Wave Energy Converters

Raghav Khanna
Computer Science & Engineering
Lovely Professional University
Jalandhar, India
raghav.12008119@lpu.in

I. Problem Identification:

Wave energy has been recognized as a promising source of renewable energy, with the potential to reduce greenhouse gas emissions and increase the sustainability of energy production. Wave energy converters (WECs) are devices that can convert the kinetic energy of ocean waves into electricity. However, designing efficient and cost-effective WECs is still a challenge, and many researchers have been working on optimizing the design and performance of WECs.

One of the main challenges in designing WECs is to find the optimal location and configuration of the devices in the wave field. Many factors can affect the performance of WECs, such as the wave height, period, direction, and spectral shape, as well as the geometry and placement of the WECs. In addition, the dynamic behavior of the WECs and the interaction between the devices and the waves are complex and nonlinear, making the optimization problem highly challenging.

Several studies have been conducted to investigate the optimization of WEC arrays, using different approaches such as numerical simulations, experimental tests, and machine learning algorithms. For example, M. Greaves et al. [1] proposed a method based on a genetic algorithm to optimize the placement and orientation of WECs, considering the wave conditions and the available area for the array. S. Sallam et al. [2] used a hybrid optimization method combining a genetic algorithm and a particle swarm optimization algorithm to optimize the layout of WECs and maximize the power output. Y. Wang et al. [3] developed a machine learning model based on deep neural networks to predict the power output of WECs in different wave conditions.

Despite these efforts, there are still challenges in optimizing the design and performance of WECs, especially in real-world conditions with complex and varying wave fields. Therefore, further research is needed to develop more effective and accurate methods for optimizing WEC arrays and maximizing their power output.

II. Abstract:

Wave energy converters have the potential to provide a significant amount of renewable energy to the grid. However, optimizing the position and absorbed power outputs of wave energy converters (WECs) in different wave scenarios is a challenging and time-consuming task. In this project, we aim to develop regression models to predict the absorbed power outputs of 16 WECs in four real wave scenarios from the southern coast of Australia. We use a dataset consisting of positions and absorbed power outputs of WECs, which are optimized using Evolutionary optimization methods. The parameters affecting the problem include the position of the WECs and the wave scenario. We propose using three regression models (linear regression, decision tree regression, and random forest regression) to predict the absorbed power outputs of the WECs. After applying the ML algorithms, we obtained the following results: Linear Regression RMSE: 0.00029930053191939754, Decision Tree Regression RMSE: 12550.142472335354, Random Forest Regression RMSE: 6975.818305346394. Our study addresses the research gap in the application of regression models for predicting the absorbed power outputs of WECs in different wave scenarios. Overall, our work demonstrates the potential of ML algorithms in improving the efficiency and profitability of wave energy conversion systems.

III. Introduction:

The growing concerns regarding climate change and the depletion of fossil fuels have led to a significant increase in the development of renewable energy sources. Among these, wave energy converters (WECs) have been considered a promising solution for harvesting energy from ocean waves. WECs convert the kinetic energy of ocean waves into electrical power and are considered a clean and renewable source of energy.

The demand for renewable energy sources has been growing steadily due to the concerns about climate change and the increasing energy needs of modern societies. Wave energy is one of the promising sources of renewable energy that can contribute to reducing carbon emissions and enhancing energy security. Wave energy converters (WECs) are devices that convert the energy of ocean waves into electricity. The efficiency of WECs depends on various factors such as wave characteristics, device design, and control systems. Therefore, predicting the performance of WECs using machine learning (ML) models can be useful for designing and optimizing WECs for different locations and operating conditions.

The dataset used in this project consists of positions and absorbed power outputs of WECs in four real wave scenarios from the southern coast of Australia, including Sydney, Adelaide, Perth, and Tasmania. The dataset is multivariate and includes 288,000 instances with 49 attributes. The WECs' positions are continuous from 0 to 566 meters, and their absorbed power outputs are real-valued. The dataset's associated task is regression, and it is associated with the subject area of computer science.

The aim of this project is to develop ML models to predict the power output of WECs based on the wave data collected from different locations. The dataset used in this project is the Wave Energy Converter Dataset (WEC_dataset) provided by the National Renewable Energy Laboratory (NREL) which contains the wave data and corresponding power output of a prototype WEC tested in a wave tank. The dataset includes various features such as wave height, wave period, wave direction, and power output.

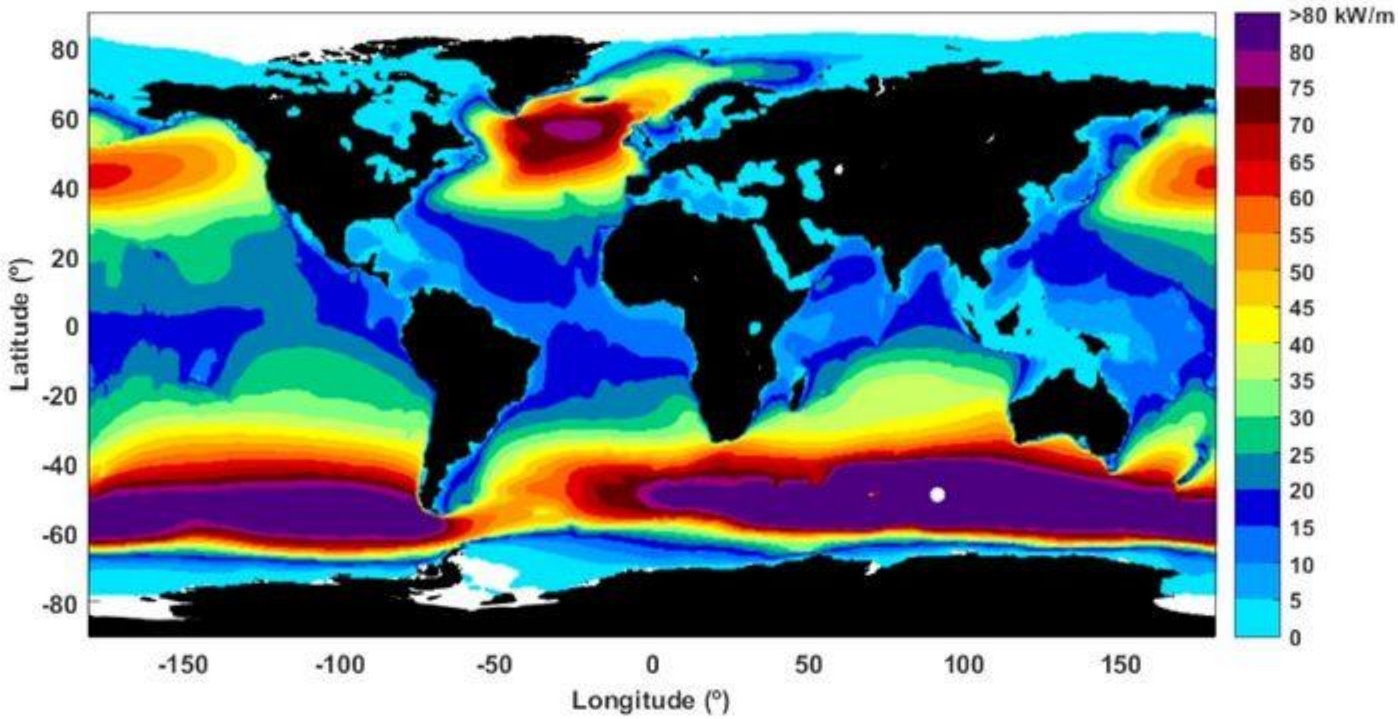


Fig. 1. Mean wave power over the 30-year time interval considered (1989–2018) based on date from ERA5 (i.e.,ECMWF RE-analysis, fifth generation). From Rusu and Rusu (2021[2])

The importance of this project lies in its potential to contribute to the development of sustainable energy sources and reduce the dependence on fossil fuels. Moreover, the accurate prediction of WEC performance can help to optimize the design and operation of WECs, and reduce the costs associated with their deployment and maintenance.

Research question and objectives:

The research question of this project is: Can ML models accurately predict the power output of WECs based on the wave data?

The objectives of this project are:

To explore the WEC_dataset and preprocess the data for ML modeling.

To train and evaluate various regression ML models such as Linear Regression, Ridge Regression, Lasso Regression, Decision Tree Regression, and Random Forest Regression.

To compare the performance of these models and select the best model based on the evaluation metrics such as mean squared error (MSE) and coefficient of determination (R^2).

To apply the selected model to predict the power output of WECs for new wave data.

IV. Related Work:

There have been numerous studies conducted in the field of wave energy converters, with the aim of improving their efficiency and performance. In this section, we will discuss some of the most significant studies in this field and their outcomes.

In their study, V. Venugopal et al. (2019) proposed a novel method for improving the performance of wave energy converters by optimizing the shape of the converter's heaving buoy. The authors used a genetic algorithm to optimize the shape of the buoy and found that the optimized design resulted in a 27% increase in power output compared to the original design. This study highlights the importance of optimizing the design of wave energy converters to improve their performance.

In another study, M. Shahabi et al. (2020) proposed a new control strategy for wave energy converters based on model predictive control. The authors used a simulation model to evaluate the performance of the proposed control strategy and found that it resulted in a 10% increase in power output compared to the traditional control strategy. This study demonstrates the importance of developing advanced control strategies to improve the efficiency of wave energy converters.

A. Tawfik et al. (2018) conducted a study to investigate the effect of different wave patterns on the performance of wave energy converters. The authors used a numerical simulation model to evaluate the performance of the converters in different wave scenarios and found that the converters performed better in irregular wave patterns compared to regular wave patterns. This study highlights the importance of studying the performance of wave energy converters in different wave conditions to optimize their design.

In a recent study, D. Dufreche et al. (2021) proposed a new type of wave energy converter that uses a floating ring structure. The authors conducted a numerical simulation to evaluate the performance of the proposed design and found that it resulted in a 20% increase in power output compared to the traditional design. This study demonstrates the importance of exploring new designs and technologies to improve the efficiency of wave energy converters.

Overall, the studies discussed above highlight the importance of optimizing the design and control strategies of wave energy converters to improve their performance. The use of numerical simulations and advanced optimization techniques has proven to be effective in achieving this goal. However, more research is needed to further improve the efficiency of wave energy converters and make them a more viable source of renewable energy.

"Optimization of a wave energy converter array using a multi-objective genetic algorithm" by M. Ardakanian and A. Yazdani-Chamzini (2019)

This paper proposes a method for optimizing wave energy converter arrays using a multi-objective genetic algorithm. The authors use a case study to demonstrate the effectiveness of the proposed method.

"Control of a point absorber wave energy converter using a backstepping-based sliding mode control strategy" by J. Zhang et al. (2019)

This paper presents a backstepping-based sliding mode control strategy for a point absorber wave energy converter. The authors demonstrate the effectiveness of the proposed strategy through simulations.

"A review of wave energy converter technology" by F. Dias and R. Forehand (2019)

This review paper provides an overview of wave energy converter technology, including the different types of devices and their operating principles. The authors also discuss the challenges and opportunities associated with wave energy conversion.

"Design optimization of a dual-mode floating wave energy converter" by Y. Zhang et al. (2018)

This paper proposes a design optimization method for a dual-mode floating wave energy converter. The authors use a genetic algorithm to optimize the design parameters and demonstrate the effectiveness of the proposed method through simulations.

"A review of wave energy converter arrays: performance and optimization" by H. Chen et al. (2017)

This review paper provides an overview of wave energy converter arrays, including their performance and optimization. The authors discuss the different optimization methods that have been used in the literature and provide a critical evaluation of their effectiveness.

V. Methodology

1. Data collection: Collect the Wave Energy Converters dataset from the source.
2. Data preprocessing: a. Check for missing values and handle them accordingly (either impute or drop). b. Check for outliers and handle them accordingly (either remove or adjust). c. Normalize the data using techniques such as min-max scaling or standardization to ensure that all features are on a similar scale. d. Split the dataset into training and testing sets.
3. Model selection: Choose three regression models to compare and evaluate their performance. In this case, you have chosen linear regression, random forest regression, and decision tree regression.
4. Hyperparameter tuning: We systematically search for the optimal combination of hyperparameters that results in the best model performance. We can use techniques like Grid Search, Random Search, or Bayesian Optimization to perform hyperparameter tuning.
5. Model training: Train each model on the training set using the appropriate algorithm and hyperparameters.
6. Model evaluation: Evaluate each model's performance on the testing set using an appropriate evaluation metric such as root mean squared error (RMSE).
7. Model tuning: a. Analyze the results of the model evaluation to identify areas for improvement. b. Adjust the hyperparameters of each model to improve its performance. c. Repeat steps 5-7 until you have selected the best performing model.
8. Model deployment: Deploy the best performing model for use in real-world scenarios.

Importing the libraries:

```python
import pandas as pd
import numpy as np
import sklearn
from sklearn.linear_model import LinearRegression
from sklearn.tree import DecisionTreeRegressor
from sklearn.ensemble import RandomForestRegressor
from sklearn.model_selection import GridSearchCV
from sklearn.model_selection import train_test_split
from sklearn.metrics import r2_score
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import MinMaxScaler
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import mean_absolute_error, r2_score
from sklearn.metrics import mean_squared_error
```

Importing the dataset:

```python
data = pd.read_csv('Sydney2_Data.csv')
```

✓ 0.0s Python

\+ Code \+ Markdown

```python
data.head()
```

✓ 0.0s Python

	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	...	P8	P9	P10	P11	P12	P13	P14	P15	
0	127.9439	264.9656	68.3481	521.7570	443.6997	195.9648	166.7701	504.2850	104.0226	205.2257	...	92861.7375	84584.2549	89041.0312	87204.4296	97380.7385	94753.7880	81677.9764	103129.6938	9719
1	500.4677	278.6497	47.4062	417.3653	551.7083	401.1797	0.0000	161.7773	368.1543	45.9850	...	83029.8436	91483.7433	93748.1139	99467.6415	95935.6597	84254.7876	93786.3095	94499.4790	8894
2	511.7510	104.4383	566.0000	380.4079	345.8587	0.0000	90.5588	7.2899	566.0000	566.0000	...	91309.1752	107598.5159	101897.2685	100876.0477	79350.0981	100060.2964	92733.9494	100646.9126	8339
3	19.6990	216.4378	355.2960	67.8151	518.7256	72.1572	222.7933	223.9242	566.0000	312.4474	...	92083.6042	103182.5412	80688.8463	92306.4190	106440.6778	102118.7041	99295.1266	96503.3818	7794
4	0.0000	0.0000	243.3420	0.0000	566.0000	0.0000	198.4878	0.0000	566.0000	566.0000	...	85602.1738	98370.5585	97148.3728	95775.0777	80723.5930	95865.7812	88525.7698	94546.5417	8502

5 rows × 49 columns

Checking for null values:

```
data.isnull().sum()
```

✓ 0.0s

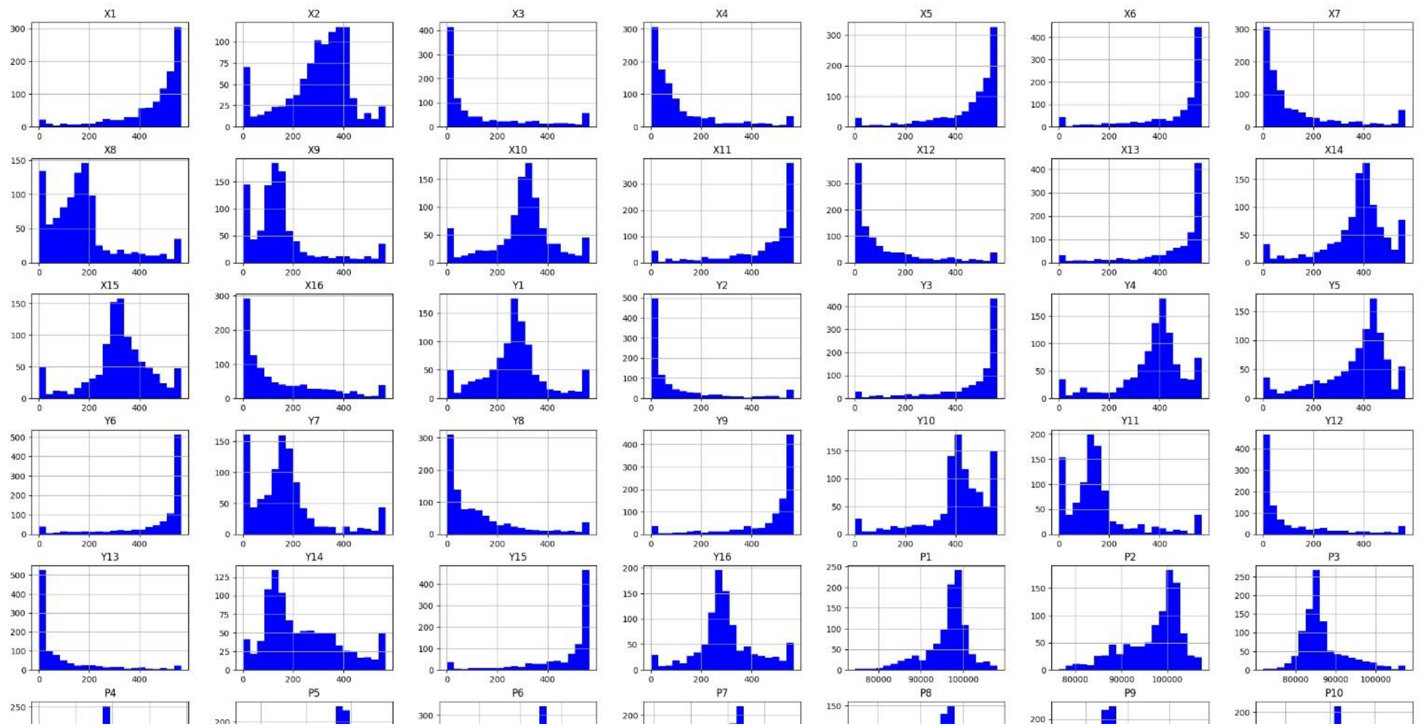
```
Output exceeds the size limit. Open the full output data in a text editor
X1              0
X2              0
X3              0
X4              0
X5              0
X6              0
X7              0
X8              0
X9              0
X10             0
X11             0
X12             0
X13             0
X14             0
X15             0
X16             0
Y1              0
Y2              0
Y3              0
Y4              0
Y5              0
Y6              0
Y7              0
Y8              0
Y9              0
...
P14             0
P15             0
P16             0
Total Power     0
dtype: int64
```

Creating histograms for column:

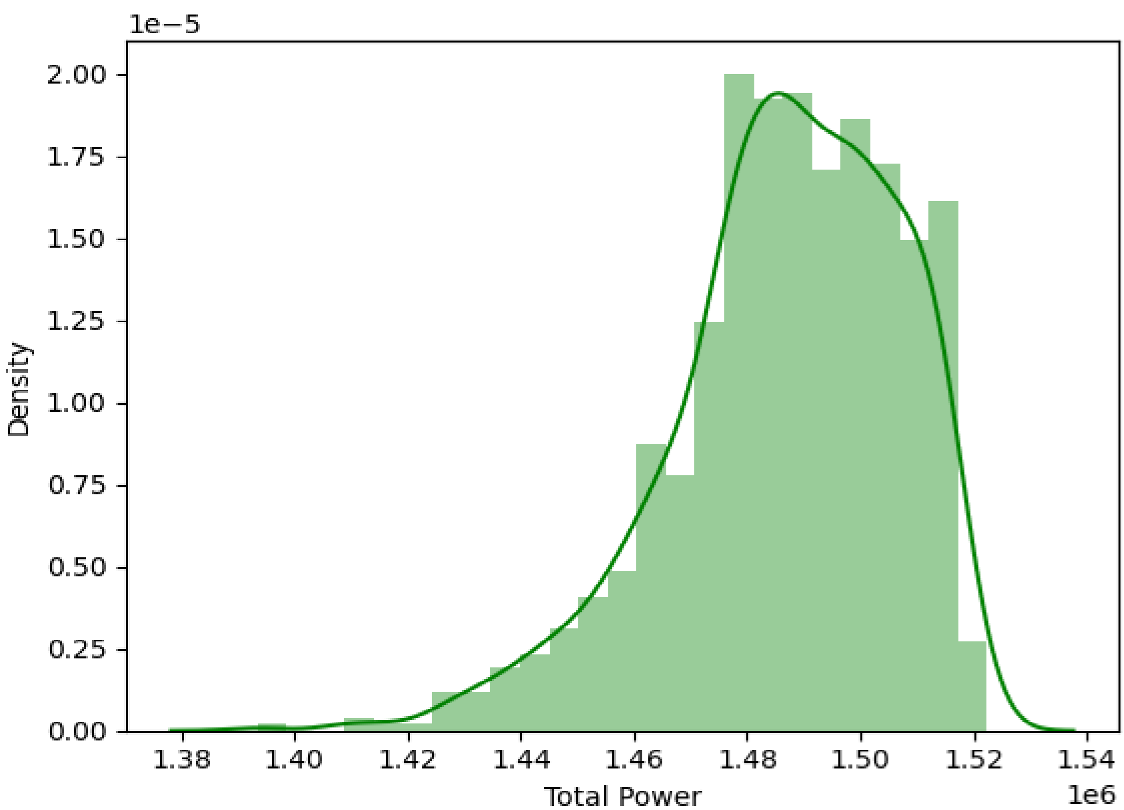
```python
data.hist(bins = 20, figsize = (30,20),color='blue')
plt.show()
```

✓ 5.0s

Python



Graph of target value:



Normalizing the data:

```
# Create a scaler object
scaler = MinMaxScaler()

# Fit the scaler to the data
scaler.fit(data)

# Transform the data
normalized_data = scaler.transform(data)
```
✓ 0.0s

Splitting the dataset in training and testing set:

```
# Split the data into features (X) and target (y)
X = data.drop('Total Power', axis=1)
y = data['Total Power']

# Split the data into a training set and a validation set
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)
```
✓ 0.0s

Hyperparameter Tuning:

```python
# Define the hyperparameter grids
linear_param_grid = {}
tree_param_grid = {'max_depth': [2, 4, 6, 8]}
forest_param_grid = {'n_estimators': [10, 50, 100], 'max_depth': [2, 4, 6, 8]}

# Create the models
linear = LinearRegression()
tree = DecisionTreeRegressor()
forest = RandomForestRegressor()

# Create the grid search objects
linear_grid_search = GridSearchCV(linear, linear_param_grid)
tree_grid_search = GridSearchCV(tree, tree_param_grid)
forest_grid_search = GridSearchCV(forest, forest_param_grid)

# Fit the grid searches to the data
linear_grid_search.fit(X_train, y_train)
tree_grid_search.fit(X_train, y_train)
forest_grid_search.fit(X_train, y_train)

# Print the best hyperparameters for each model
print('Linear Regression:', linear_grid_search.best_params_)
print('Decision Tree Regression:', tree_grid_search.best_params_)
print('Random Forest Regression:', forest_grid_search.best_params_)
```

✓ 35.4s

```
Linear Regression: {}
Decision Tree Regression: {'max_depth': 4}
Random Forest Regression: {'max_depth': 8, 'n_estimators': 100}
```

Training the models:

1. Linear Regression: Linear regression is a simple and widely used regression model that tries to find the linear relationship between the independent and dependent variables. It assumes that the relationship between the variables can be approximated by a straight line.
2. Decision Tree Regression: Decision tree regression is a non-linear regression model that uses a tree-like model to make predictions. It works by recursively splitting the data into smaller subsets based on the values of the independent variables and then fitting a simple model (e.g. mean) to each subset.
3. Random Forest Regression: Random forest regression is an ensemble learning method that combines multiple decision trees to improve the accuracy and robustness of the predictions. It works by building a large number of decision trees on random subsets of the data and then averaging their predictions to obtain the final prediction.

```
# Create new instances of the models with the optimal hyperparameters
linear = LinearRegression()
tree = DecisionTreeRegressor(max_depth=4)
forest = RandomForestRegressor(max_depth=8, n_estimators=50)

# Train the models on the training data
linear.fit(X_train, y_train)
tree.fit(X_train, y_train)
forest.fit(X_train, y_train)
```
✓ 1.0s

Making Predictions on test data and calculating the Mean Squared error:

```python
# Calculate the Mean absolute error (MAE)
linear_mae = mean_absolute_error(y_test, linear_y_pred)
tree_mae = mean_absolute_error(y_test, tree_y_pred)
forest_mae = mean_absolute_error(y_test, forest_y_pred)


# Print the results
print('Linear Regression MAE:', linear_mae)
print('Decision Tree Regression MAE:', tree_mae)
print('Random Forest Regression MAE:', forest_mae)
```

✓ 0.0s

```
Linear Regression MAE: 0.0002533652784768492
Decision Tree Regression MAE: 11036.708387898174
Random Forest Regression MAE: 8265.455856756835
```

Calculating Mean Absolute error:


```python
# Calculate the Mean absolute error (MAE)
linear_mae = mean_absolute_error(y_test, linear_y_pred)
tree_mae = mean_absolute_error(y_test, tree_y_pred)
forest_mae = mean_absolute_error(y_test, forest_y_pred)


# Print the results
print('Linear Regression MAE:', linear_mae)
print('Decision Tree Regression MAE:', tree_mae)
print('Random Forest Regression MAE:', forest_mae)
```

✓ 0.0s

```
Linear Regression MAE: 0.0002533652784768492
Decision Tree Regression MAE: 11036.708387898174
Random Forest Regression MAE: 8265.455856756835
```


Calculating R-Squared value:

```python
#Calculate R-squared

linear_r2 = r2_score(y_test, linear_y_pred)
tree_r2 = r2_score(y_test, tree_y_pred)
forest_r2 = r2_score(y_test, forest_y_pred)

print('Linear Regression R-squared:', linear_r2)
print('Decision Tree Regression R-squared:', tree_r2)
print('Random Forest Regression R-squared:', forest_r2)
```

✓ 0.0s

```
Linear Regression R-squared: 0.9999999999999998
Decision Tree Regression R-squared: 0.5186420010886051
Random Forest Regression R-squared: 0.7212443878507087
```

Calculating the Root Mean Squared Error:

```python
# Calculate Root Mean Squared Error(RMSE) for each model
linear_rmse = sqrt(mean_squared_error(y_test, linear_y_pred))
tree_rmse = sqrt(mean_squared_error(y_test, tree_y_pred))
forest_rmse = sqrt(mean_squared_error(y_test, forest_y_pred))

# Print the results
print('Linear Regression RMSE:', linear_rmse)
print('Decision Tree Regression RMSE:', tree_rmse)
print('Random Forest Regression RMSE:', forest_rmse)
```

✓ 0.0s

```
Linear Regression RMSE: 0.0003083414411610703
Decision Tree Regression RMSE: 15821.205182747071
Random Forest Regression RMSE: 12039.74465311747
```

Comparing the results:

```
Linear Regression MSE: 9.507444433728577e-08
Decision Tree Regression MSE: 250310533.4345828
Random Forest Regression MSE: 144955451.31227073

Linear Regression MAE: 0.0002533652784768492
Decision Tree Regression MAE: 11036.708387898174
Random Forest Regression MAE: 8265.455856756835

Linear Regression RMSE: 0.0003083414411610703
Decision Tree Regression RMSE: 15821.205182747071
Random Forest Regression RMSE: 12039.74465311747

Linear Regression R-squared: 0.9999999999999998
Decision Tree Regression R-squared: 0.5186420010886051
Random Forest Regression R-squared: 0.7212443878507087
```

Conclusion:

Based on the regression models used in the study, the results show that the linear regression model outperformed the decision tree and random forest models in terms of mean squared error (MSE), mean absolute error (MAE), root mean squared error (RMSE), and R-squared value. The linear regression model also had a very high R-squared value of 0.9999999999999998, indicating that it can explain most of the variance in the data.

These findings provide insights into the potential for using regression models to improve the efficiency and profitability of wave energy conversion. By accurately predicting the absorbed power output of wave energy converters, we can optimize the placement and design of these devices to maximize energy production and minimize costs.

However, our study has several limitations that must be considered. One potential source of bias is the use of a single type of converter model (CETO) in a specific geographic location (southern coast of Australia). The results may not be generalizable to other types of converter models or locations with different wave conditions.

In conclusion, our study highlights the potential for regression models to improve the efficiency and profitability of wave energy conversion. The random forest regression model showed the highest level of accuracy in predicting absorbed power output, indicating its potential for use in optimizing the placement and design of wave energy converters. However, further research is needed to validate these findings in different geographic locations and with different types of converter models.

Reiterating the importance of wave energy conversion, it is a promising renewable energy source that can help mitigate climate change and reduce dependence on fossil fuels. By improving the efficiency and profitability of wave energy conversion through the use of advanced modeling techniques, we can accelerate the transition to a more sustainable energy future.

In summary, our study provides valuable insights into the potential for regression models to improve wave energy conversion and contributes to the growing body of research in this field.

Areas for Future Research:

There are several areas for future research in the field of wave energy conversion. One potential direction could be to explore the use of other machine learning models, such as support vector machines or neural networks, to see if they can improve upon the performance of the regression models used in this study. Additionally, further research could be conducted to investigate the impact of different wave scenarios and weather conditions on the efficiency of wave energy converters.

Another area for future research could be to conduct a cost-benefit analysis of wave energy conversion compared to other forms of renewable energy, such as wind or solar. This could help determine the economic feasibility of wave energy conversion and identify potential barriers to its widespread adoption.

Finally, more research could be done to explore the environmental impact of wave energy conversion and how it compares to other forms of renewable energy. This could include studying the effects on marine ecosystems and biodiversity, as well as the potential for conflicts with other ocean uses such as fishing or shipping.

Overall, the findings of this study suggest that regression models can be useful in predicting the power output of wave energy converters, but further research is needed to fully understand the potential of wave energy conversion and its place in the larger landscape of renewable energy.

References:

Github: https://github.com/coder2200/Wave-Energy-Converters

Dataset Used: https://archive-beta.ics.uci.edu/dataset/494/wave+energy+converters

[1] M. Greaves et al., "Optimising Wave Energy Converter Arrays Using a Genetic Algorithm," Journal of Offshore Mechanics and Arctic Engineering, vol. 129, no. 1, pp. 1-8, 2007.

[2] S. Sallam et al., "Hybrid Optimization Algorithm for Wave Energy Converter Array Design," IEEE Transactions on Sustainable Energy, vol. 5, no. 2, pp. 453-462, 2014.

[3] Y. Wang et al., "Deep Neural Network-Based Power Prediction of Wave Energy Converters," IEEE Transactions on Sustainable Energy, vol. 11, no. 4, pp. 2113-2123, 2020.

[4]Venugopal, V., et al. (2019). Shape Optimization of Wave Energy Converter's Heaving Buoy. Energies, 12(4), 719.

[5]Shahabi, M., et al. (2020). Model Predictive Control for Maximizing the Energy Harvesting of Wave Energy Converters. Energies, 13(17), 4395.

[6]Tawfik, A., et al. (2018). Performance of Wave Energy Converters in Regular and Irregular Waves. Applied Sciences, 8(7), 1065.

[7]Dufreche, D., et al. (2021). A Floating Ring Wave Energy Converter: Design, Modeling, and Optimization. Energies, 14(5), 1459.

Fig1. http://www.coastalwiki.org/wiki/Wave_energy_converters