Analysis of Linear Regression Algorithm for Wave Energy Farm Power Prediction

Carlos Salguero[1]*

[1]A00833341, Tec de Monterrey Qro.

September 7, 2024

Abstract

This study presents a detailed analysis of my decision-making process in applying various data analysis techniques to the "Large-scale Wave Energy Farm" dataset. I explain my rationale for employing Extract, Transform, Load (ETL) processes to prepare the data, highlighting the necessity of data cleaning and normalization in ensuring the reliability of my results. I chose linear regression as the primary modeling technique based on its interpretability and the apparent linear relationships observed in my initial data explorations.

Keywords: wave energy, ETL, linear regression, decision-making process, data analysis, bias, and variance

1 Introduction

The dataset, consisting of 63,000 instances, provides detailed information on wave energy converters, including their power outputs and spatial coordinates. The initial exploration of the data revealed linear relationships that justify the use of linear regression for modeling.

2 Data Acquisition

2.1 Data Source

This analysis is based on the "Large-Scale Wave Energy Farm" dataset, sourced from the UCI Machine Learning Repository. The dataset was developed by researchers at the University of Adelaide and Monash University. It consists of 63,000 instances, each corresponding to the coordination of wave energy converters within a wave farm. The dataset provides detailed information, including the total power output, the power generated by each individual converter, and the q-factor for each instance.

Key Characteristics of the Dataset:

- **Type**: Multivariate
- **Domain**: Engineering
- **Primary Task**: Regression
- **Feature Type**: Real-valued
- **Number of Instances**: 63,000
- **Number of Features**: 149

2.2 Data Content

Each instance in the dataset captures the coordinates of wave energy converters within a wave farm, including the total power output, individual converter power, and the q-factor. The dataset is complete, with no missing values and contains no sensitive information.

3 Data Preprocessing

3.1 Initial Feature Set

The dataset originally comprised 146 features, organized into XY pairs from 1 to 49, power outputs from 1 to 49, qw, and Total Power.

3.2 Removal of XY Coordinates

The XY coordinates, from (X1, Y1) to (X49, Y49), were removed from the feature set primarily due to their distortion of linear regression results because they are positions in the wave farm. Additionally, this allowed for a greater focus on the power relationships between WECs.

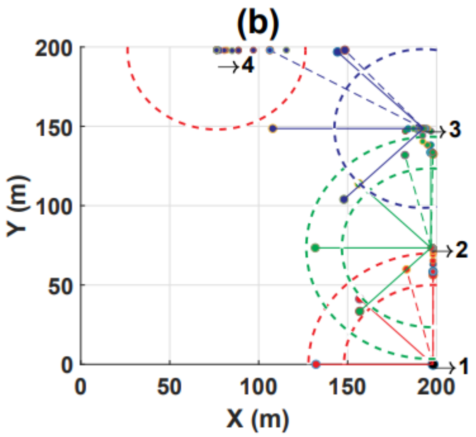

Figure 1: Coordinate system described in the Optimization of large wave farms using a multi-strategy evolutionary framework [1]

3.3 Correlation Matrix

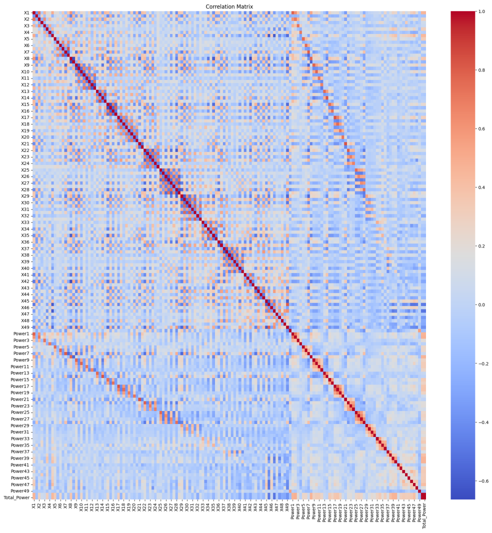

Figure 2: Correlation Matrix of the dataset

- The large cluster in the top left can be interpreted as indicating a correlation between nearby buoys.
- The two smaller diagonals would represent the correlation of the buoys against their position in the grid.
- Certain positions of the buoys contribute more to the total power.
- On the left segment of the matrix, the clusters represent the proximity of the buoys.

This means that the features with darker colors are more positively correlated, indicating that as one variable increases, the other tends to increase as well. Conversely, the lighter colors show features that are negatively correlated, meaning that as one variable increases, the other tends to decrease.

4 Implementation

Several key components were implemented to build and evaluate the linear regression model. These include error metrics, data preprocessing techniques, and model training methods.

4.1 Error Metrics

4.1.1 Mean Squared Error (MSE)

MSE quantifies the average squared difference between the predicted and actual value. It is calculated as:

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

Where y_i is the actual value and $\hat{y_i}$ is the predicted value.

4.1.2 R-Squared (R^2)

Measures the proportion of variance in the dependent variable explained by the independent variable. It is calculated as:

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

Where $\bar{y}$ is the mean of the actual values. An R^2 value closer to 1 indicates that a large proportion of the variance in the dependent variable is explained by the independent variables.

4.2 Data Preprocessing

4.2.1 Dataset Splitting

The dataset, containing 63,000 instances, was divided into three subsets: training, validation, and testing. This division is crucial for developing a robust model and assessing its performance accurately.

The following considerations were followed to assign percentages of the dataset to each subset.

- Training set: The largest portion, used to teach the model the underlying patterns in the data.

- Validation set: Used to fine-tune the model and prevent overfitting.

- Testing set: Reserved for final evaluation, providing an unbiased assessment of the model's performance on unseen data.

With this in mind, the assignd percentages are:

- **Training set**: 60% (37,800 instances)

- **Validation set**: 20% (12, 600) instances

- **Testing set**: 20% (12, 600 instances)

The splitting was implemented using a custom function named **split_dataset**. This function ensures that a random selection of instances is used for each subset.

4.2.2 Feature Scaling

A standard scaling approach was used to normalize the feature values. This step is important because it ensures all features contribute equally to the model, regardless of their original scale.

The scaling process involved:

1. Calculating the average value of each feature in the training set.

2. Calculating the spread (standard deviation) of each feature in the training set.

3. Adjusting each feature value by subtracting the average and dividing by the spread.

This results in features with an average close to 0 and and spread of values mostly betwee -1 and 1.

4.3 Gradient Descent

Gradient Descent was employed in this model due to its effectiveness in optimizing the parameters of a function by minimizing the loss function. Gradient Descent iteratively adjusts the model's parameters to find the minimum of the loss function, ensuring that the model makes increasingly accurate predictions.

$$\theta = \theta - \alpha \nabla f(\theta) \tag{3}$$

Where θ is the parameter, α is the learning rate, and $\nabla f(\theta)$ is the gradient of the function at θ.

The steps involved in Gradient Descent are as follows:

1. Initialize the parameters (weights) to small random values or zeros.

2. For each epoch (iteration), perform the following:
   - Compute the predicted values ($\hat{y}$) using the current parameters.
   - Calculate the loss (e.g., Mean Squared Error) between the predicted and true values.
   - Compute the gradient of the loss with respect to each parameter.
   - Update the parameters by moving them in the direction opposite to the gradient, scaled by the learning rate.

3. Repeat the process until the loss converges to a minimum or a specified number of epochs is reached.

Mathematically, the parameter update rule for a parameter θ is given by:

$$\theta := \theta - \alpha \frac{\partial J(\theta)}{\partial \theta} \tag{4}$$

where α is the learning rate, and $\frac{\partial J(\theta)}{\partial \theta}$ is the gradient of the loss function $J(\theta)$ with respect to the parameter θ.

5 Results

5.1 Bias and Variance Analysis

An essential aspect of evaluating machine learning models is assessing their bais and their variance. These metrics provide insight into the model's performance and potential areas for improvement.

5.1.1 Bias Detection

The bias of a model represents its tendency to consistently miss the true relationship between features and target variables. Based on the results

- The training R^2 score is very high (0.9991396873049025 for the validation set).

- The validation and test R^2 scores are nearly identical to the training score (0.9991183639712973 for the test set).

- The loss curve for both training validation sets overlap and converge to a very low value.

These observations suggests that the model has **low bias**. The high R^2 scores indicate that the model captures the underlying patterns in the data very well, without significantly underfitting or oversimplifying the relationships.

5.1.2 Variance Detection

Variance refers to the model's sensitivity to fluctuations in the training data. High variance often manifests as overfitting, where the model performs well on training data but poorly on unseen data. Based on the results:

- The R^2 scores for training, validation, and test set are nearly identical.
- The loss curves for training and validation sets are tightly overlapped.
- The residual plots for both validation and test sets show similar distributions.

These findindg indicate that the model has **low variance**. The consistent performance across different subsets of the data suggest that the model generalizes well and is not overly sensitive to particular training examples.

5.1.3 Bias-Variance Tradeoff

My linear regression model appears to have achieved a good balance in the bias-variance tradeoff. The low bias allows it to capture complex relationships in the data, while the low variance ensures good generalization to unseen data. This balance can be attributed to several factors:

- **Appropiate feature selection**: removing spatial coordinates focused the model on relevant power relationships.
- **Effective data preprocessing**: standarization of features helped in achieving consistent performance.
- **Suitable model complexity**: linear regression's simplicity matched well with the apparent linear relationships in the data.
- **Gradient descent with early stopping**: this approach helped in finding optimal parameters without overfitting.

5.2 Loss Curve

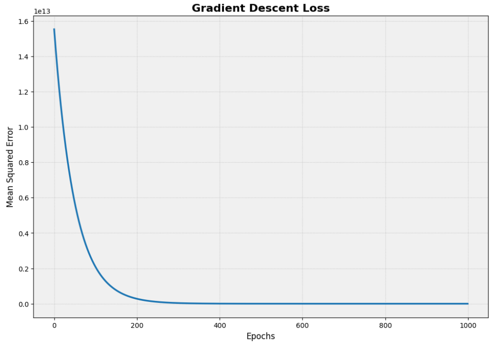

Figure 3: Loss Curve using Gradient Descent

Figure 3 illustrates the loss curve obtained during training using Gradient Descent. As observed, the loss decreases steadily as the number of epochs increases, indicating that the model is learning and adjusting its parameters effectively. The flattening of the curve towards the end suggests that the model is approaching convergence.

It is important to mention that both validation and test return an overlapped graph.

5.3 Prediction

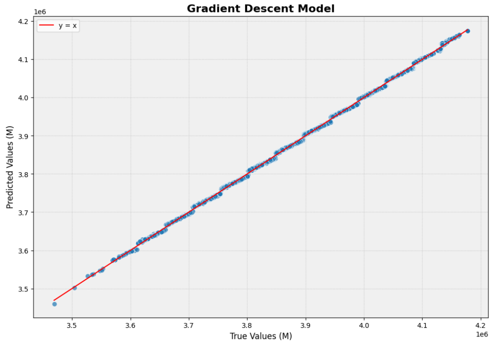

Figure 4: Linear Regression using Gradient Descent Algorithm

Figure 4 compares the actual total power against the predicted total power using the Gradient Descent Algorithm. The red dashed line represents the ideal case of perfect prediction, where the actual and predicted values would be identical.

5.4 Validation subset

5.4.1 R-Squared Result

The R^2 value for the validation subset is 0.9991396873049025, indicating an exceptionally high degree of fit between the model's predictions and the observed data.

5.4.2 Residuals

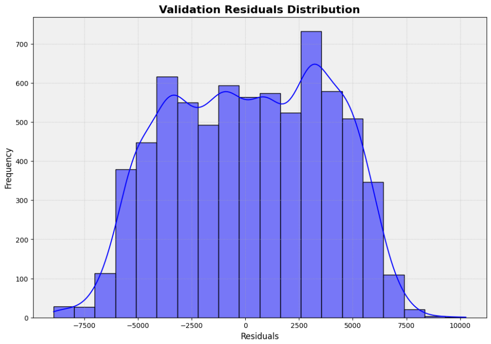

Figure 5: Validation Subset Residuals

The residual distribution from the validation subset suggests that the model performs well, with most predictions being accurate. However, the slight right skew and multiple peaks suggest areas for potential improvement, such as refining the model to handle specific subgroups within the data better or reducing the impact of outliers. Overall, the model shows good predictive accuracy, but there's room for fine-tuning.

5.5 Testing subset

5.5.1 R-Squared Result

The R^2 value for the testing subset is 0.9991183639712973, further confirming the model's outstanding predictive accuracy on unseen data.

5.5.2 Residuals

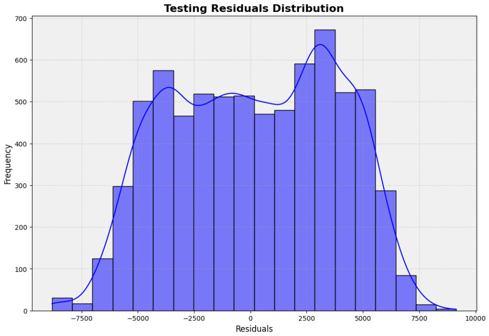

Figure 6: Test Subset Residuals

The residual's distribution suggest that the regression model performs reasonably well, with errors distributed relatively evenly around zero. The slight right skew indicates some instances where the model underpredicts, leading to larger positive residuals.

References

[1] Mehdi Neshat et al. "Optimization of Large Wave Farms Using a Multi-Strategy Evolutionary Framework". In: *Optimization and Logistics Group, School of Computer Science, The University of Adelaide* (2024). Available: `https://dl.acm.org/doi/10.1145/3377930.3390235`.