Linear Feature Engineering

Ahmed Najeeb, Gobinda Pandey

September 2024

Training Error: 31.44
Predicted Test Error: 36.15

1 Introduction

In this project, we applied Least Squares regression to a given dataset with the aim of exploring various feature engineering techniques and transformations to improve model performance. We began by analyzing the data, expanding features, and evaluating model performance using mean squared error as our metric. To ensure that our model doesn't over fit we employ cross-validation. We finally, reported the training and predicted test errors based on the selected model.

2 Data Analysis

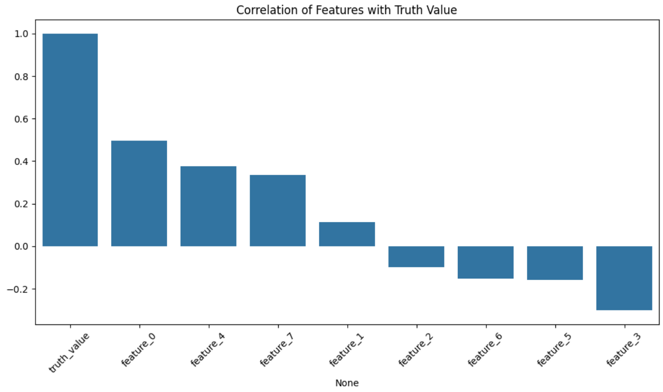

Figure 1: Feature correlation against Truth value.

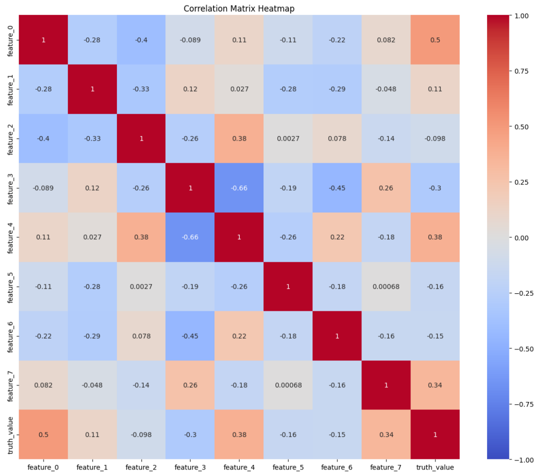

Figure 2: Correlation Heatmap

We analyzed the dataset, which consists of 8 features and 926 data points, by examining the Correlation Matrix Heatmap and the correlation of each feature with the target variable.

The correlation analysis revealed that ‘feature_0‘ had the highest positive correlation with the target (0.498), while ‘feature_3‘ had the highest negative correlation (-0.302). The Correlation Matrix Heatmap provided a comprehensive view of the relationships between all features.

Initial analysis also indicated that out of the 8 features, 4 were positively skewed and the remaining features were negatively skewed. The figures 1,2 illustrate the correlations of individual features with the target variable and provide insights into the overall feature relationships within the dataset.

3 Feature Engineering

To improve model performance, we applied feature engineering techniques by implementing various transformations on the features. The selection of transformations was guided by analyzing the correlation between the transformed features and the target variable as shown in Fig 3. We retained only those transformations that resulted in a stronger correlation, as these were deemed more likely to contribute positively to the model's predictive accuracy.

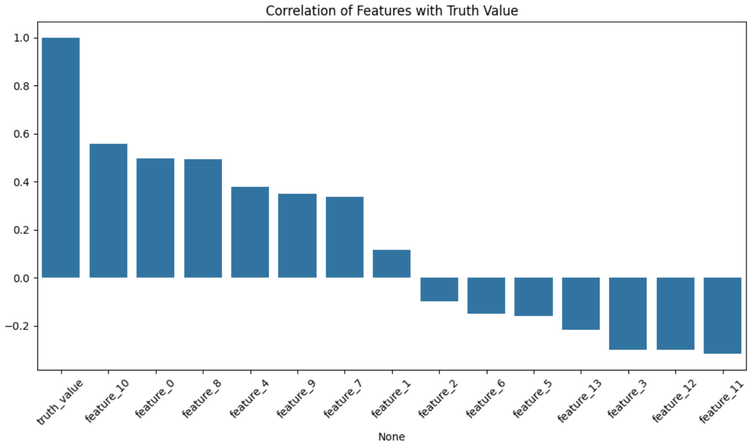

Figure 3: Correlations after log transformation

We applied the following transformations to the features:

- **Logarithmic Transformation**: Applied as $\log(x)$, where $x > 0$. For $x \leq 0$, the value was set to 0.
- **Polynomial Expansion**: Included quadratic (x^2) and cubic (x^3) terms.
- **Trigonometric Transformations**: Applied sine ($\sin(x)$) and cosine ($\cos(x)$).
- **Reciprocal Transformation**: Computed as $1/x$, where $x \neq 0$.
- **Square Root Transformation**: Applied as $\sqrt{x}$.

These transformed features were concatenated with the original features to form the final feature set used for model training and evaluation.

4 Evaluation

We employed linear regression using the least squares method to fit our data and **MSE** evaluate the model's performance. Additionally, we incorporated various feature transformations to enhance model accuracy, including polynomial expansions, trigonometric functions, and reciprocal and square root transformations.

To address potential overfitting, particularly due to the limited size of our dataset (926 data points), we utilized K-fold cross-validation with K=10 folds. This method helped in robustly assessing the model's generalization ability by training on different subsets of the data and predicting the performance on unseen test data. The combination of polynomial expansions of degrees 2 and 3, along with sine, cosine, and logarithmic transformations, led to the best performance, with a minimum mean cross-validation error of 36.15, indicating the optimal balance between model complexity and predictive accuracy.