The problem-solving process began with data cleaning, where I string "(null)" in columns with np.nan. Next, I removed rows where the target variable, *ActualTOW*, was missing, since such samples are unsuitable for training the model. Then, I set the appropriate data types for the relevant columns. Afterwards, I proceeded with data preprocessing. I visualized the distributions of the features and applied a *PowerTransformer* to those exhibiting skewed distributions to normalize them. For categorical features, I used *LabelEncoder* instead of *OneHotEncoder* due to the large number of unique categories, which made label encoding a more efficient choice. I then created a preprocessing function that performs these operations and returns the prepared input features (*X*) in the proper format. I applied a Standard Scaler to standardize the entire dataset. Subsequently, I trained three different models: Linear Regression, Random Forest Regression, and Support Vector Regression (SVR). The Random Forest Regressor performed the best, so I focused on tuning its hyperparameters to improve performance further. I visualized the training results to compare the models, and finally, using the trained Random Forest model, I predicted *ActualTOW* on the validation set. To solve this exercise I used Google Colab.