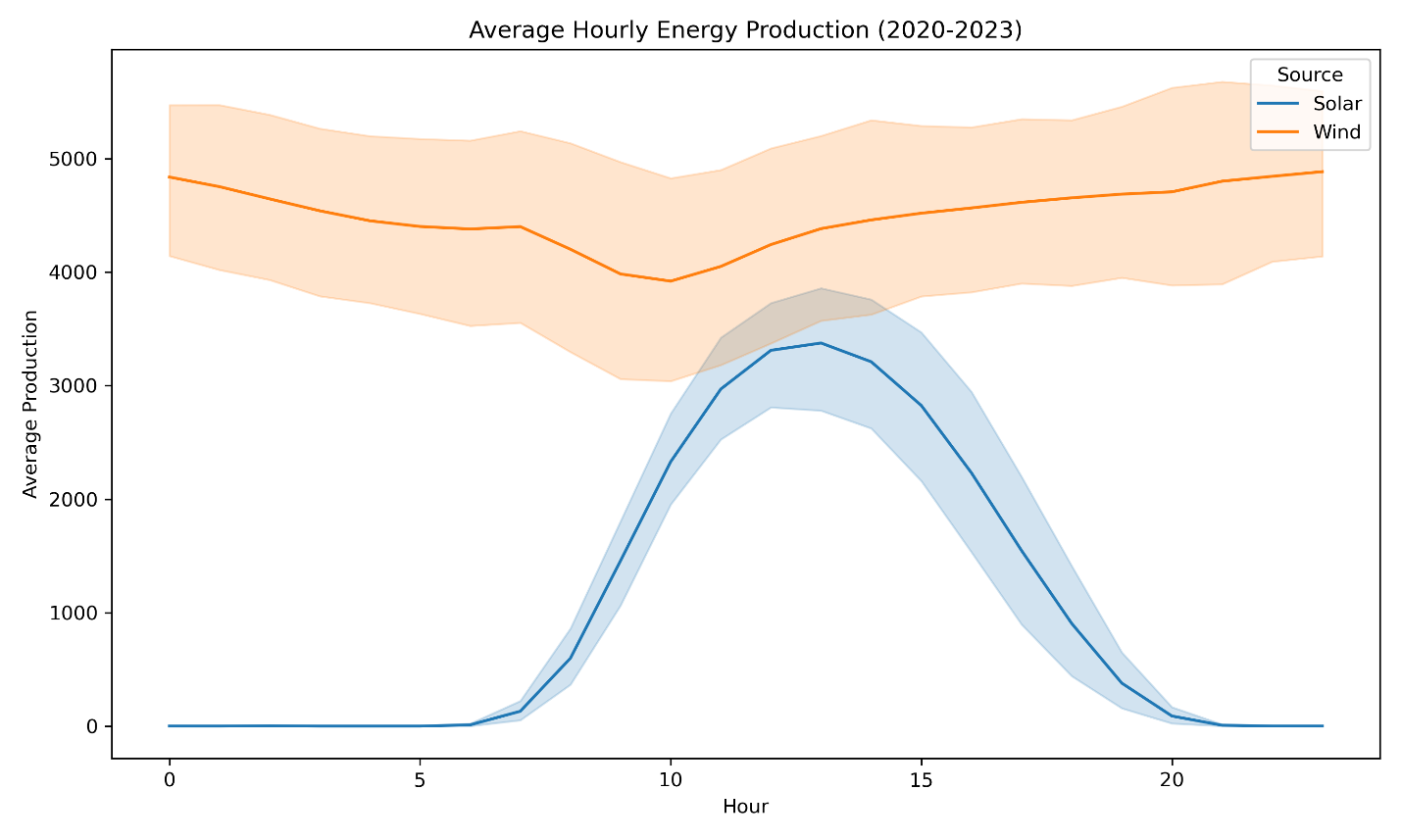


Fig 23. Average Hourly Energy Production (2020-2023)

Figure 23 shows that in the year of 2020-2023, most of the energy is generated by the wind compared to solar. But the production by wind fell during the hours of 7 to 18, which could mean that it was day and therefore, energy was efficiently generated by the solar panels.

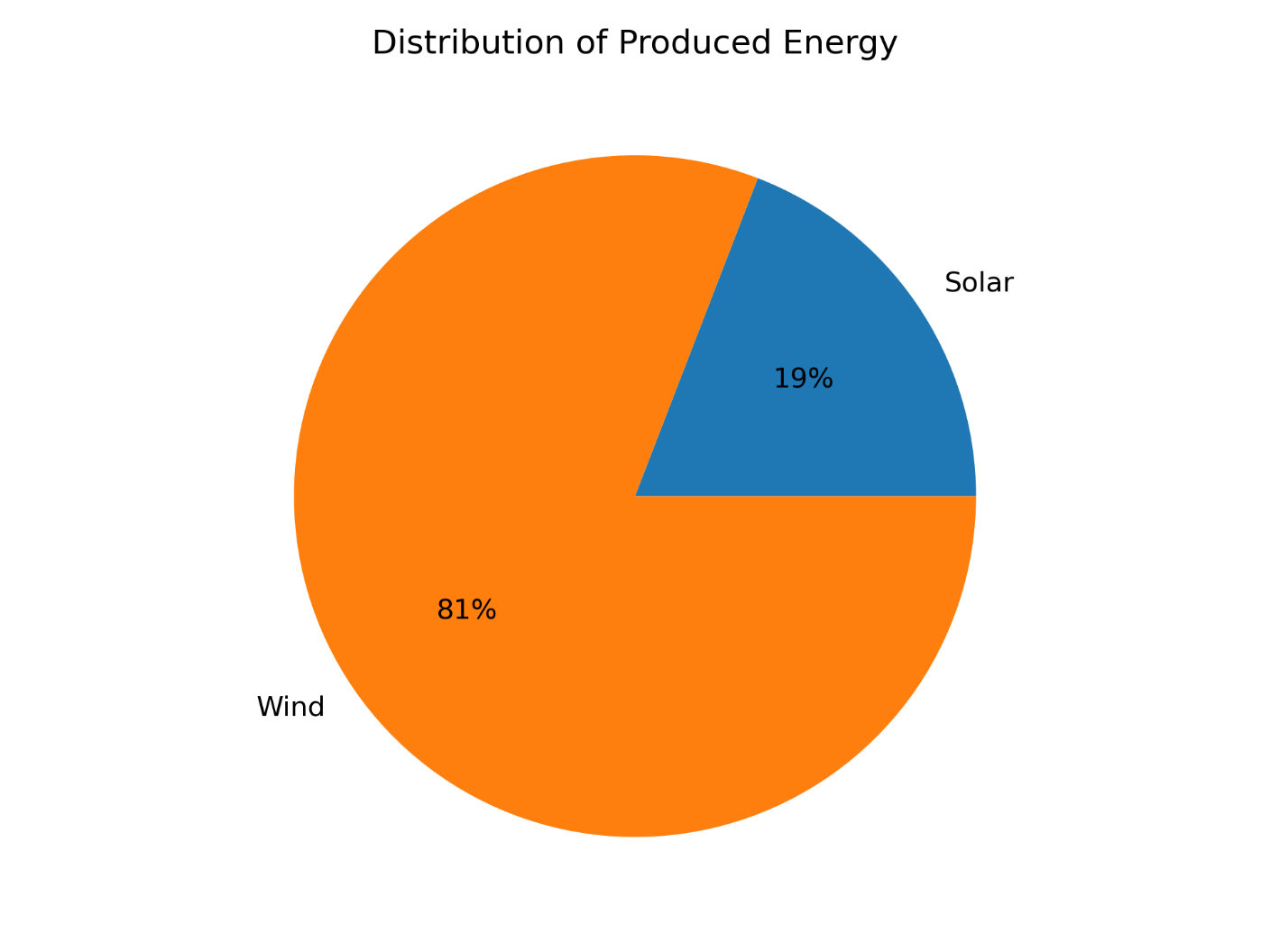


Fig 24. Distribution of Produced Energy

Figure 24 depicts that 81% of energy is produced by wind and 19% energy is produced by solar.

So, it can be effectively said that wind energy is more productive in producing energy compared to solar energy. But a factor should be considered as well i.e. a machine can not be run for 24/7, therefore, it is best to use the hybrid system in such a way that wind is used to generate energy when there is no sun and kept at rest during daylight as solar would be the best option to harness energy then.

# **Chapter 7: Testing**

**7.1 Integrate Models**

**Supervised Machine Learning Algorithm Regressor Models for Wind/Solar Power Production:** Supervised machine learning classification models, the necessary instruments for wind or solar power are so very easy to behave according to the operational status, in which the primary focus is to set up the data into interrelated groups. These models carry learning from a dataset, where both the properties of the data and the corresponding categorical target values are given. The renewable energy sector requires a process to access the relationship between different factors, for example, the meteorological situation, the time of the day, and power production data over time, and to classify the power. each category has a fixed size.

First, the process begins with data collection and preparation, identifying the variables. With regard to wind energy, for example, the data might involve temperature, wind speed, and wind direction, while, in the case of a solar plant, factors like sunshine, temperature, and humidity would be included. Usually, the category in a classification model is a written single concept that is divided according to the amount of electricity the source produces. It can be "low," "medium," or "high" production.

After the data set has been collected, it is separated into training and the testing subsets. The training data is the one that the classification model uses to create the model, and it learns how to categorize the power production using the input features. However, the learning process is all about the model's optimization so that the errors are minimized, and the categorization is done more accurately. A very frequent way of measuring the model's improvement is by applying metrics like accuracy, precision, recall, and the F1 score.

After the model is developed, the performance is scrutinized on the set testing data in order to ascertain its ability to generalize to new unseen data. One of the best methods for finding the right allocation of the power production levels according to different situations is the use of the model. The most important metrics of the classification models are accuracy, which is the exact number of power production levels classified correctly; precision, which tells how many times the model makes the right decision; recall, which represents the power production levels actually present in the input data; and the F1 score, which highlights the entire performance of the model.

The means of operation for the model starts with the teaching of the process of power level prediction resulting in a learned model, and then, the model is used with the new input data. For example, the first element of the input may be wind or solar panels, so it should be able to be able to assess the output of wind or solar panels according to their respective weather conditions if only the first ones are the inputs. In the context of wind power, the model can factor in the terrain and wind interactions in the locality, both of which can drastically impact the power output. The utilization of historical data relating to wind speed and production will enable the model to make a projection of the power output done by considering multiple influencing factors.

Consequently, when it comes to the solar-powered devices, the model considers the fact of the sun's radiation intensity and changes in weather conditions that are posited to affect power production. It goes through data analysis of prior records on sunlight radiation and power production, thus evaluating and putting in different categories the power output amounts subjected to diverse environmental conditions.

One of the most important things is that this kind of classifier has various advantages, the first of which is that they can precisely and quickly classify power production. Nevertheless, two factors, namely the quality of the data and their relevance, are central to their accuracy. Incorrect, inadequate, or even misleading data can result in less reliable predictions. Moreover, making the right decision concerning the issue of the accuracies of models and computational efficiencies also necessitates the correct understanding of how the model will be used in reality. The ultimate opportunity for the proper functioning of the implemented supervised machine learning classifiers is through the division and optimization of renewable energy sources. So, on the whole, they will make a significant contribution to the search for effective and sustainable energy solutions.

While the Random Forest Classifier is an excellent choice for categorizing various weather phenomena and operational states affecting the production of both the wind and solar power, it can be extremely effective for this purpose. For example, this method can be used to group weather conditions, such as "high wind," "low wind," "few clouds," or "cloudy," that can then be associated with the power production efficiency prediction. The strongest point of Random Forest is the ensemble method, that is to say, numerous decision trees, which work together to increase the accuracy. This technique is well-skilled in performing non-linear and complex relationships and different types of interaction that occur among characteristics such as temperature, wind speed, and humidity.

Gradient Boosting Classifier is a powerful tool in identifying potential outages or failures in power plants. Through developing a chain of models focusing on fixing mistakes of the previous ones, Gradient Boosting can increase the accuracy of the forecasting. For instance, historical data and current conditions can be used by the system to categorize days into "high risk" or "low risk" situations of turbine or panel failures. Feedback-based learning results in improved quality of predictions and brings to light causal patterns in the errors and breakdowns of the system.

Another fitting commendation for the application of K-Neighbors classifier is an "Anomaly detection in power production data." Proposing the power generation data analysis, the instance of the data when it is in trouble can be recognized and also classified from the historical patterns with the help of K-Neighbors. The system follows the principle of classifying the data by a majority of data that is closer to the data, and this is done is a very cost-effective way. It detects the days that are not peculiar in the conditions that differ from the usually used patterns.

Support Vector Classifier is the best option for solving binary classification problems, for example, it is used in the classification process it is required to decide if the power production is going to be high or low due to grid requirements. To complete this task, SVM (selects) the optimal hyperplane that arises the two classes of points, so this fund transfer business is very important because the connection between input and output is complex and nonsystematic. This model's strength of creating decision boundaries that are visible can be used to manage production levels accurately.

However, Linear Classifier is more suitable for simple sorting tasks, where power production and its linear relationships with the input features are the decisive constraints. For instance, it would tell whether power production will be above or below a specific threshold through a simple linear separator. The straightforwardness and excellent computing time of this model make it a good option for the situations where the relationship is linear.

Let's not forget that CatBoostClassifier is really good at a number of situations, such as a data set including categorical features e.g. weather conditions or types of equipment. CatBoost uses a smooth version of the gradient boosting algorithm with the additional handling of categorical data that is much more effective than many of the other models. This is due to its capability of managing and encoding categorical variables rightly. Thus, CatBoost can describe different power production situations based on the underlying variables' complicated interactions, thus becoming a quite powerful tool that is instrumental for the study of complex data.

When performing a GridSearchCV algorithm on the Random Forest Classifier to get the energy production via wind and solar, one first has to define a grid of hyperparameters like the number of trees in the forest(n\_estimators), maximum depth of the trees(max\_depth) as well as the minimum number of samples required to split the internal node(min\_samples\_split). The GridSearchCV method on stitching tries out all the permutations of them (parameters) by using switch technology techniques, enabling operators of information system to test the software. You can find the suite consisting of one, for example, which yields the best recognition rate under some measure.

One of the key topics of study of the project was in the NBcom class, a class of web-based educational systems, including both technical and non-technical support ends. The main architecture and the different levels that can be included can be shown here on the second level of the 3-tier architecture. It is a statistical technique that will help the model in accurately predicting the risks by using historical and present data effectively.

When working with the K-Neighbors Classifier, GridSearchCV may be used to adjust parameters such as the number of neighbors (n\_neighbors) and the distance metric used to determine the neighbors (metric). GridSearchCV checks other combinations to make the classifier work quite well in identifying the anomalies in power production data. This basically guarantees that our model is able to properly perceive divergence from the historical pattern as part of our vision for the complex of subsequent reliable operations.

When analyzing the data, Support Vector Classifier (GC) and GridSearchCV, can effectively optimize the important hyper-parameters such as the regularization parameter (C), the kernel type (kernel), and kernel-specific parameters like the gamma parameter for RBF kernel (gamma).

By looking at different combinations of these parameters, GridSearchCV can figure out the best settings that allow certain classes of power production to be separated, thus the performance will be improved in the binary classification tasks.

For the Linear Classifier, A GridSearchCV enables the tuning of parameters like the regularization strength (C) and the type of regularization (L1 or L2). The iterative process helps to formulate a more realistic model using simple classifications, for example, predicting the aroundness of a line with the input features.

Finally, in the case of the CatBoostClassifier, GridSearchCV will be an assistant in searching, and finding the best hyperparameters among which are the learning rate (learning\_rate), the number of boosting iterations (iterations), and the depth of the trees (depth). By trying different combinations of techniques, GridSearchCV guarantees that CatBoostClassifier handles categorical and not so categorical variables efficiently and makes the most accurate predictions based on complex interactions throughout the dataset.

**Supervised Machine Learning Algorithm Regressor Models for Hybrid Model:** In hybrid solar and wind power plants, supervised machine learning algorithms, mostly regression models, are very vital in predicting and optimizing generated power. These models were developed to learn from the history of data and make accurate predictions of outcomes in the future using a given set of variables.

First of all, one needs in supervised machine learning a dataset—a historic record of measurements—of the hybrid plant's power output and at least some of the environmental parameters that have factors of solar irradiance, wind speed, temperature, and other meteorological conditions. Thus, the quality and relevance of such data are prepared with great care. Preprocessing may include cleaning from anomalies, normalization or scaling values, and handling missing information.

Subsequently, the prepared data is trained in a regression model. More precisely, this is a supervised learning algorithm that predicts a continuous outcome. For example, with a hybrid power plant, the continuous outcome is the amount of electrical power produced It is through this process that the regression model learns to map the input features, such as solar and wind conditions, onto the target output. The trends and correlations in historical data teach the model what changes in environmental variables predict changes in power production.

In the training process, the model is fed input-output pairs from the historical data. It would then tune its internal parameters so that its predictions best agreed with the actually observed power output in data. This is an iterative process of optimization in accuracy for the model. By the end of this phase, it will have learned how the various factors—both as discrete and combination effects—interact to influence power generation at the plant.

This will be usually tested on another subset of data called a test set after training is done. Since this testing data has not been used in training, it provides a more objective measure as to how well the model generalizes completely new data. The kinds of metrics for evaluating performance include accuracy, precision, and error rates. If the model performs well, then it can be used with present environmental conditions to predict future power production.

There are multiple forms of regression models that might be used depending on the nature of the relationships in the data. Simple linear regression involves just a direct, linear relationship between input features and power output. Real-world scenarios will involve more complex interactions. Polynomial regression generalizes linear models by including polynomial terms of order higher than one, which model non-linear relationships. Ensemble methods combine hundreds or thousands of different regression models to provide better accuracy and improve on complex, nonlinear relationships. It combines the predictions of several models to come up with a more reliable estimate.

Another advanced approach incorporates neural networks, especially deep learning models. These are models characterized by a large number of connected nodes, called neurons, over multiple layers that enable learning very intricate patterns and interactions within the data. In very large datasets with complex relationships, neural networks become very powerful and return very accurate predictions due to their deep understanding of the underlying data.

If correctly chosen and validated, this regression model is ready to be applied in many power plant applications. It will predict live the amount of power production by using live weather data, thus assisting the operator in making decisions related to energy production and grid integration. It will also be applied in the optimization of the operation of the plant by predicting when and how adjustments in the settings should be made to achieve maximum efficiency and reliability.

Supervised machine learning regression models give a modern tool for hybrid solar and wind power plant management. Given historical data, they predict the future output of power, allowing better optimization of energy resources and more effective integration of renewable energy into the grid. The ability to model complex relationships and provide accurate forecasts of power output has made this research quite instrumental in improving efficiency and reliability in hybrid power systems.

In the context of supervised machine learning algorithms applied to the optimization and prediction tasks for hybrid solar and wind-powered stations, several models can be used for performance and efficiency enhancement. Here's a detailed explanation of how each of these models works and its applicability to the domain.

Decision Tree Classifier: It's one of the simplest algorithms that is used to classify. Basically, it works by branching the data to base decisions on specific features. Decision Trees can be used to either classify the various operation states or predict maintenance needs against several input features for hybrid solar and wind-powered stations, such as weather conditions, energy output levels, and indicators of the health status of the system. One major advantage of decision trees is their intuitiveness and relatively easy interpretability of the rules the classification decisions are based on. Those may turn out to be especially useful in diagnosis or for strategic decisions on system management.

Random Forest Classifier generalizes the idea of decision trees through an ensemble approach: it creates many decision trees during training and merges the results for stability and accuracy. For example, in a hybrid solar and wind power station, random forest improves the predictive performance through aggregating predictions from many decision trees, hence reducing overfitting and improving generalization. The model will be very effective if there are complicated interactions among different features like solar irradiance, speed of wind, data on the generation of energy historically. It gives more reliable predictions and classification.

Logistic Regression, as the name may suggest is rather used for classification tasks and not regression. It models event probability for a binary outcome based on input features through a logistic function. Logistic regression, when applied to hybrid energy stations, enables the prediction of the likelihood of certain events, like equipment failure or the probability of surpassing thresholds of energy production. This is a very simple, yet effective way to approach binary classification problems, especially where one would like to see the impact of various factors on an outcome for operational planning purposes of decision-making.

Gradient Boosting Classifier: Gradient boosting classifier is a much more complex ensemble method in which a series of weak learners—in most cases, decision trees—are built sequentially; that is, each new model corrects the mistakes of the previous ones. This iterative approach is one through which gradient boosting helps in the development of a robust predictive model by making combinations of predictions from several other weak models. By capturing fine patterns and complex interactions inherent in the data, gradient boosting can be an effective tool when it combines with hybrid solar and wind-powered stations. Therefore, handling nonlinearities and interactions, makes it suitable for tasks that have a high requirement for predictive accuracy and robustness.

The other ensemble method is referred to as the AdaBoost Classifier, which means Adaptive Boosting. This approach will be to manipulate the weights on instances misclassified to enhance the performance of weak learners. AdaBoost is another meta estimator that combines weak classifiers into a single strong classifier in such a manner that every classifier is trained with much emphasis on data points misclassified by previous classifiers. In the context of hybrid energy systems, this makes AdaBoost useful for improving the performance of classification in applications such as fault detection or performance evaluation where classes may be underrepresented and thus harder to predict accurately.

K-Nearest Neighbors (KNN) Classifier: This is another non-parametric approach that classifies a new data point by the majority class among its k nearest neighbors in the feature space. The working of KNN is simple and based on simple distance metrics to make a prediction. In the case of hybrid solar and wind-powered stations, KNN can be useful in tasks like categorizing different operational modes or predicting energy output levels based on similar historical instances. It is very practical to use because of its simplicity and effectiveness in dealing with different kinds of data, despite being computationally intense in the cases of large datasets or high-dimensional feature spaces.

These models shall have individual strengths, starting from the simple and interpretable decision trees and logistic regression up to complex ensembling techniques, like random forest and gradient boosting, in the management and optimization of hybrid solar and wind-powered stations. Model choice often depends on the specific requirements of the task at hand, including requirements for interpretability, accuracy, or the ability to handle complex interactions within the data.

For instance, GridSearchCV is a very important tool for hyperparameter optimization during enhanced performance of supervised machine learning models built for hybrid solar and wind-powered stations. GridSearchCV entails the evaluation of a grid of predefined hyperparameter values by training the model with such settings and evaluating its performance using cross-validation. GridSearchCV can test a number of optional settings in the Decision Tree Classifier, such as the maximum depth of the tree or the minimum number of samples required at a node. This guarantees enabled parameters with a careful balance regarding model complexity against generalization to ensure correct state classification of operation or predicted maintenance requirements in energy systems. It is in this respect that GridSearchCV would go a step further to optimize the number of trees in the forest or the maximum depth of the tree in a Random Forest Classifier. This step improves the accuracy and robustness of the ensemble model, as it picks the best combination of parameters, raising predictability in energy output or even system efficiency.

Interestingly, with Gradient Boosting Classifier, through GridSearchCV, exploration is done over various learning rates, the number of boosting stages, and the maximum depth of trees in boosting. Further fine-tuning of these parameters, the GridSearchCV will be enabled to develop a model more efficiently, which may correctly capture the complex patterns and interactions of data with respect to energy production and system performance.

For models like K-Nearest Neighbors, the number of neighbors and distance metric used can be optimized sensibly using GridSearchCV, which will give a conjunctive result. It seeks optimal settings that affect how distances are calculated and how neighbors are considered to ensure exactly the classification of energy production levels or system states.

Through the use of GridSearchCV, practitioners can ensure not only model effectiveness but also model overfitting resilience, thanks to the cross-validation process. This results in more reliable predictions and classifications, which are critical for efficient management and optimization of hybrid solar and wind-powered stations.

# **Chapter 8: Analysis**

**8.1 Validation Result Analysis**

Table 3. Initial Model Results for Windmill Power Production

|  |  |  |  |
| --- | --- | --- | --- |
|  | MSE | MAE | R2 Score |
| Linear Regression | 4.237552 | 1.585079 | 0.432065 |
| Random Forest Regressor | 0.352166 | 0.325774 | 0.952801 |
| Support Vector Regressor | 2.062976 | 1.002725 | 0.723511 |
| Gradient Boosting | 0.442132 | 0.419548 | 0.940744 |
| K-Neighbors | 2.892978 | 1.295511 | 0.612271 |
| CatBoostRegressor | 0.417199 | 0.408115 | 0.944085 |

The table titled "Initial Model Results for Windmill Power Production" in the text shows three different regression models results which have been examined with regard to their prediction of wind power production from windmills. The table consists of the following: Mean Squared Error (MSE), Mean Absolute Error (MAE), and R-squared (R²) score for each model.

Mean Squared Error (MSE) measures the average squared difference between the predicted and actual values, with lower values of the better model performance. It is the model "Random Forest Regressor" which has been the most accurate in predicting windmill power production more times. It has the smallest squared error with MSE = 0.352166, which is the lowest among other models, meaning there is less difference between the actual and predicted windmill power production. So, the Random Forest Regressors are valid and the most accurate among them. On the other hand, linear regression is the one that shows the highest MSE. As a result, it gives 4.237552 as the value of the model's prediction error, which is higher than the other models.

The Mean Absolute Error (MAE) is a measure showing the average size, in terms of difference between the predicted and actual values, of the errors in the predictions. The Random Forest Regressor is the model that performs best according to the metric of MAE which is the lowest of 0.325774. This confirms its legitimacy. In contrast, the Linear Regression variety displays the most MAE of 1.585079, thereby giving an indication that, on average, its predictions are more divergent from the actual values compared to the predications given by the other models.

The R-squared (R²) score indicates the proportion of variance in the dependent variable that is predictable from the independent variables. The R² score is a statistical parameter that sheds light on the model's frequency of fitting a dataset. The answer is the Random Forest Regressor first with a high R² score of 0.952801 because it captures about 95% of the variance in the power output. And after that, models changing and tight are set having also high R² ones: The R² cores for Gradient Boosting and CatBoostRegressor are 0.940744 and 0.944085, correspondingly tell that they are promising in the prediction good performance. As an opposite Linear Regression has only the highest R² score indicator having 0.432065. This follows that Linear Regression can be put as a model using Varying coefficient regression for grouping with R being the variable. Hence, from the variance perspective, the data is well represented by the output from one model which is R²=0.432065 and it is the inverse of the other values which are increasing for the other models.

For the most part, the Random Forest Regressor model is the most stable one in the context of windmill power production prediction. Its unmatched results in the MSE, MAE, and R² scoring systems are the manifestation of its competence to capture the data's patterns and give forecasts with high precision.

Table 4. Optimized Model Results for Windmill Power Production

|  |  |  |  |
| --- | --- | --- | --- |
|  | MSE | MAE | R2 Score |
| Random Forest | 0.422544 | 0.377724 | 0.943369 |
| Gradient Boosting | 0.308694 | 0.323037 | 0.958628 |
| K-Neighbors | 2.737085 | 1.265830 | 0.633164 |
| Support Vector Regressor | 1.670192 | 0.913549 | 0.776154 |
| Linear Regression | 4.235872 | 1.585038 | 0.432290 |
| CatBoostRegressor | 0.334904 | 0.346686 | 0.955115 |

The table titled "Optimized Model Results for Windmill Power Production" enumerates the performance indicators of the various regression models after optimization, concentrating on the one that best predicts wind turbine power generation. Among the metrics included are mean-square error (MSE), mean absolute error (MAE), and R-squared (R²) score.

Mean Squared Error (MSE) measures the average squared differences between predicted and actual values, with smaller values reflecting better models. In this case, the model with the least MSE of 0.308694 is Gradient Boosting, which leads the race by the slimmest of margins in completing the predictions with the smallest average squared error of the rest of the optimized ones. It is, therefore, an indication of the possibility of data leakage or multicollinearity for Gradient Boosting which may have lent it to overfitting. A disturbingly high MSE of 167.60 is evidence of these. Error is the difference between the truth and what the model believes in this case the lower the value the better the model is performing.

Mean Absolute Error (MAE) is a mean of the differences of the actual and predicted values as with the negative errors' signs removed, the average positive errors. Equally, norming the value gives a new normalized. Value, in which every parameter that you passed in becomes a new parameter that is normed according to the amount it contributes to the prediction. Like min-max, the gradient is either zero or infinity for the activation cells. In those cases, the regularization term should be active to limit the rate of learning and stop the model from diverging.

The R-squared () score is a measure of the amount of variance in the dependent variable explained by the independent variables. A high R² score suggests a well-fit model for data. The model with the highest R² score of 0.958628 is the Gradient Boosting model, implying that it explains most of the windmill's energy production variance, and therefore, it seems the most effective model. The model called CatBoostRegressor, on the other hand, has a rating of 0.955115 which of the above is ideal for highway energy harvesting. Linear Regression on the other hand has a low R² score of 0.432290 which shows that it is far less effective and the most inadequate model.

In general, the commanded Gradient Boosting model demonstrates it standing out as the best model for predicting the windmill power production, being the best in all three-performance metrics compared to others: MSE, MAE, and R². This means that the optimization process has, indeed, brought this change. CatBoostRegressor is also good at MAE and R², but the Gradient Boosting algorithm outperforms it in the general case of prediction. Linear Regression, which still has a poor performance notwithstanding its optimization progress, is now showing more room for improvement compared to the other models.

Table 5. Initial Model Results for Solar Power Production

|  |  |  |
| --- | --- | --- |
|  | MSE | MAE |
| Linear Regression | 0.155032 | 0.266655 |
| Random Forest Regressor | 0.000000 | 0.000000 |
| Support Vector Regressor | 1.000000 | 1.000000 |
| Gradient Boosting | 0.024038 | 0.060821 |
| K-Neighbors | 0.123471 | 0.148265 |
| CatBoostRegressor | 0.012205 | 0.037256 |

Table "Initial Model Results for Solar Power Production" shows a summary view of how well the different regression models do in solar power production without any type of optimization. In these results, resulting metrics include the Mean Squared Error and the Mean Absolute Error. Mean Squared Error (MSE) is the average of the squared difference between predictions and the observed/target data points. The objective is to reduce the value of MSE while fitting a model to the dataset. It is clear from Table 3 that the value of MSE for the Random Forest Regressor attains the minimum value of 0.000000; it leads to perfect prediction of solar power production without error and, hence, gives an excellent sign of aptitude. CatBoostRegressor is also having an efficient performance with a quite lower value of MSE: 0.012205, which is my average squared error while predicting. MSE for Gradient Boosting: 0.024038, showing fairly good accuracy, just marginally less than the most optimum CatBoostRegressor. The Support Vector Regressor holds an accuracy of 1.000000 with an MSE, making this model comparatively highly erroneous in its predictions than others.

Mean Absolute Error represents the average magnitude of errors in the predictions. A lower MAE shows that the model makes its estimations much closer to the actual ones. The same applies here, where Random Forest Regressor recorded zero errors in its predictions, while its MAE is equal to 0.000000. It scores very well; its MAE equals 0.037256, which means a pretty low average error is being fulfilled. On the other hand, Gradient Boosting with an MAE of 0.060821, while pretty respectable, is much lower, considering that most algorithms show about the same performance in purely accurate predictions. In contrast, Support Vector Regressor has the highest MAE equal to 1.000000, which means the model's predictions deviate, on average, from the actuals by a great extent. On the whole, the MSE and MAE for the prediction of solar power production with the Random Forest Regressor model are really outstanding, even in this baseline evaluation. CatBoostRegressor also did well, particularly in MAE; being high in the value gives a high predictive accuracy. Though good enough, Gradient Boosting's accuracy is lesser than the best-preferred models, that is, CatBoost and XGBoost. K-Neighbors and Support Vector Regressor are less effective, as reflected by their higher MSE and MAE, with Support Vector Regressor being the worst among the listed.

Table 6. Optimized Model Results for Solar Power Production

|  |  |  |
| --- | --- | --- |
|  | MSE | MAE |
| Random Forest | 0.048085 | 0.148474 |
| Gradient Boosting | 0.076384 | 0.000000 |
| K-Neighbors | 0.595918 | 0.526512 |
| Support Vector Regressor | 1.000000 | 0.944024 |
| Linear Regression | 0.950576 | 1.000000 |
| CatBoostRegressor | 0.000000 | 0.038685 |

The following is the table "Optimized Model Results for Solar Power Production", which displays the analysis of different models created through simulation and optimized to go W.R.T. the solar power generation. The two key performance metrics presented in this table are Mean Squared Error and Mean Absolute Error.

The mean squared error is one of the metrics used to estimate the mean squared difference between the forecasted and factual values. The lower the MSE is, then the model is better optimized with respect to prediction accuracy. Out of all, MSE values of 0.000000 really make CatBoostRegressor stand out very accurately in predicting the production of solar power, with no squared error. Thus, this is indeed a remarkable feat that signifies the CatBoostRegressor has been optimized to the data that it conforms to. The Random Forest model also exemplifies a good fit with its MSE being very low at 0.048085, which suggests quite accurate predictions, yet not as improved as the CatBoostRegressor. The Support Vector Regressor and Linear Regression models are on the other end of the scale, being the models that have the highest MSE values (1.000000 and 0.950576, respectively), thus their predictions are the least accurate compared to the other models.

Mean Absolute Error (MAE) calculates the errors’ average absolute value that could be made as predictions besides their orientation without regard to their sign. It is not essential to mention that the lower MAE is, the closer to the foreseen figures the model’s predictions are. Gradient Boosting results in a perfect MAE of 0.000000, which is extremely good and indicates no error. CatBoostRegressor also stands out with the MAE value of 0.038685 and implies very low average prediction error. In contrast, only the Linear regression model has the highest MAE with a value of 1.000000, which points to a fault in the predicted values. The Support Vector Regressor also has a large MAE value of 0.944024, which suggests considerable prediction error. On the whole, solar power production forecasting is dominated by CatBoost Regressor which is the #1, being the best track record in both MSE and MAE. Its capability to obtain almost the same errors in both criteria shows its superlative correctness. In the meantime, the Gradient Boosting model also does an excellent job, especially in terms of MAE. In contrast, its MSE is a little higher than CatBoostRegressor's. The Random Forest model although it is effective, it has not yet achieved near-perfect performance as CatBoostRegressor and Gradient Boosting. The Linear Regression and Support Vector Regressor are not as effective as in the case of the other models, demonstrated by their higher MSE and MAE scores implying that they display bigger prediction errors.

Table 7. Model Performance on Hybrid Power Generation System

|  |  |
| --- | --- |
| **Model** | **Accuracy** |
| Decision Tree | 89% |
| Random Forest | 90% |
| Linear Regression | 90% |
| Gradient Boosting | 88% |
| K-Nearest Neighbors | 88% |
| AdaBoost | 82% |

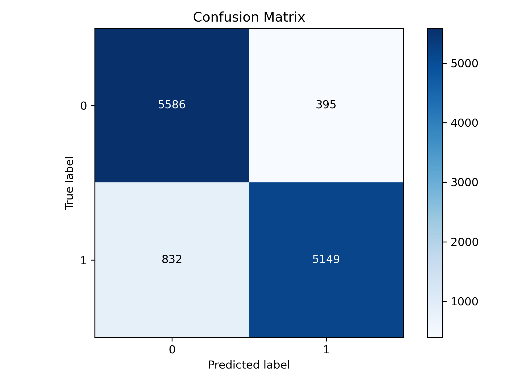
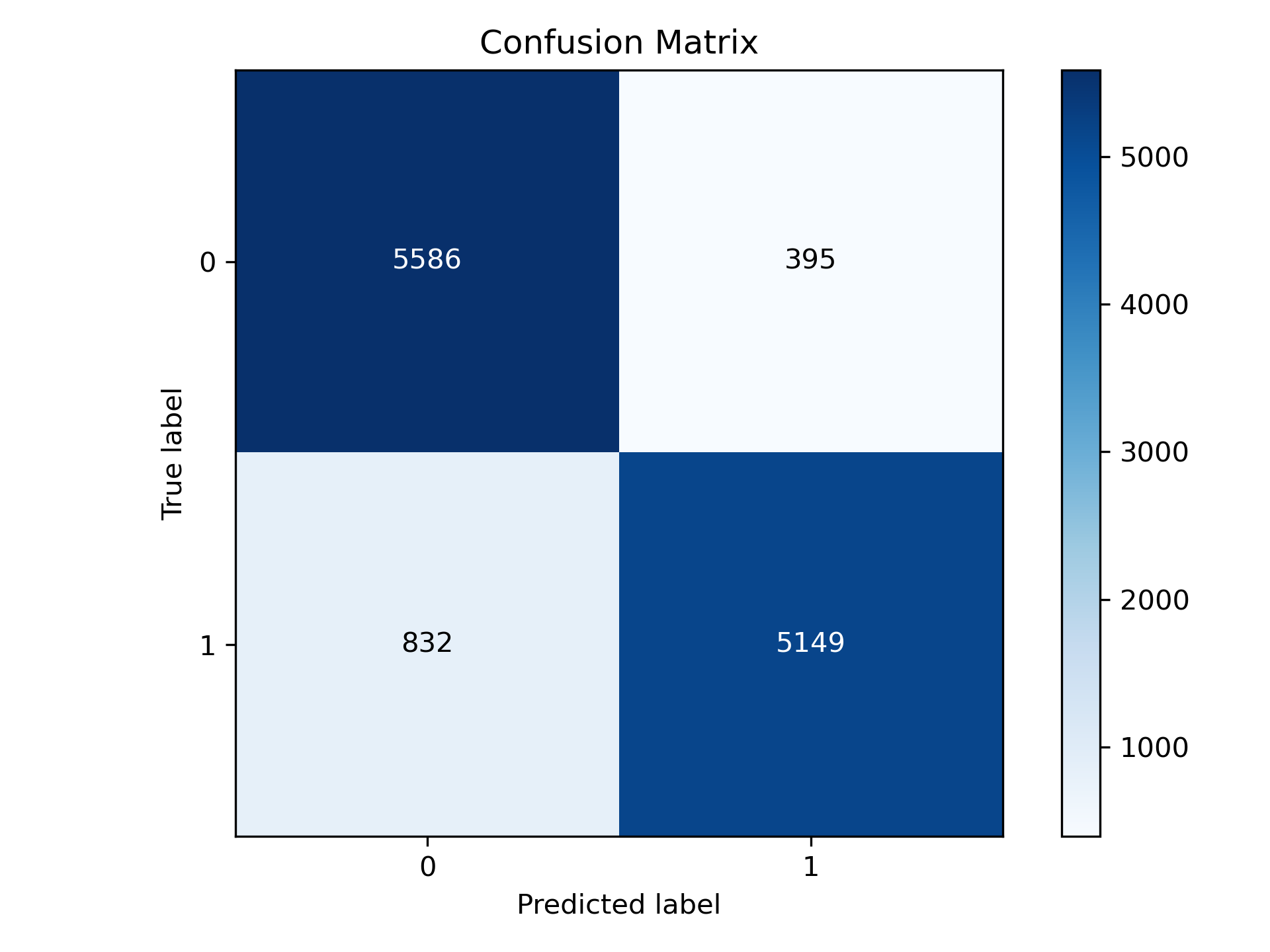
 

Fig 25. Random Forest CM Fig 26. Linear Regression CM

Fig 25 shows the confusion matrix of random forest where it depicts the results of the model’s performance. The results are as follows: True Negative = 5586, True Positive = 5149, False Negative = 395, False Positive = 832.

Fig 26 shows the confusion matrix of linear regression where it depicts the results of the model’s performance. The results are as follows: True Negative = 5586, True Positive = 5149, False Negative = 395, False Positive = 832.

These two confusion matrixes produce the results for the best performing models having the highest possible accuracy of 90%.

**8.2 Analysis of Test Optimization**

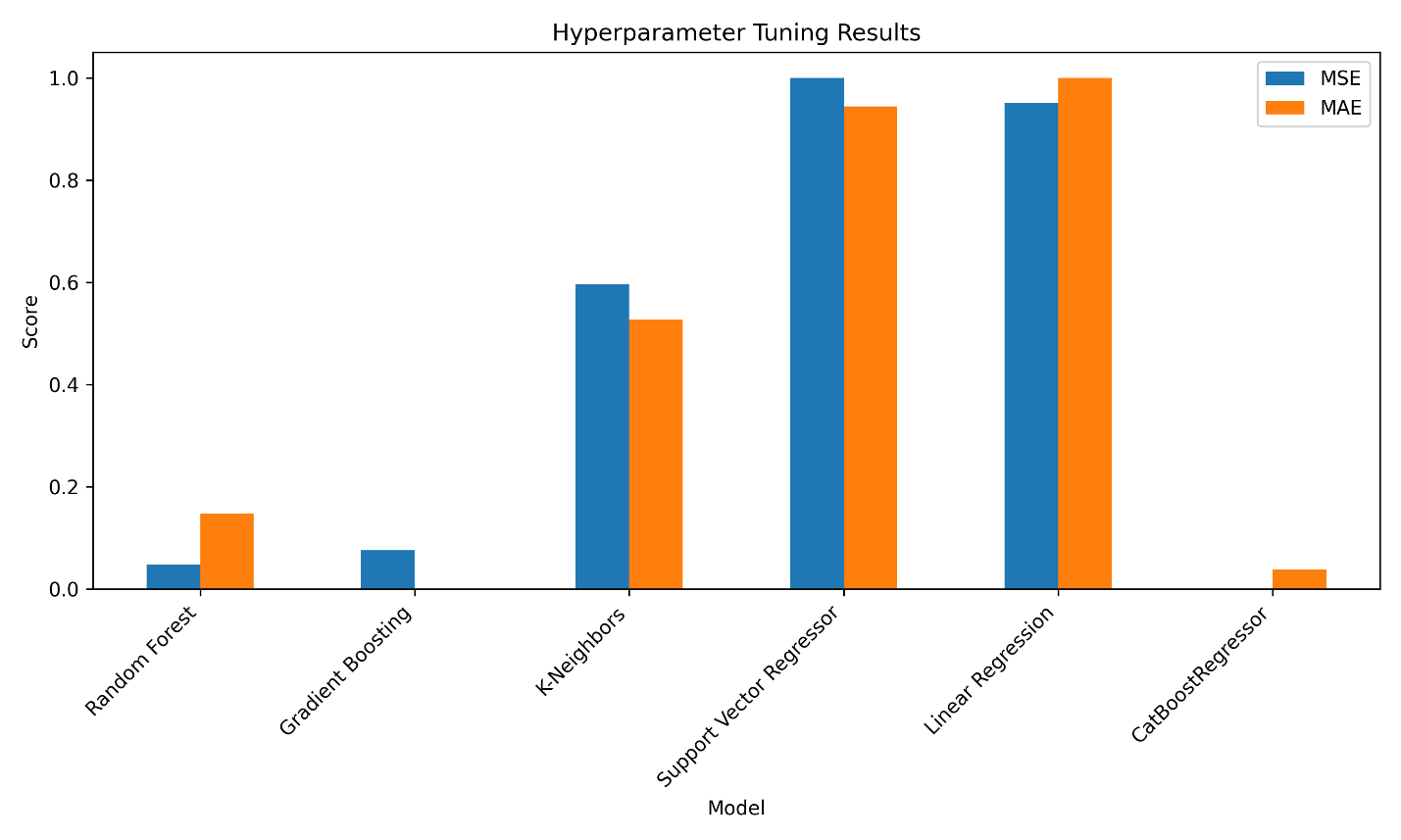


Fig 27. Hyperparameter Tuning Results (Solar)

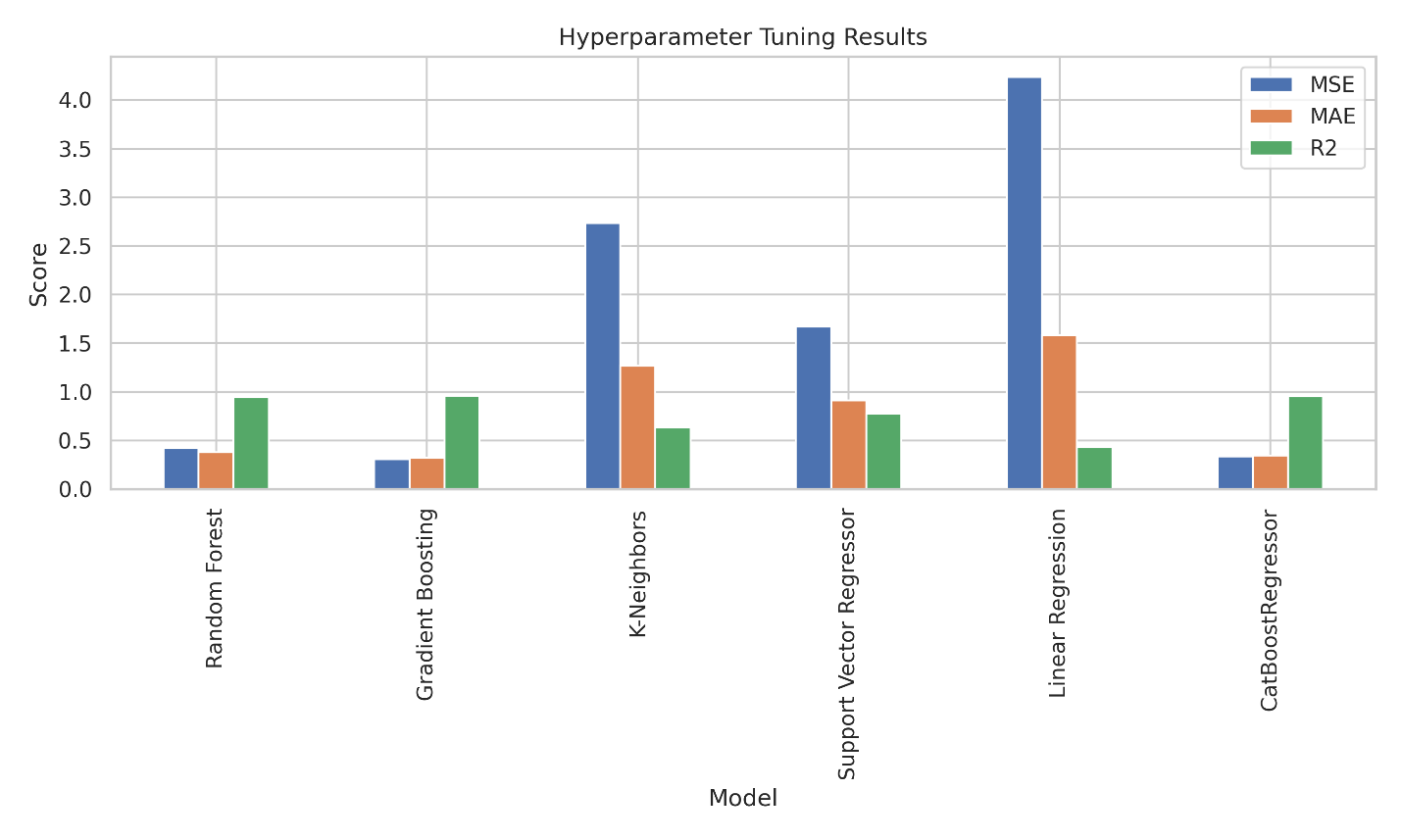


Fig 28. Hyperparameter Tuning Results (Wind)

Figure 27 represents how the performance results changed after tuning the hyperparameters of the solar powered generator dataset. It can be seen that SVC and Linear Regression had the highest MSE and MAE respectively, indicating they caused the highest errors and hence should not be used. On the other hand, CatBoostRegressor and Gradient Boosting had the lowest MSE and MAE respectively, and should be considered more efficient and reliable to use.

Figure 28 shows the bar graph for the hyperparameter tuning results for all the models used in training the wind powered generator dataset. It is found that linear regression still had the highest MSE and MAE and the lowest R2 score and hence should not be used. On the other hand, gradient boosting and catboostregressor yielded the lowest MSE and MAE and the highest R2 score, indicating they should be given priority as they would predict more accurately.

Fig 29. Hyperparameter Tuning Results (Hybrid)

Figure 29 represents the hyperparameter tuning results for the hybrid system. It can be seen that the tuned models produced unfavored results compared to the untuned models and hence it was unwise to use these models.

# **Chapter 9**

9.1 Monitoring Procedures for the model

9.2 Risk Assessment Specific and General risk On Industry write in details

9.3 Government Risk Assessment (Local): Keep Blank

9.4 Government Risk Assessment (Global): Keep Blank

# **Chapter 10**

IDC Operators: Keep Blank

IDC Managers: Keep Blank

IDC on AWS: Keep Blank

# **References**

[1] D. Turek and P. Radgen, “Optimized data center site selection—mesoclimatic effects on data center energy consumption and costs,” Energy Efficiency, vol. 14, no. 3, Mar. 2021, doi: <https://doi.org/10.1007/s12053-021-09947-y>.

[2] Enida Sheme, Neki Frasheri, S. Holmbacka, S. Lafond, and Drazen Luzanin, “Datacenters powered by renewable energy: A case study for 60 degrees latitude north,” Sep. 2016, doi: <https://doi.org/10.1109/softcom.2016.7772131>.

[3] I. Goiri, W. Katsak, K. Le, T. D. Nguyen, and R. Bianchini, “Designing and Managing Data centers Powered by Renewable Energy,” IEEE Micro, vol. 34, no. 3, pp. 8–16, May 2014, doi: <https://doi.org/10.1109/mm.2014.6>.

[4] Mohd. R. S. Shaikh, “A Review Paper on Electricity Generation from Solar Energy,” International Journal for Research in Applied Science and Engineering Technology, vol. V, no. IX, pp. 1884–1889, Sep. 2017, doi: <https://doi.org/10.22214/ijraset.2017.9272>.

[5] A. O. M. Maka and J. M. Alabid, “Solar energy technology and its roles in sustainable development,” Clean Energy, vol. 6, no. 3, pp. 476–483, Jun. 2022, doi: <https://doi.org/10.1093/ce/zkac023>.

[6] Z. Gan, “Research on Wind Power Generation Technology in New Energy Power Generation,” IOP Conference Series: Earth and Environmental Science, vol. 651, no. 2, p. 022013, Feb. 2021, doi: <https://doi.org/10.1088/1755-1315/651/2/022013>.

[7] C. Huang, C. Liu, M. Zhong, H. Sun, T. Gao, and Y. Zhang, “Research on Wind Turbine Location and Wind Energy Resource Evaluation Methodology in Port Scenarios,” Sustainability, vol. 16, no. 3, pp. 1074–1074, Jan. 2024, doi: <https://doi.org/10.3390/su16031074>.

[8] R. Yadev and Mr. M. Sharma, “Hybrid Power Generation System Using Solar -Wind Energy: A Review,” *International Journal of Trend in Scientific Research and Development*, vol. Volume-2, no. Issue-3, pp. 941–946, Apr. 2018, doi: <https://doi.org/10.31142/ijtsrd11115>.

[9] A. Rahaman, K. N. Noor, T. A. Abir, S. Rana, and M. Ali, “Design and Analysis of Sustainable Green Data Center with Hybrid Energy Sources,” Journal of Power and Energy Engineering, vol. 09, no. 07, pp. 76–88, 2021, doi: <https://doi.org/10.4236/jpee.2021.97006>.

[10] niloshandj, “Functions of Computer Hardware Components,” Functions of Computer Hardware Components, 2020. <https://computerhardwarecomponentsandfunctions.wordpress.com/>

[11] “Solar energy power generation dataset,” www.kaggle.com. <https://www.kaggle.com/datasets/stucom/solar-energy-power-generation-dataset>

[12] N. Gupta, “Predict the Power(Kw/h) produced by Windmills,” Kaggle.com, 2021. https://www.kaggle.com/datasets/emnikkhil/predict-the-powerkwh-produced-by-windmills (accessed Aug. 25, 2024).

[13] H. Upton, “Wind & Solar Daily Power Production,” Kaggle.com, 2020. https://www.kaggle.com/datasets/henriupton/wind-solar-electricity-production (accessed Aug. 25, 2024).

[14] G. Lawton, “Data Preprocessing: Definition, Key Steps and Concepts,” *SearchDataManagement*, Jan. 2022. <https://www.techtarget.com/searchdatamanagement/definition/data-preprocessing>