ML Practice:

Certain Machine Learning models will be applied to predict the price of a used car.

Importing the libraries:

- Pandas
- Numpy
- Scipy
- Plotly
- Seaborn
- Matplotlib

Importing the dataset:

- The source of the dataset used for this practice is: 1985 Auto Imports Database

Displaying some observations from the dataset:

5 rows × 26 columns

Viewing the dataset after including the headers to the columns:

5 rows × 26 columns

Repalcing "?" with np.NaN to address the missing values later:

Before Replacing the "?" with the np.NaN:

5 rows × 26 columns

After Replacing the "?" with the np.NaN:

5 rows × 26 columns

Getting some insight about the dataset:

Figuring out the data type of each column:

From the outcome we could see that there are some columns are not with the right data type:

- normalized-losses, bore, stroke, horsepower, peak-rpm, and price should be numeric values.

Applying one hot encoding on the "make" column:

- Then select the one with the highest correlation with the price:

Selecting only the one hot encoding (make) and the price columns:

Figuring out the column with a high correlation with the price:

The output shows that "mercedes-benz" has the highest correlation with the price:

- "mercedes-benz" will be selected for the further analysis and predictions.
- The other make columns related to the "make" column will be removed from the dataset.

Viewing the dataset after removing the columns:

5 rows × 26 columns

Converting the one hot encoded columns into integer data type:

More insights about the dataset:

[illegible]

Shape and size of the dataset:

[illegible]

Getting some statistical information about the dataset:

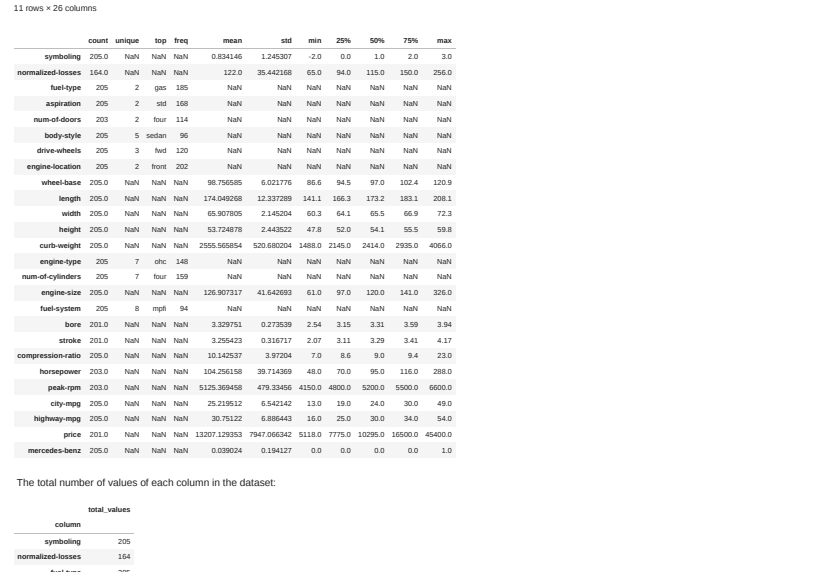
Only the Numeric Columns or values:

For string or categorical values:

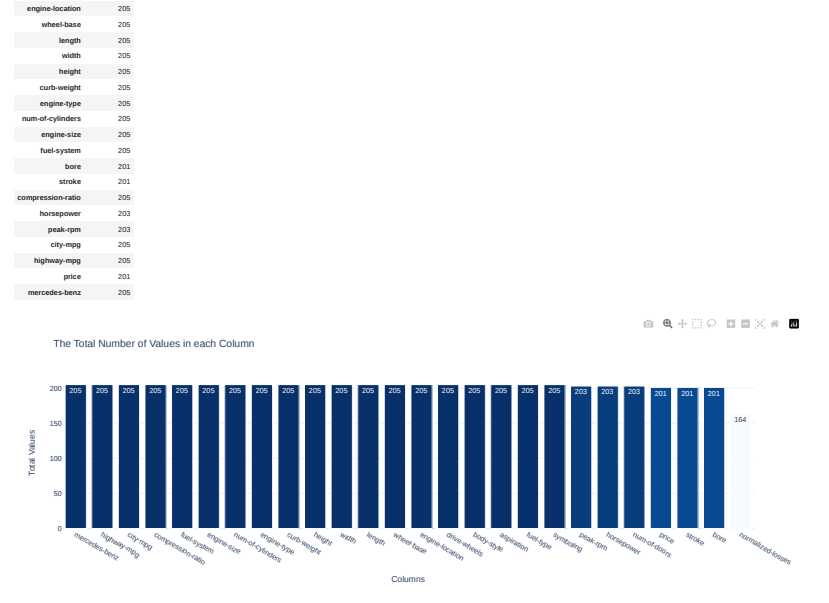
For all the columns at once:

11 rows × 26 columns

The total number of values of each column in the dataset:

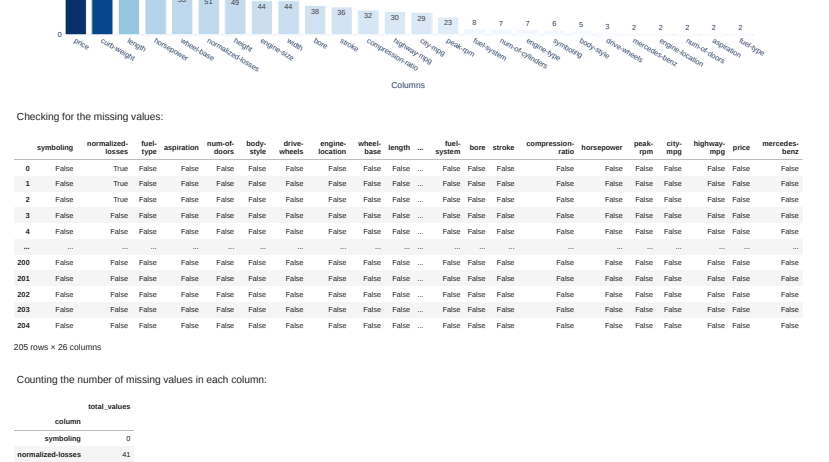


Unique values in each column:



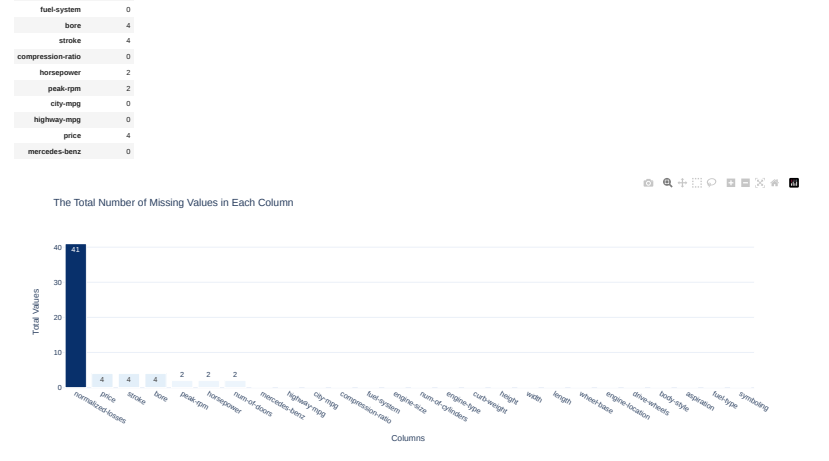
Checking for the missing values:

Counting the number of missing values in each column:



- Filling the missing values in the "num-of-doors" with the **most frequently appeared** word i.e "four".

Checking out one more time for the missing value, especially for the "num-of-doors" column:



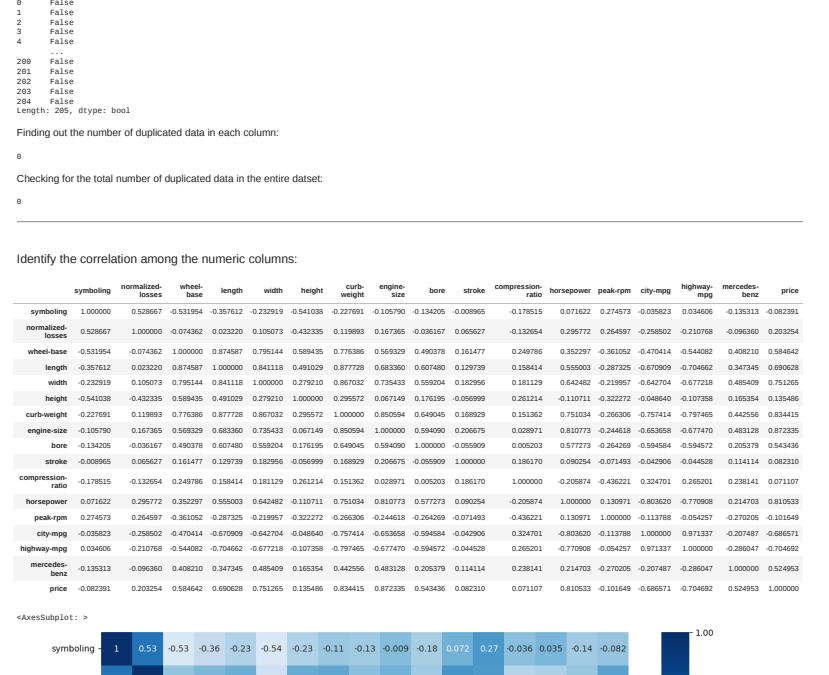
Getting the total number of missing values in the entire dataset:

Checking for the duplicated data:

Finding out the total number of duplicated data in each column:

Checking for the total number of duplicated data in the entire dataset:

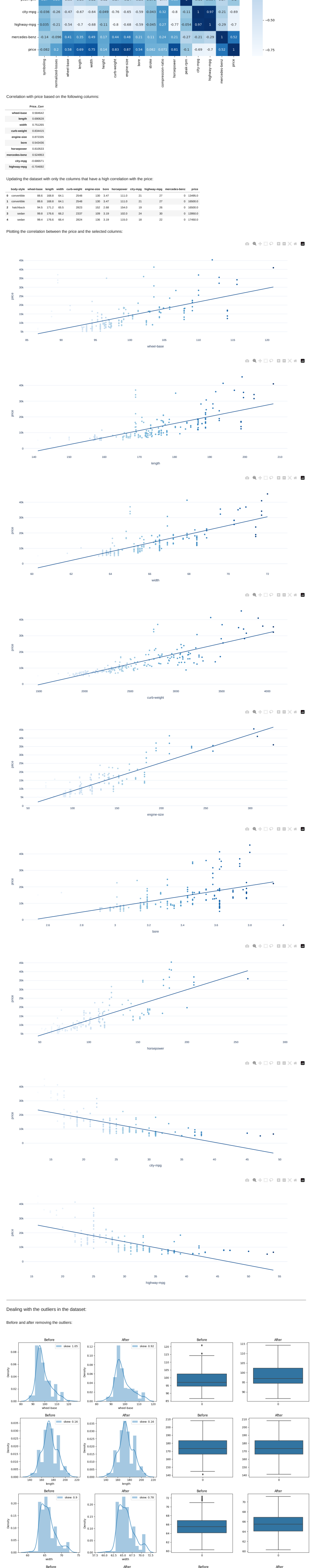
Identify the correlation among the numeric columns:



Correlation with price based on the following columns:

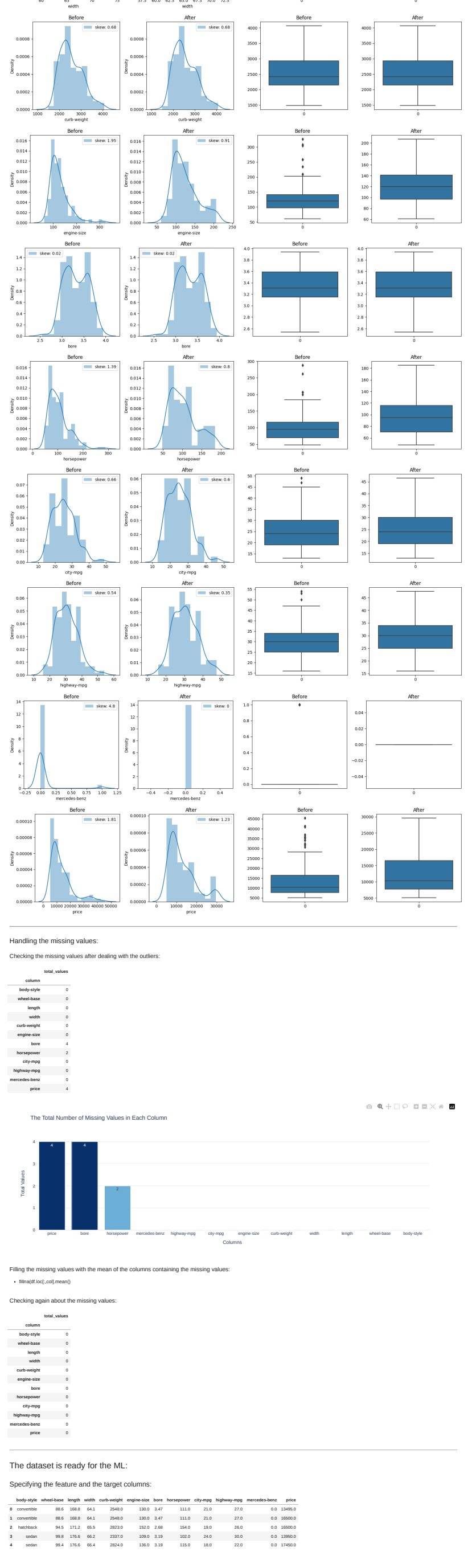
Updating the dataset with only the columns that have a high correlation with the price:

Plotting the correlation between the price and the selected columns:



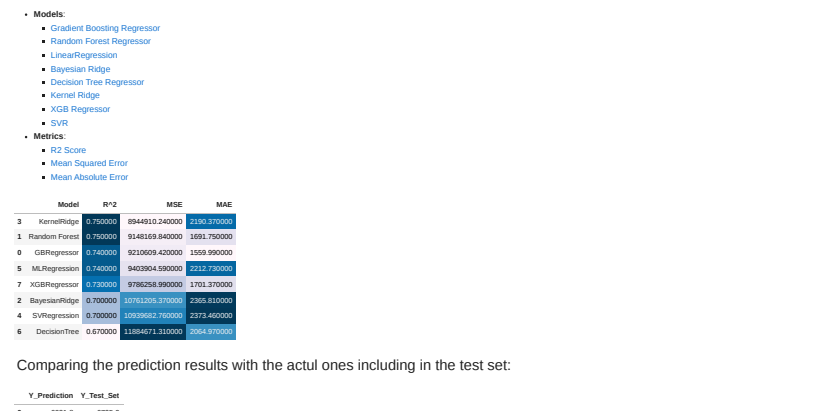
Dealing with the outliers in the dataset:

Before and after removing the outliers:



Handling the missing values:

Checking the missing values after dealing with the outliers:



Filling the missing values with the mean of the columns containing the missing values:

Checking again about the missing values:

The dataset is ready for the ML:

Specifying the feature and the target columns:

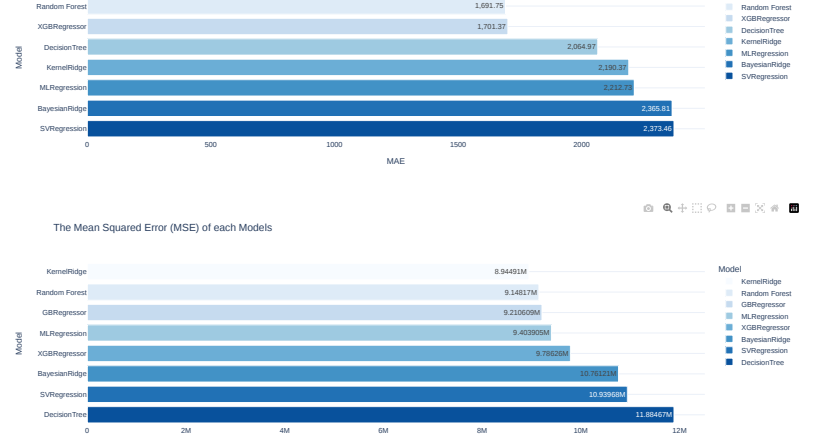
Splitting the dataset into:

- Training set (X_train, y_train)
- Test set (X_test, y_test)

Training multiple models on the "training set" and then fit the model

- Models:
  - [illegible]
- Metrics:
  - R2 Score
  - Mean Squared Error
  - Mean Absolute Error

Comparing the prediction results with the actual ones including in the test set:



From the observation:

- **Random Forest**, **Ridge**, and **Gradient Boosting Regressor** models realized the **highest** coefficient of determination (R^2), the **lowest** mean absolute error (MAE) and the **lowest** mean squared error (MSE).
- **AdaBoost**, **Gradient Boosting Regressor** and **Random Forest** achieved [illegible]