Automobile Price Prediction

Dataset consist of various characteristic of a car

Data overview

```
RangeIndex: 205 entries, 0 to 204
Data columns (total 26 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   symboling          205 non-null    int64
 1   normalized-losses  164 non-null    float64
 2   make               205 non-null    object
 3   fuel-type          205 non-null    object
 4   aspiration         205 non-null    object
 5   num-of-doors       203 non-null    object
 6   body-style         205 non-null    object
 7   drive-wheels       205 non-null    object
 8   engine-location    205 non-null    object
 9   wheel-base         205 non-null    float64
 10  length             205 non-null    float64
 11  width              205 non-null    float64
 12  height             205 non-null    float64
 13  curb-weight        205 non-null    int64
 14  engine-type        205 non-null    object
 15  num-of-cylinders   205 non-null    object
 16  engine-size        205 non-null    int64
 17  fuel-system        205 non-null    object
 18  bore               201 non-null    float64
 19  stroke             201 non-null    float64
 20  compression-ratio  205 non-null    float64
 21  horsepower         203 non-null    float64
 22  peak-rpm           203 non-null    float64
 23  city-mpg           205 non-null    int64
 24  highway-mpg        205 non-null    int64
 25  price              201 non-null    float64
dtypes: float64(11), int64(5), object(10)
```

This dataset consist of data From 1985 Ward's Automotive Yearbook from bellow sources:

- 1985 Model Import Car and Truck Specifications, 1985 Ward's Automotive Yearbook.
- Personal Auto Manuals, Insurance Services Office, 160 Water Street, New York, NY 10038
- Insurance Collision Report, Insurance Institute for Highway Safety, Watergate 600, Washington, DC 20037

There are 25 columns and 205 rows, with some missing values.

Target : Price (continous)
Categorical features : 10
Numerical features : 14

Problem and Objective

Our target variable (price) is **continuous**.
Therefore, the machine learning problem we face is **regression**.

Our objective in this analysis focuses on **prediction** to obtain the **highest accuracy** from the model

Missing value

	null (sum)	null (%)
normalized-losses	41	20.00
price	4	1.95
stroke	4	1.95
bore	4	1.95
peak-rpm	2	0.98
num-of-doors	2	0.98
horsepower	2	0.98
engine-type	0	0.00
highway-mpg	0	0.00
city-mpg	0	0.00
compression-ratio	0	0.00
fuel-system	0	0.00
engine-size	0	0.00
num-of-cylinders	0	0.00
symboling	0	0.00

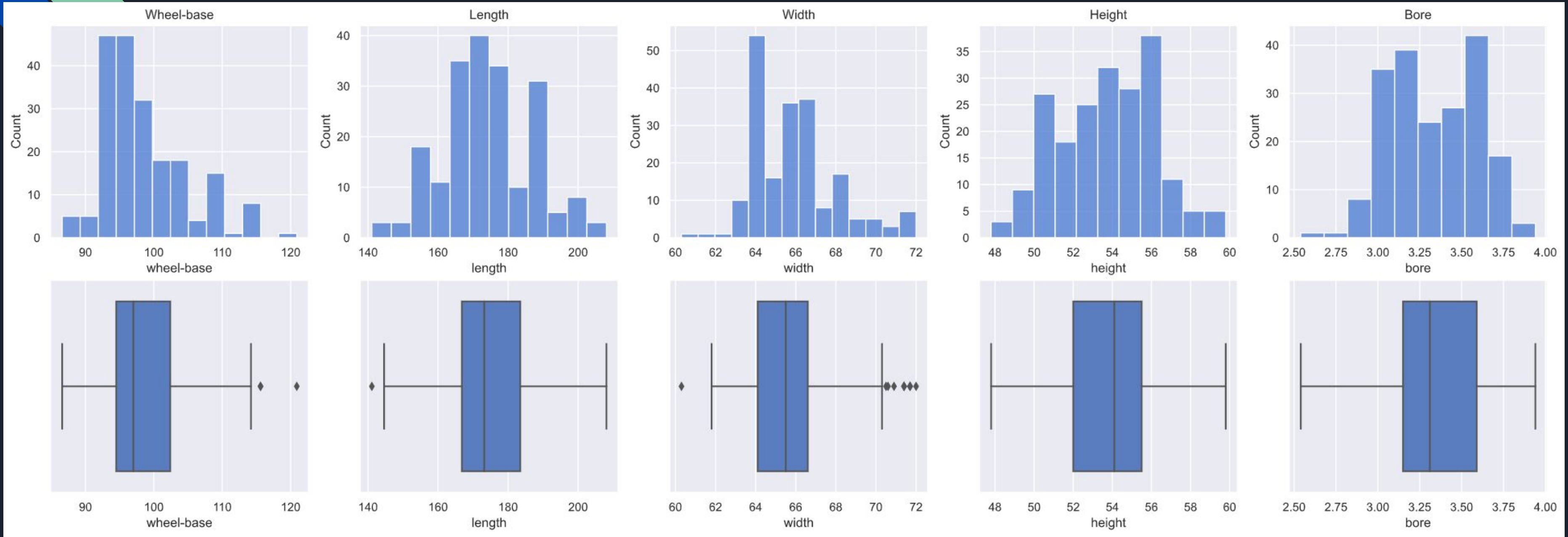
In our data set, there are several missing values with the highest number on normalized-losses feature.

Treatment for missing value:

- Remove rows with missing value in the price variable.
- Discard the normalized-losses column because the amount of missing value is too high.
- Performs imputation on stroke, bore, peak-rpm, num-of-doors, and horsepower features with their median, mean, or mode values (depending on the type and distribution of data).

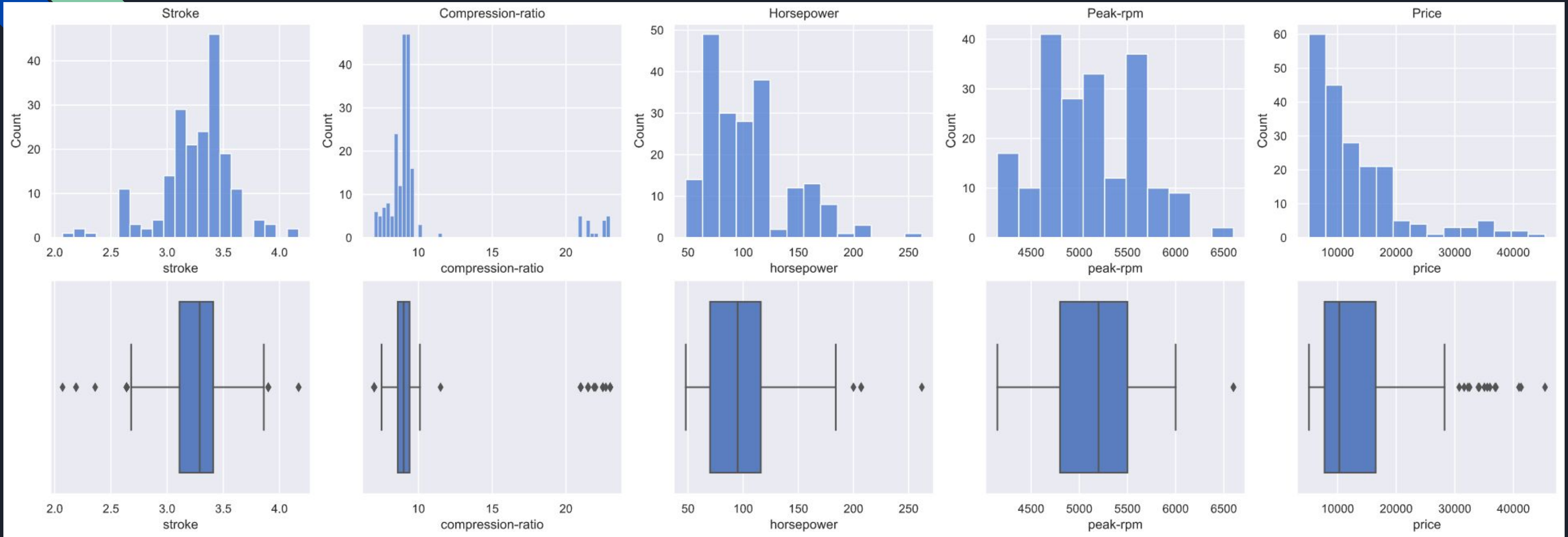
Exploratory Data Analysis

Univariate Analysis (continuous variable)



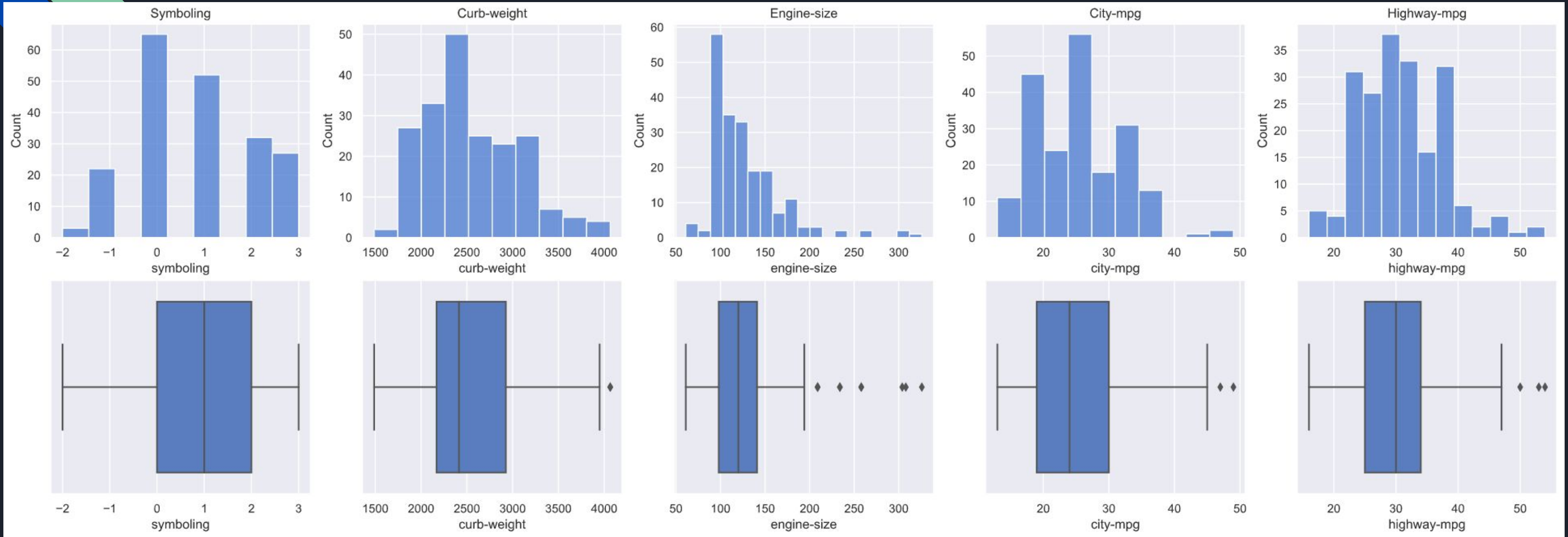
This is our first 5 continuous variables.
From our analysis, there are 2 variables which have high skew (greater than 0.85). Those variables are wheel-base and width.
We will do log transformation to these variables.

Univariate Analysis (continuous variable)



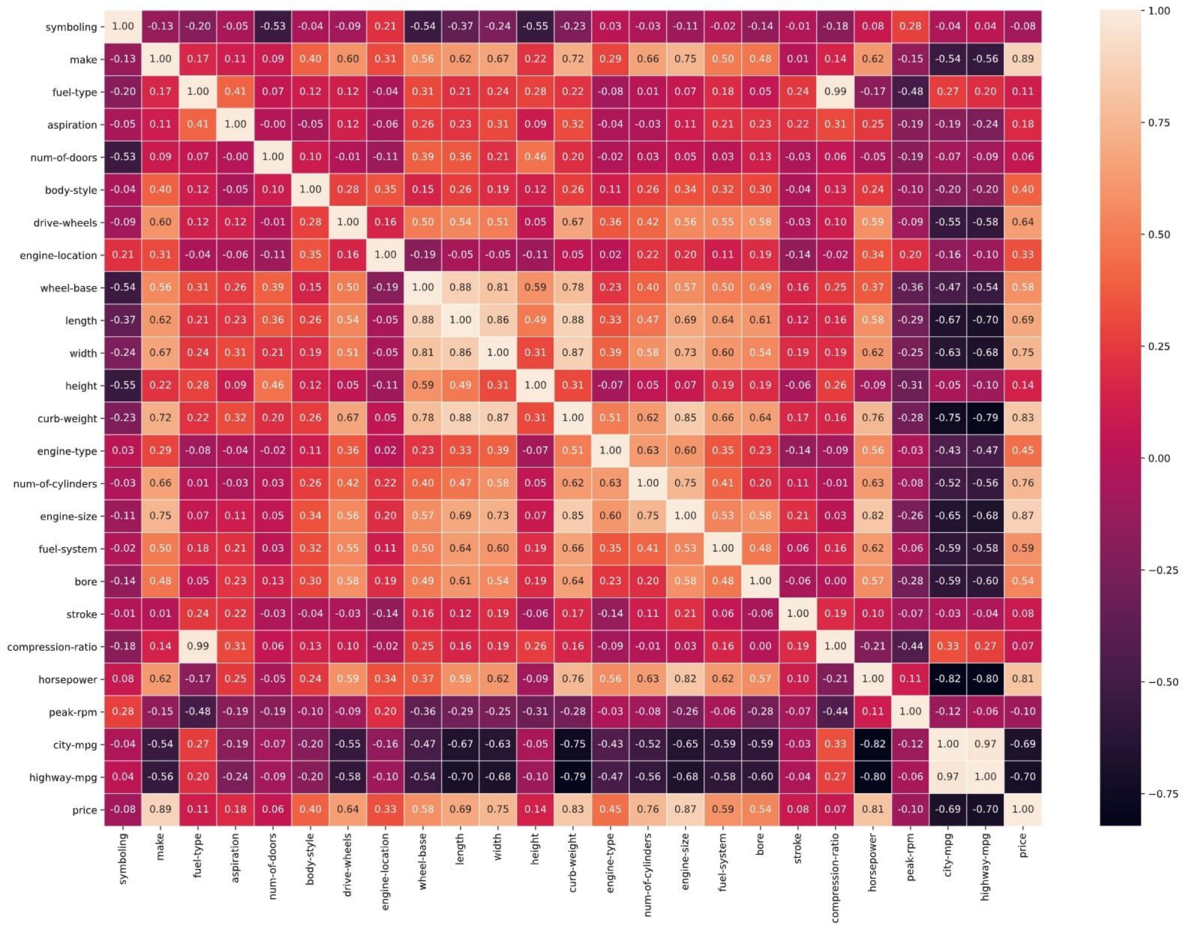
This is our next 5 continuous variables.
From our analysis, there are 3 variables which have high skew (greater than 0.85). Those variables are compression-ratio and horsepower. We will do log transformation to these variables.

Univariate Analysis (discrete variable)



This is our 5 discrete variables.
From our analysis, there are 1 variables which have high skew (greater than 0.85) which is engine-size.
We will do log transformation to these variables.

Correlation Heatmap



Our target variable has a strong correlation with several variables such as:

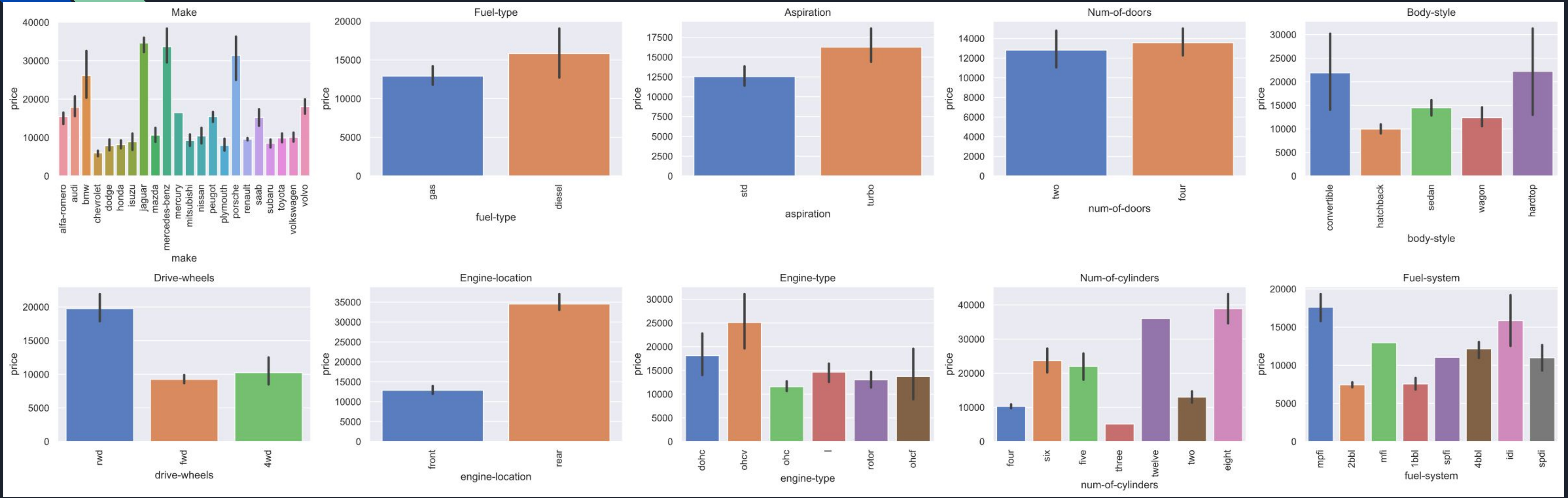
- make
- engine-size
- curb-weight
- horsepower
- num-of-cylinders
- etc

Problem:

There are 2 independent variables whose correlation is close to perfect, namely fuel-type with compression-ratio and city-mpg with highway-mpg.

We need to remove one of these variables to reduce the multicollinearity problem in the Linear Model.

Multivariate Analysis (categorical variables)



- Diesel and turbo car have higher average car price compared to the other category.
- Car with four doors have slightly higher price compared to car with two doors.
- Car with rwd type have more higher average car price compared to the other type.
- For make variable, there are to much category. We will group the low count car to “Other” category.

Data Preprocessing

Preprocessing

For preprocessing data, we do the following treatment:

1. Missing value imputation
   Perform imputation using median, mean, or mode values (depending on the type and distribution of data).
2. Outlier / skewness handling
   Using log transformation.
3. Categorical encoding
   Using labels and one-hot encoding.
4. Scaling
   Do scaling with minmaxscaler
5. Feature engineering
   Added polynomial features.

Modeling

Base Model (LinearRegression)

Model	Train R2	Test R2	Train RMSE	Test RMSE
Linear	0.9707	0.9071	1162.1067	3166.4252

There is no additional polynomial features on the base model. Because the number of features is large, and if you add the polynomial features it will overfit.

Ridge and Lasso Regression

Model	Train R2	Test R2	Train RMSE	Test RMSE
Ridge	0.9900	0.9505	679.2758	2311.8932
Lasso	0.9938	0.9234	534.2514	2874.9289

After adding polynomial features and scaling, we get higher scores with Ridge and Lasso (default alpha 1.0 parameter).

After tuning alpha

```
lasso = Lasso(alpha=4.192572864421096, max_iter=9999)
ridge = Ridge(alpha=1.732333139228904, max_iter=9999)

models = {'Lasso': lasso,
          'Ridge': ridge}

evaluate_model(models, X_train_sc_pf, X_test_sc_pf, y_train, y_test)
```

Model	Train R2	Test R2	Train RMSE	Test RMSE
Ridge	0.9932	0.9558	559.5325	2184.2458
Lasso	0.9908	0.9489	652.1030	2347.6488

Hyperparameter tuning is done by optuna.

https://optuna.org/

After tuning there is an increase in the score on both models.

The best model is Ridge with alpha 1.7323 with R-Squared on the test set 0.9558, and error (RMSE) on the test set 2184.24.

Recommendation

Recommendation

Suggestions for next steps in analyzing this data:

- This modeling was carried out with a train test split. To get better generalization results, it would be better to do it with cross validation.
- Perform different engineering features.
- Make modeling with different machine learning such as tree-based model or gradient boosting.