# Energy Production In A Combined-cycle Power Plant

**PROJECT TITLE:** “ENERGY PRODUCTION IN

A COMBINED-CYCLE POWER PLANT ”

**MENTOR NAME:** Mr. Karthik Muskula

**GROUP NO.:** 4

**START DATE:** 29 AUG 2023

# Group details

| | | |
|---|---|---|
| MR. SRIHARSHA NOORBHASHA | sriharsha.nsh02@gmail.com | 9959905437 |
| Mr. Akshay haridas rautray | akshayrautray6@gmail.com | 9637637670 |
| Ms. Jayshree Ramesh Bhoyar | jayashribhoyar27@gmail.com | 9146611282 |
| Ms.Devangam Ramya | ramyad7799@gmail.com | 9573448642 |
| Mrs. Afrin Fathima Abulkalam Azath | afrinazath@gmail.com | 971 558554503 (UAE number) |
| Vrushali Laxman Bagul | vrushibagul.96@gmail.com | 9175057012 |
| Ms.Ashwini Eknath Jadhav | jadhavashwini4454@gmail.com | 9373287718 |

# Business Objective:

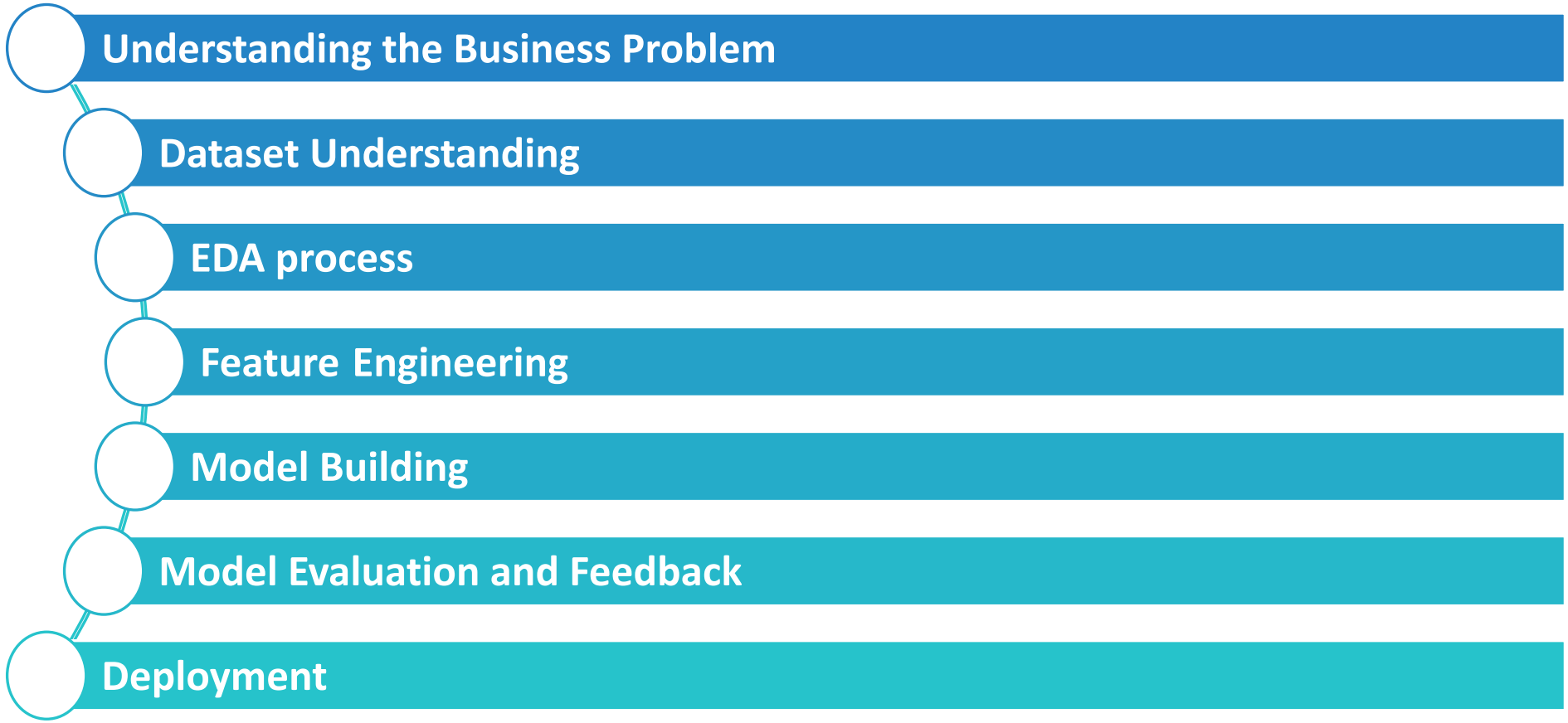
- A combined-cycle power plant comprises gas turbines, steam turbines, and heat recovery steam generators. In this type of plant, the electricity is generated by gas and steam turbines combined in one cycle. Then, it is transferred from one turbine to another. We have to model the energy generated as a function of exhaust vacuum and ambient variables and use that model to improve the plant's performance.

# Business Problem

- **To develop a predictive model to predict full-load power output in a combined-cycle power plant.**
- **Evaluate the performance of the model**

*Since the goal is to predict the output power based on some parameters, this is a* ***regression problem.*** *Regression aims to establish a relationship between predictors (variables that help us make a prediction) and the target (the value we want to predict).*

# Project Architecture / Project Flow



# Data Set Details:

In this project variable to be predicted is **energy production**

The data file contains **9568 observations** with **five variables** collected from a combined cycle power plant over six years when the power plant was set to work with a full load.

***The variables, or features, are the following:***

1. **temperature, in degrees Celsius.**
2. **exhaust_vacuum, in cm Hg.**
3. **amb_pressure, in millibar. (Ambient pressure)**
4. **r_humidity, in percentage. (Relative humidity)**

***Independent Variables***

5. **energy_production, in MW, net hourly electrical energy output.** → ***Dependent Variables***

# Data Set

In [5]:
```
data
```

Out[5]:

| | temperature | exhaust_vacuum | amb_pressure | r_humidity | energy_production |
|---|---|---|---|---|---|
| **0** | 9.59 | 38.56 | 1017.01 | 60.10 | 481.30 |
| **1** | 12.04 | 42.34 | 1019.72 | 94.67 | 465.36 |
| **2** | 13.87 | 45.08 | 1024.42 | 81.69 | 465.48 |
| **3** | 13.72 | 54.30 | 1017.89 | 79.08 | 467.05 |
| **4** | 15.14 | 49.64 | 1023.78 | 75.00 | 463.58 |
| **...** | ... | ... | ... | ... | ... |
| **9563** | 17.10 | 49.69 | 1005.53 | 81.82 | 457.32 |
| **9564** | 24.73 | 65.34 | 1015.42 | 52.80 | 446.92 |
| **9565** | 30.44 | 56.24 | 1005.19 | 56.24 | 429.34 |
| **9566** | 23.00 | 66.05 | 1020.61 | 80.29 | 421.57 |
| **9567** | 17.75 | 49.25 | 1020.86 | 63.67 | 454.41 |

9568 rows × 5 columns

# Exploratory Data Analysis (EDA) and Feature Engineering

- In the given dataset the datatype of the observations in each variables are **floating point datatypes.**
- In data set each features contain **non-null values**.
- There are total **41 duplicate observations** present in dataset, and that has been removed.
- Now the cleaned dataset contain **9527** observation and **5 features**.

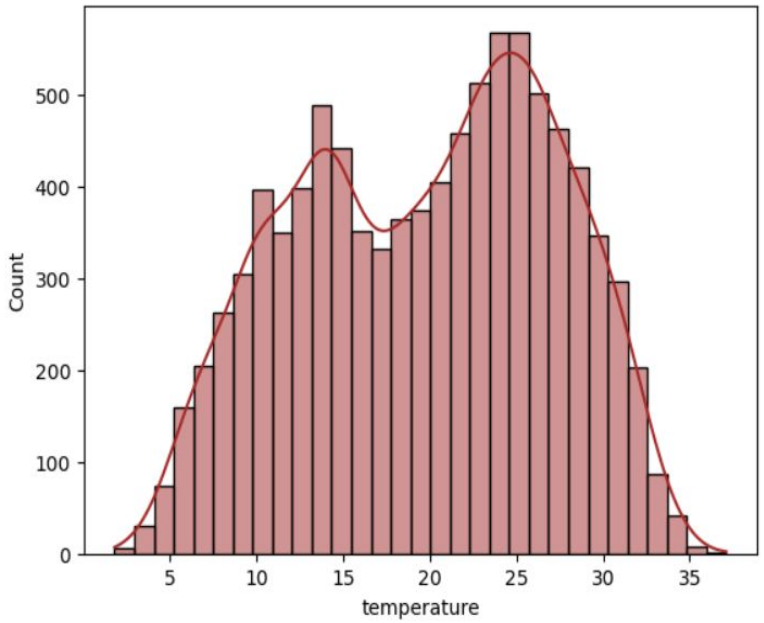
- **From the Statistical Summary**
  - Ambient Temperature in the range 1.81°C and 37.11°C,
  - Ambient Pressure in the range 992.89 – 1033.30 millibar,
  - Relative Humidity in the range of 25.56% to 100.16%
  - Exhaust Vacuum in the range 25.36 – 81.56 cm Hg
  - Net hourly electrical energy output 420.26 – 495.76 MW

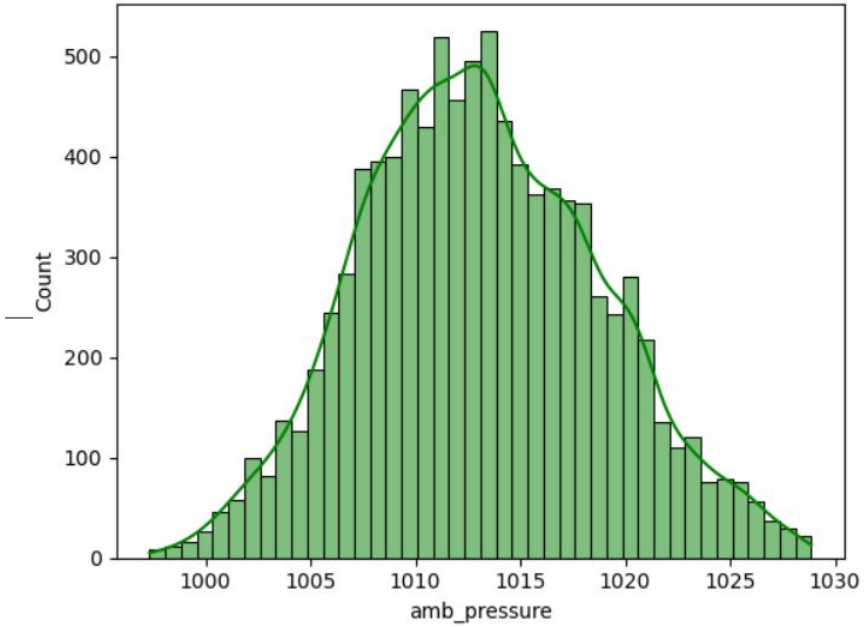
## Data Distribution:

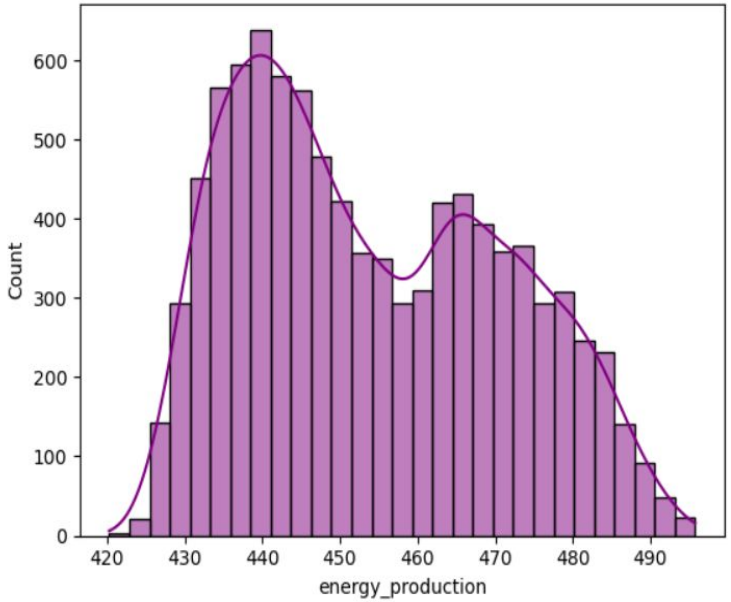
1. temperature and r_humidity – Negatively Skewed or left Skewed
2. exhaust_vacuum, amb_pressure, and energy_production - Positively Skewed or Right Skewed.

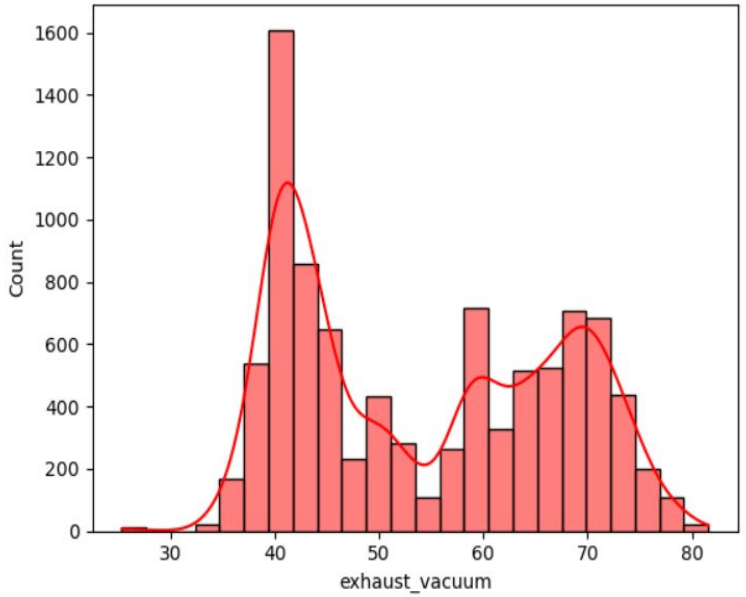
```
data_cleaned1.skew()
```

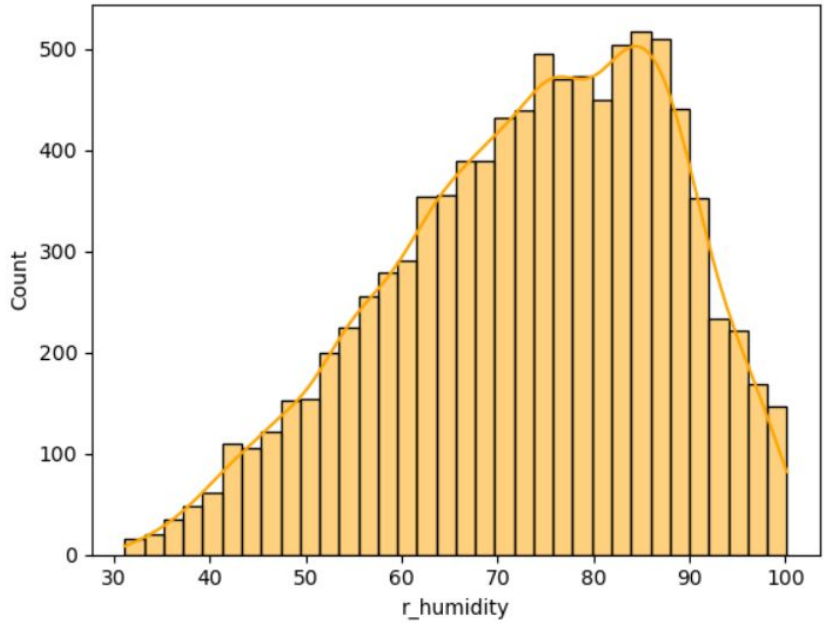
```
temperature         -0.136107
exhaust_vacuum       0.196819
amb_pressure         0.273846
r_humidity          -0.435138
energy_production    0.305791
dtype: float64
```


Count
temperature


Count
amb_pressure

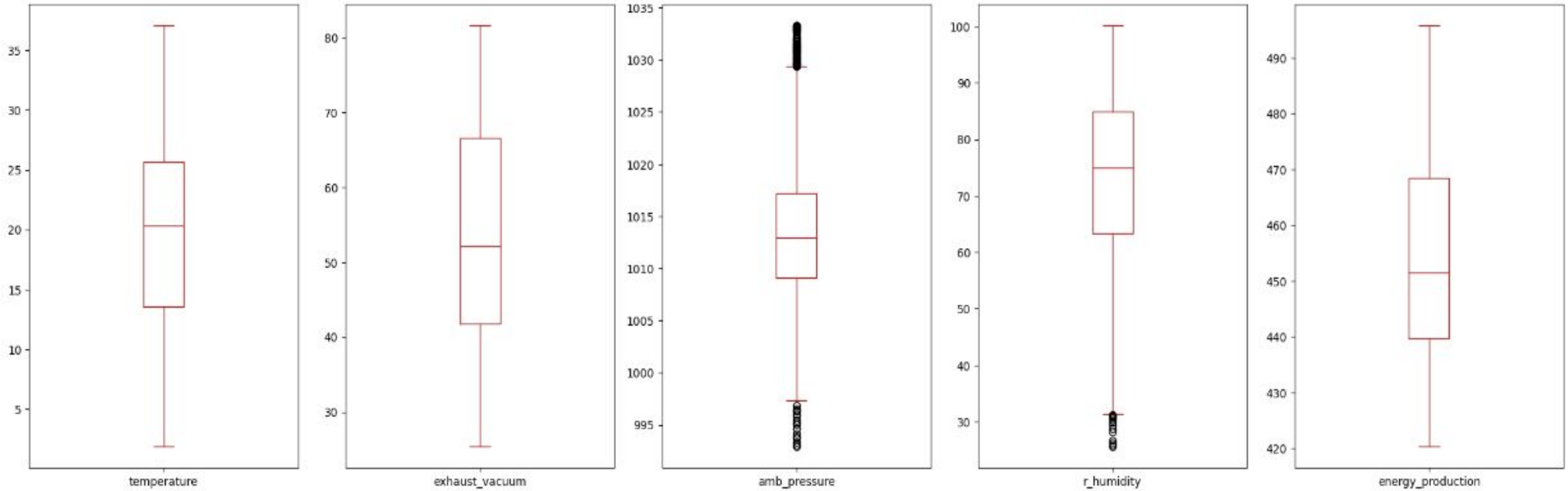

Count
energy_production


Count
exhaust_vacuum


Count
r_humidity

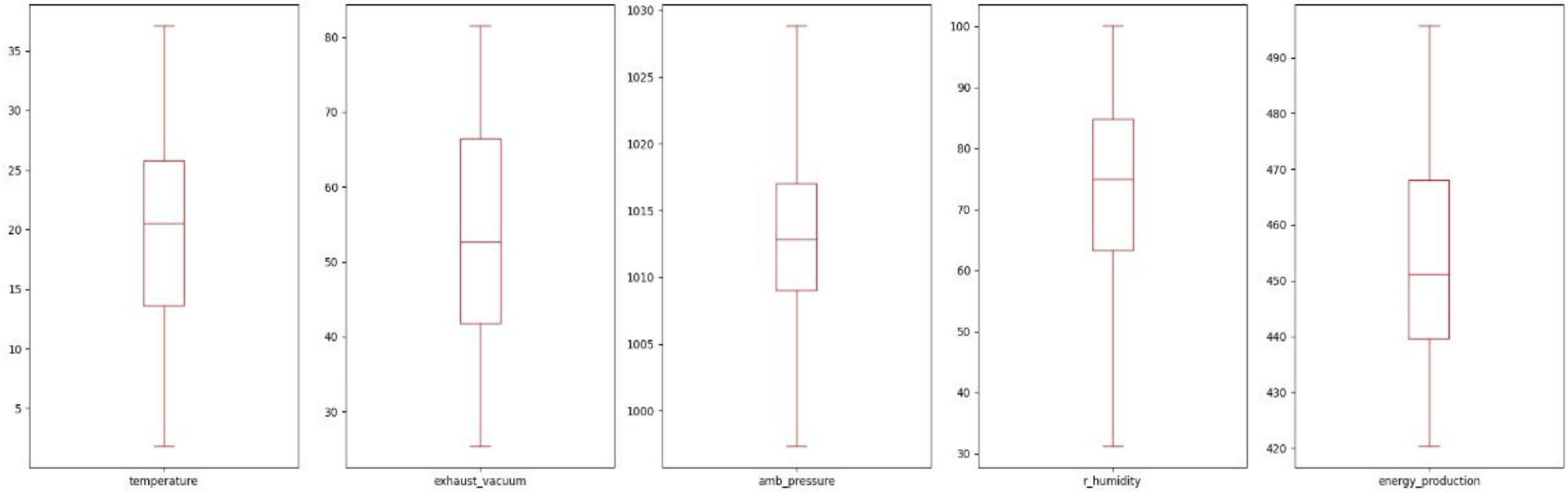
## Outlier Detection:

Outliers were present in **amb_pressure** and **r_humidity**




**Box Plots of Variables before removing the outliers**

**outliers were removed successfully.** Now the dataset contain **9416 observations** and **5 features.**




**Box Plots of Variables after removing the outliers**

# Correlation Matrix

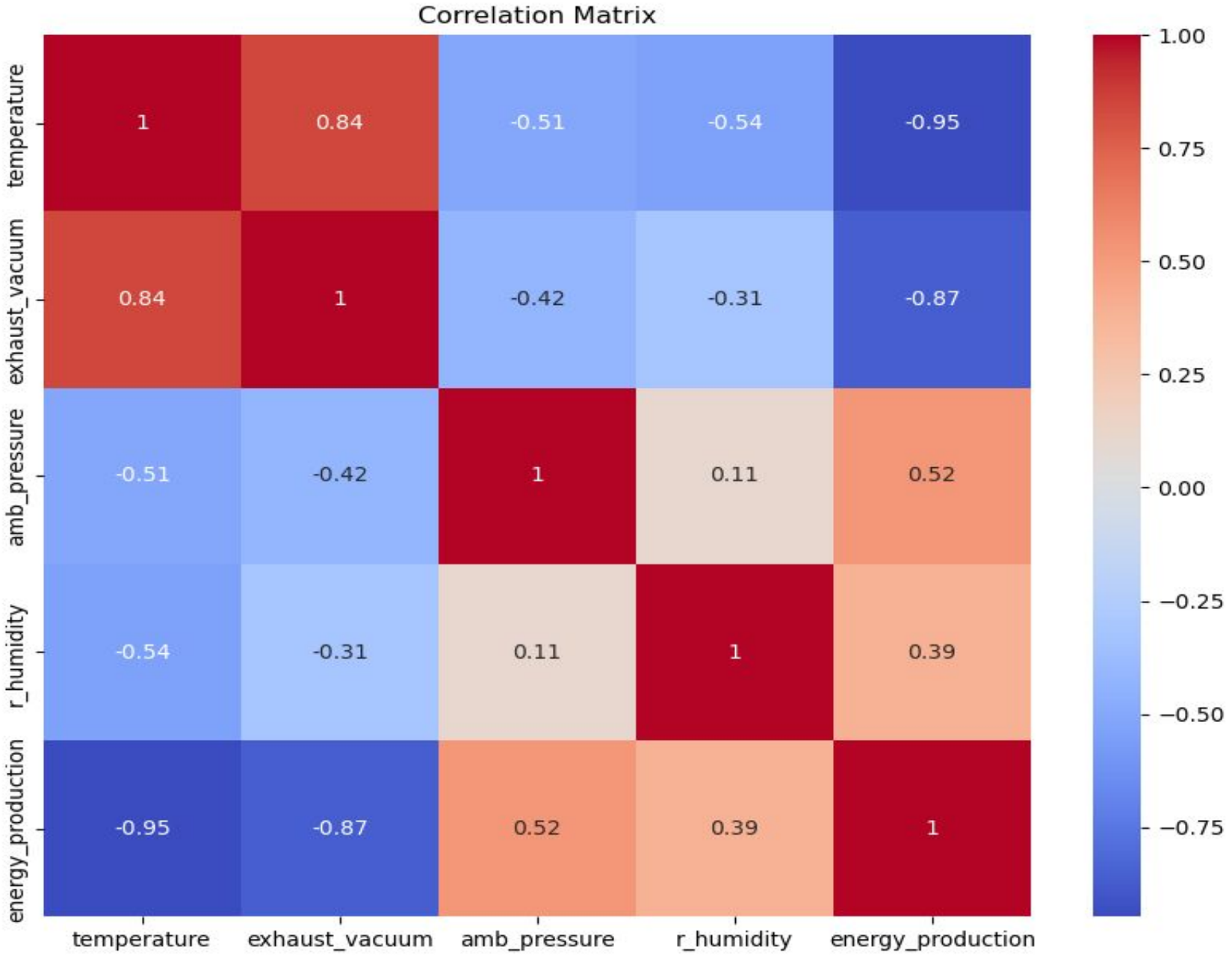
```
ccpp_new.corr()
```

executed in 14ms, finished 11:32:28 2023-09-06

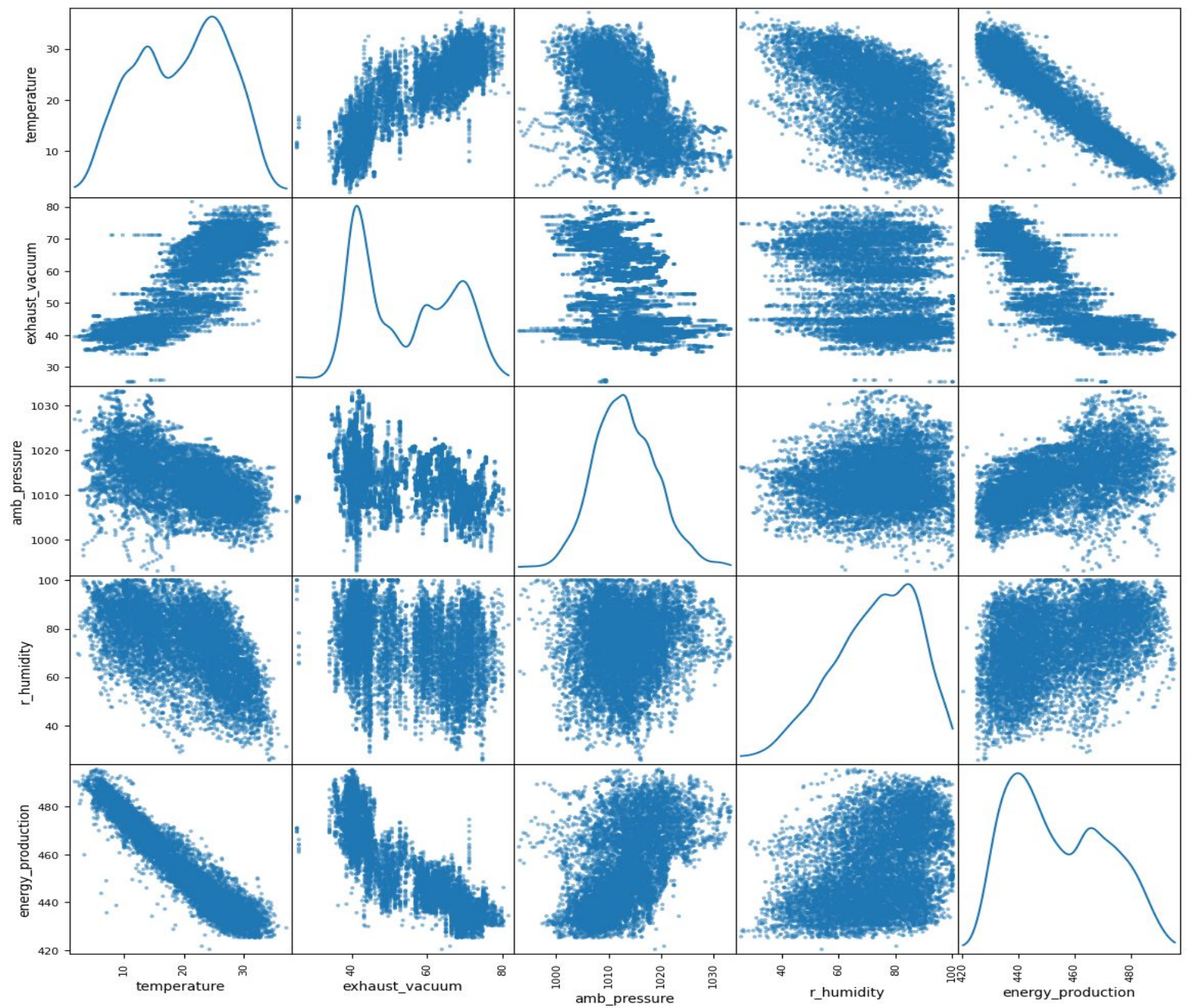
| | temperature | exhaust_vacuum | amb_pressure | r_humidity | energy_production |
|---|---|---|---|---|---|
| **temperature** | 1.000000 | 0.843689 | -0.508222 | -0.543947 | -0.947908 |
| **exhaust_vacuum** | 0.843689 | 1.000000 | -0.415718 | -0.312214 | -0.869900 |
| **amb_pressure** | -0.508222 | -0.415718 | 1.000000 | 0.101631 | 0.518687 |
| **r_humidity** | -0.543947 | -0.312214 | 0.101631 | 1.000000 | 0.391175 |
| **energy_production** | -0.947908 | -0.869900 | 0.518687 | 0.391175 | 1.000000 |

- A **correlation matrix** is a structured approach to ranking the importance of predictors or input variables (input variables that have the most impact) on the output. To do this we plot the heatmap of the correlation matrix using Seaborn.

- *Correlation is measured on a scale of -1 to 1. -1 means complete negative correlation and 1 means complete positive correlation. 0 means no correlation at all.*




- From the correlation matrix, we can see that **temperature and exhaust_vaccuum** have a **strong negative correlation** with the target variable (EP) as their correlation coefficients are **-0.95** and **-0.87** respectively.
- **Amb_pressure and r_humidity** have a **weak positive correlation** with energy_production with correlation coefficients of **0.52** and **0.39**.




When visualized we can easily see that there is a distinctive pattern (negative correlation) seen on temperature and exhaust_vacuum in relation to energy_production.

---

We can also notice that temperature and exhaust_vacuum are highly correlated with each other. This is usually not a good thing as our features should be independent of each other. This problem is called ***multicollinearity.***

*One way of solving this problem is to select the feature(s) that more strongly correlates with our target variable (energy_production). In this case that will be temperature (-0.95). In some cases, we can choose to live with the problem and use our features like that.*

## Normalization:

**Normalization of data is carried out to bring all the observations on same scale.**

## Splitting the data set:

**1. X – independent variables and Y- dependent variable**

**2. Data is splitted in Train and test**

**Train dataset- containing 7532 observations and 4 variables**

**Test dataset- containing 1884 observations and 4 variables**

# Feature Engineering

- The initial observation of the data suggests that we will need to use a linear regression model to predict the power output.
- Though we have obtained a linear model earlier using the statsmodel library, it was a simple mathematical derivation using the data.
- The actual observed value is **481.30**, which is not very far from the predicted value **480.27**
- Now will use the sklearn library, which provides different machine learning models.

**Feature Selection :**

One of the most important steps in creating a machine learning model is to divide the data set into good predictors to train the algorithm on.

# Model Building

Those selected features are trained on below regression models,

---

**1. Linear Regression**

**2. Decision Tree Regression**

**3. Random Forest Regression**

**4. Ridge Regression**

**5. Lasso Regression**

**6. ElasticNet Regression**

**7. XGBoost Regression**

**8. Gradient Boosting Regression**

**9. Support Vector Regression**

**10. AdaBoost Regression**

**11. KNN**

# Model Evaluation

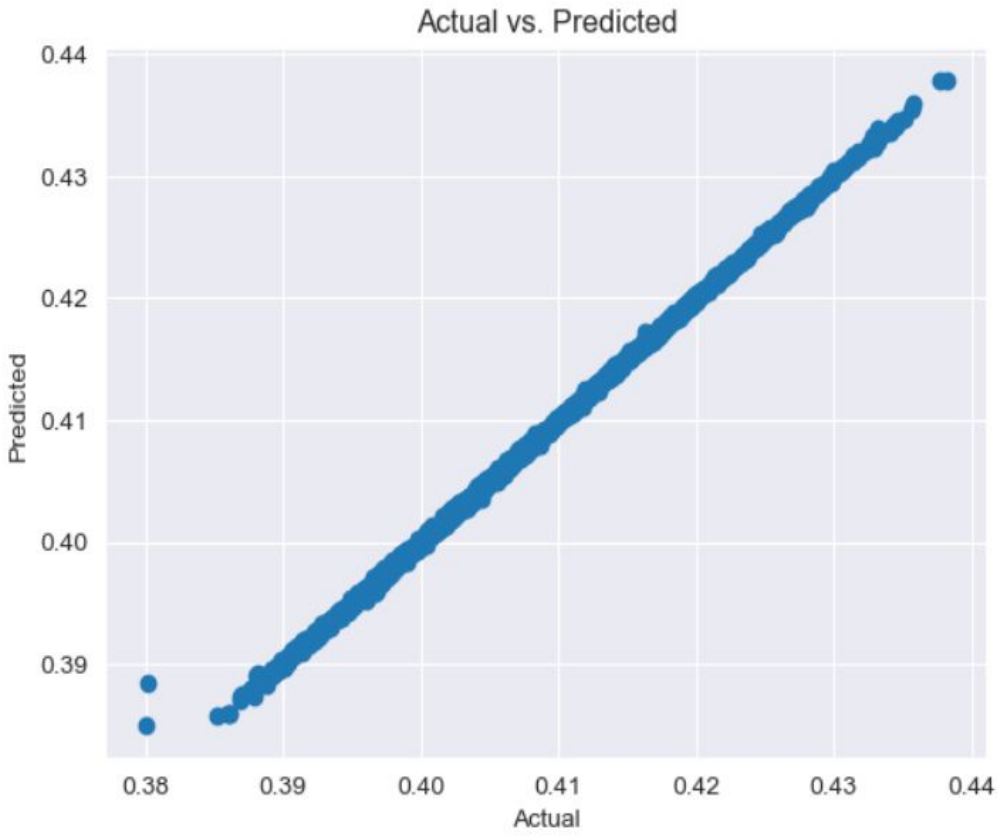
- For Regression problems, there are **3 key performance metrics** that are used to assess how well your model is performing. There are

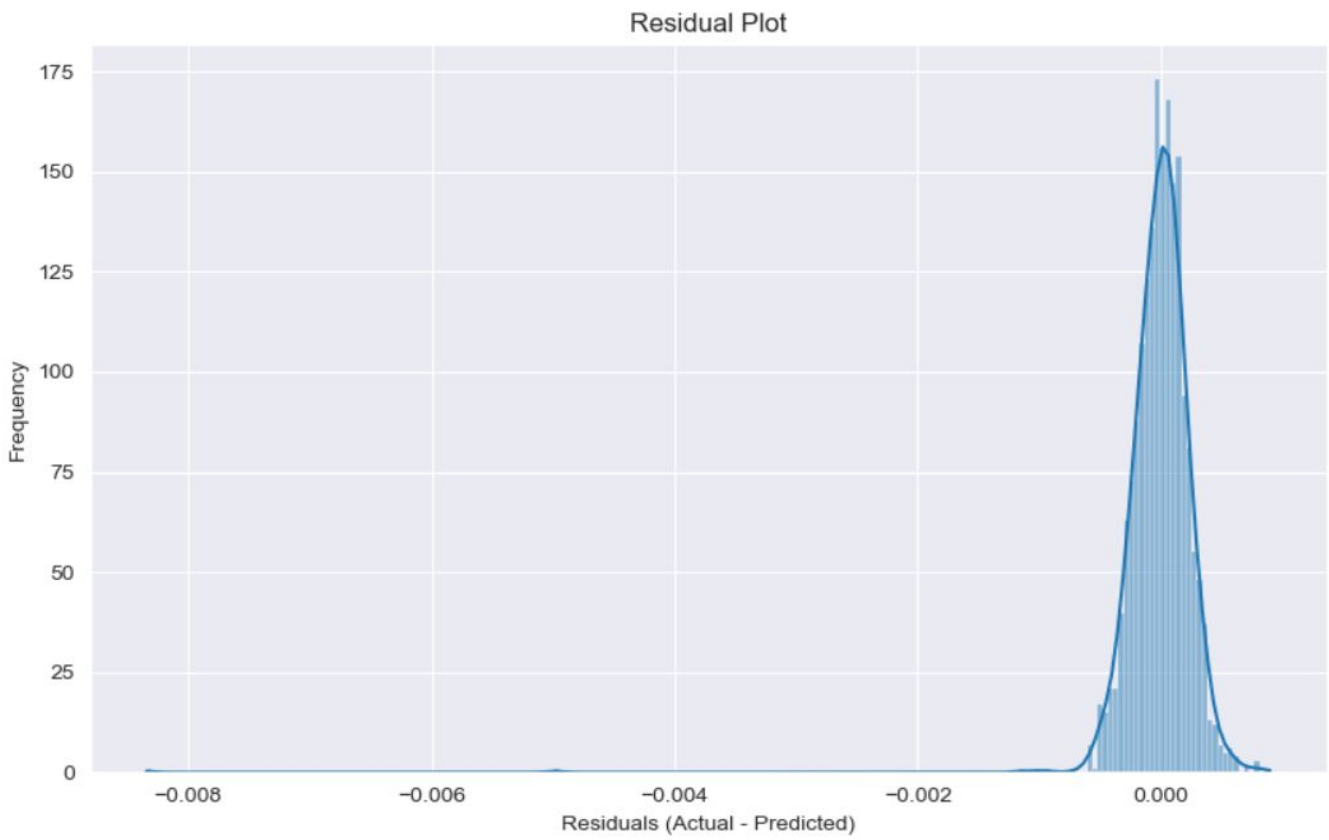
1. **Root Mean Squared Error (RMSE):** measures the average error performed by the model in predicting the outcome for an observation.
2. **R-Squared**: It means how much of the variation in the target variable that can be explained by the set of features used in training the model.
3. **Mean Absolute Error**: measures how far predicted values are away from the actual values.

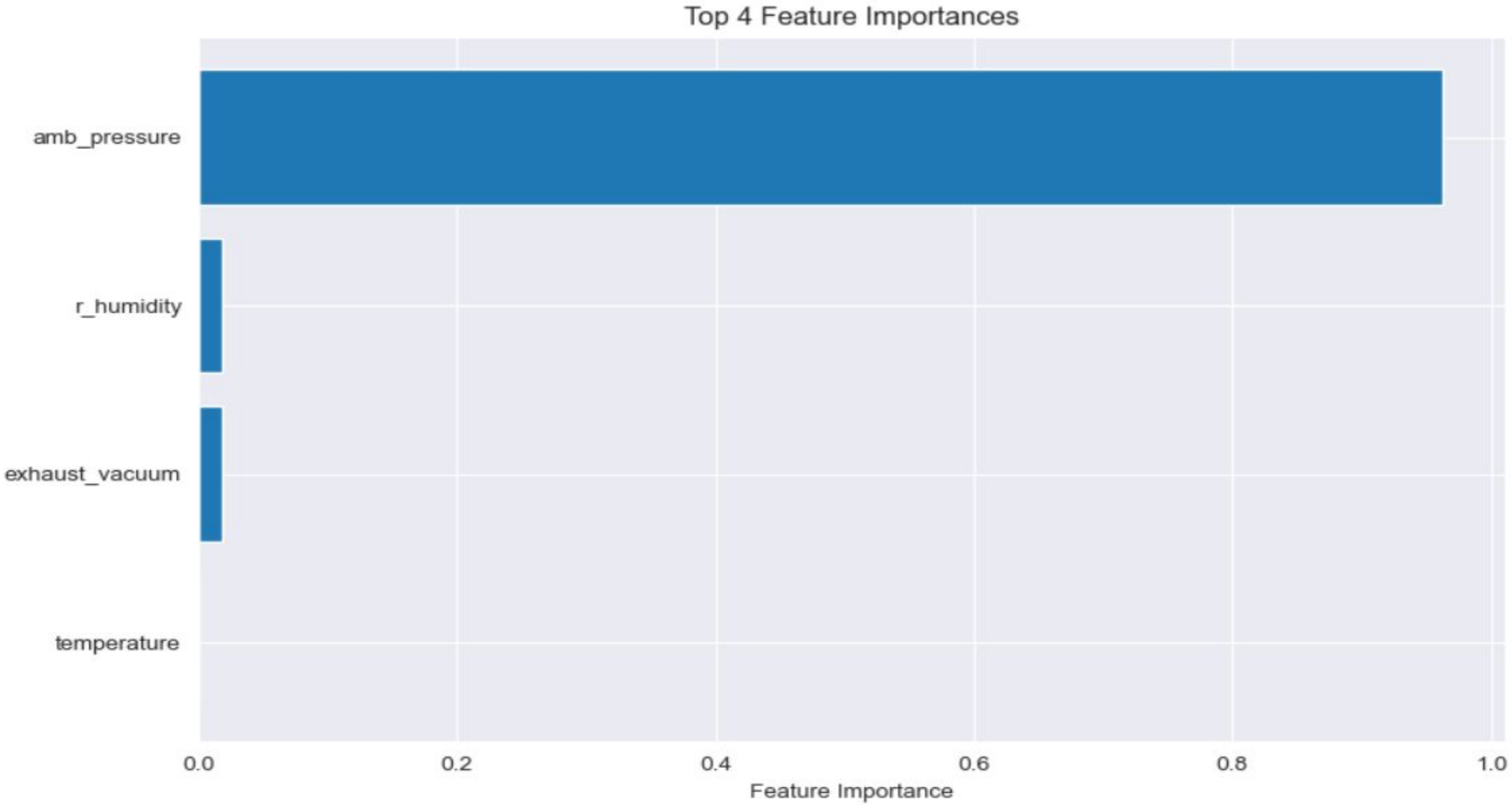
# Performance metrics

| Model Name | Mean Squared Error | R-squared | Mean Absolute Error |
|---|---|---|---|
| Linear | 1.40716e-07 | 0.999008 | 0.000291153 |
| DecisionTree | 2.8402e-07 | 0.997997 | 0.000380482 |
| Random Forest | 1.04044e-07 | 0.999266 | 0.000185098 |
| Ridge | 8.75697e-06 | 0.938247 | 0.002361 |
| Lasso | 0.000141809 | -2.3696e-05 | 0.0102863 |
| Elasticnet | 0.000141809 | -2.3696e-05 | 0.0102863 |
| XGBoost | 1.4989e-07 | 0.998943 | 0.000232755 |
| GBR | 9.45496e-08 | 0.999333 | 0.000170246 |
| SVR | 1.76706e-05 | 0.875389 | 0.00367206 |
| AdaBoost | 2.59826e-06 | 0.981677 | 0.00128764 |
| KNN | 1.22613e-06 | 0.991353 | 0.000790051 |

# Plotted the value from Gradient Boosting Regression




Top 4 Feature Importances
amb_pressure
r_humidity
exhaust_vacuum
temperature
0.0
0.2
0.4
0.6
0.8
1.0
Feature Importance

# Model Results

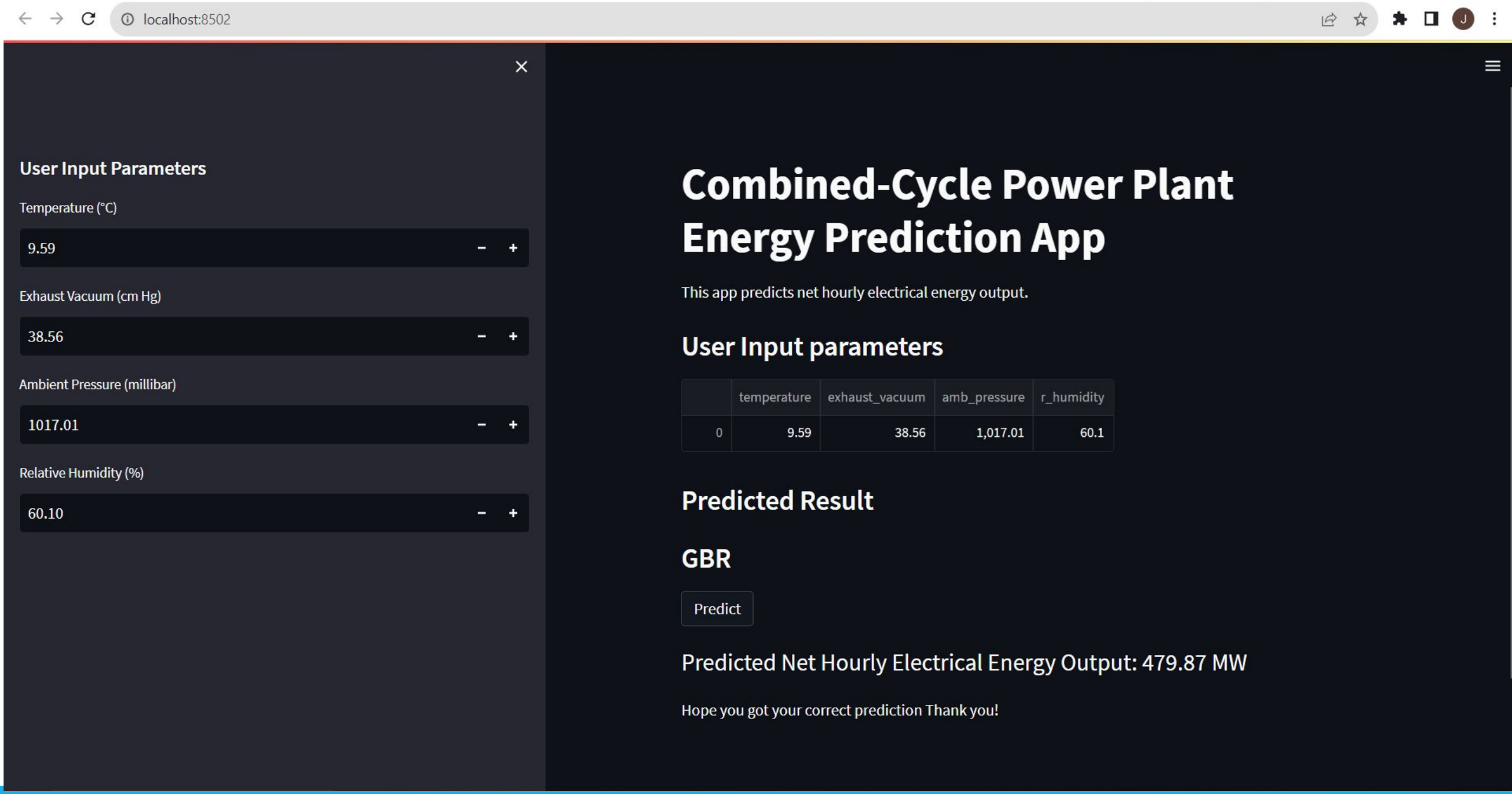
- Now that we have evaluated all our models, we can see that the **Gradient Boosting Regression** Algorithm (with all features) gave us the best performance.

- The **R-squared is 0.999333** (which means that **99.93%** of the variation in the target variable can be explained by the model).

- Similarly, it yields the lowest MSE of **9.45496e-08**.

- We can conclude from our results that **Gradient Boosting Regression** should be selected as best regression model to predict future values.

# Deployment of Model




**Deployment of Gradient Boosting Regression model by using Streamlit.**

# Challenges

- **Removing the outliers** :Outliers can negatively affect the performance of machine learning models by introducing noise or causing convergence issues in the model performance.
- **Feature Selection:** Choosing the right variables (features) to include in our model was crucial. Selecting irrelevant or redundant features could negatively impact the model's performance.
- **Model Selection:** Deciding which Regression algorithm to use for predicting energy production was a challenging task. Different models have different strengths and weaknesses.
- **Hyperparameter Tuning:** finding the best set of hyperparameters for a given algorithm or model to achieve optimal performance.
- **Deployment:** Integrating our model using streamlit, ensuring it runs reliably in real-time, and make it user friendly.

# References


1. https://pandas.pydata.org/
2. https://numpy.org/doc/
3. https://matplotlib.org/
4. https://seaborn.pydata.org/
5. https://scikit-learn.org/
6. https://www.statsmodels.org/

# Thank You