Automobile Price Prediction

Niels Moeller
Miguel Rodriguez
Tony Nguyen

Data description:

Source: https://www.kaggle.com/toramky/automobile-dataset

This dataset was compiled by Jeffrey C Schlimmer of the University of California Irvine Machine Learning Repository. It represents an aggregate of data from the following sources:

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

Overview:

There are three types of entities in this dataset:

1. The physical specifications of an Auto.
2. Its insurance risk rating.
3. Its normalized losses in use compared to other cars.

Column description:

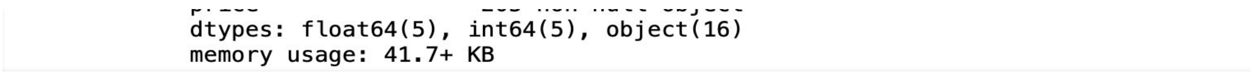
There are 5 rows and 26 columns in this set, with the features shown below:

1. symboling
2. Normalized-losses
3. make
4. fuel-type
5. aspiration
6. num-of-doors
7. body-style
8. drive-wheels
9. engine-location
10. wheel-base
11. length
12. width
13. height
14. curb-weight
15. engine-type
16. num-of-cylinders
17. engine-size
18. fuel-system
19. bore
20. stroke
21. Compression-ratio
22. Horsepower
23. Peak-rpm
24. city-mpg
25. highway-mpg
26. price

Data Types:

The data types are numeric data types (including float and integer) and text data types (including objects).

Using the command: car.info() to obtain the data structure of the dataset:



```
price                    205 non-null object
dtypes: float64(5), int64(5), object(16)
memory usage: 41.7+ KB
```

In this particular dataset, the data is categorized into numerical and categorical features.

Our Goal:

- Our goal is to ultimately predict the price of cars based on specifications detailed above.

Cleaning and Preprocessing data:

- Data cleaning: Data contains "?" replace it with NAN
- Dealing with missing values:

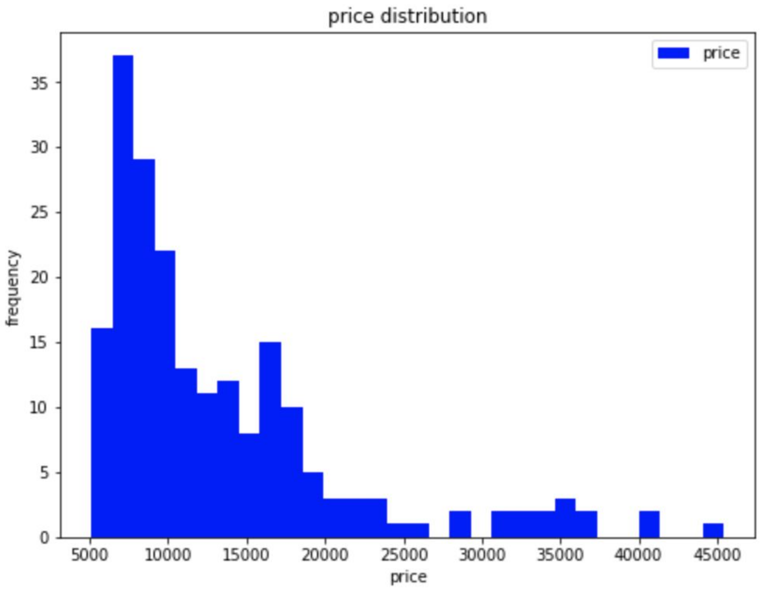
  A. Fill missing data of ‘normalised-losses’, ‘price’, ‘horsepower’, ‘peak-rpm’, ‘bore’, ‘stroke’ with the respective column mean

  B. Fill missing data category ‘Number of doors’ with the mode of the column (Four)
- Converting features to be float or integer type.

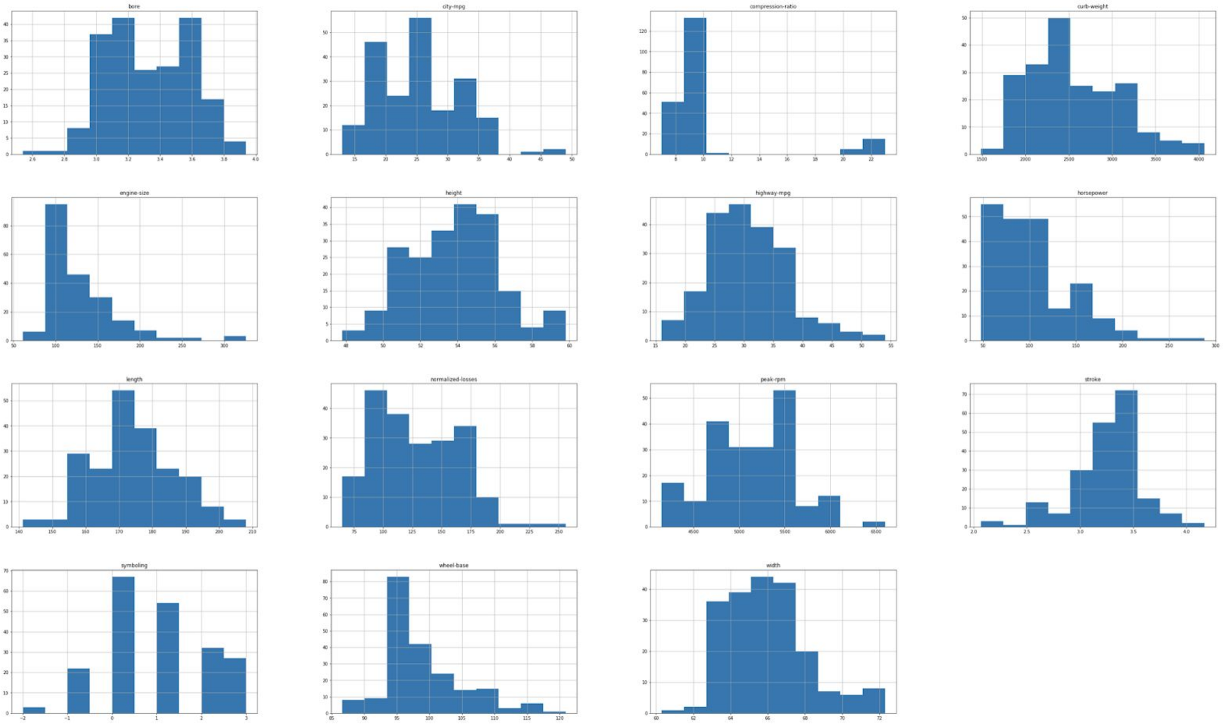
Exploratory Data Analysis:

After cleaning the data and treating the missing values, we will explore our dataset to understand and get some important insights.

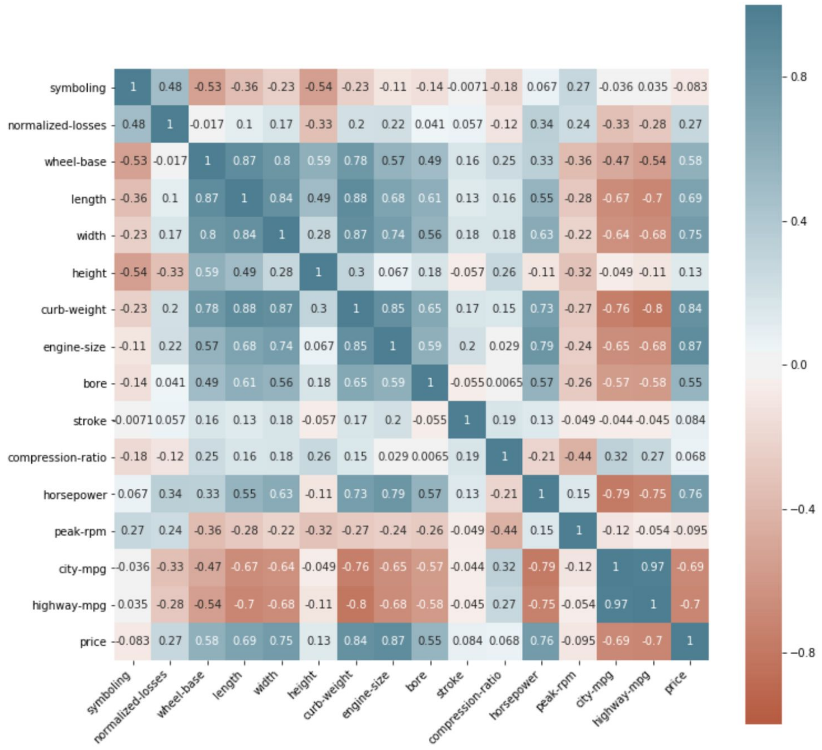
Firstly, we will take a look into our target: 'price' through a histogram of our feature price named ‘Price distribution’ with x-axis is price and y-axis is frequency:



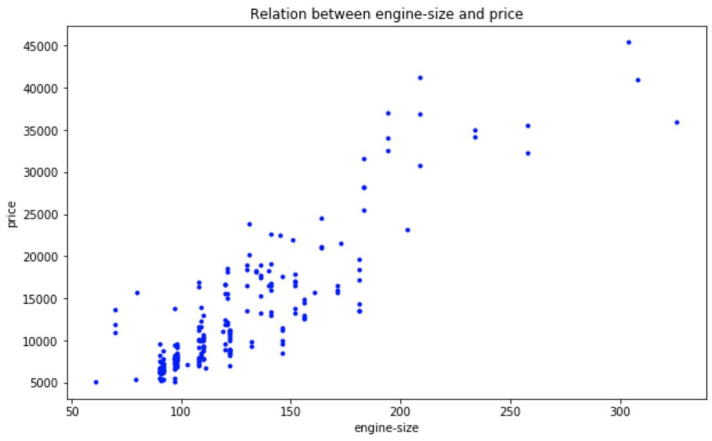
Clearly, our target does not have a normal distribution. Applying the same concepts, let's take a look into all numerical distributions through histogram for each feature as below:

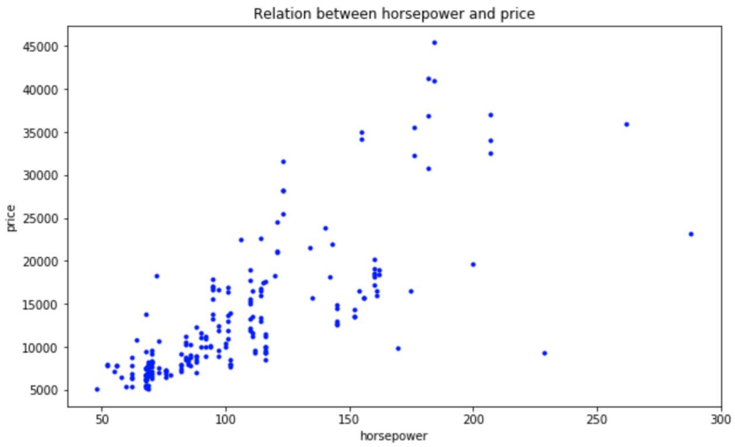


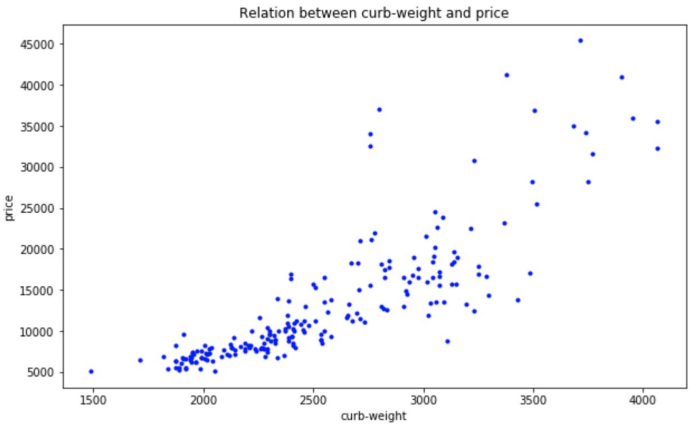
Similarly, none of our numerical features have a normal distribution, therefore let's deal with that in feature engineering in the next part. Now let us take a look into each numerical feature correlation with our target and with each other by the heatmap of the correlation matrix:

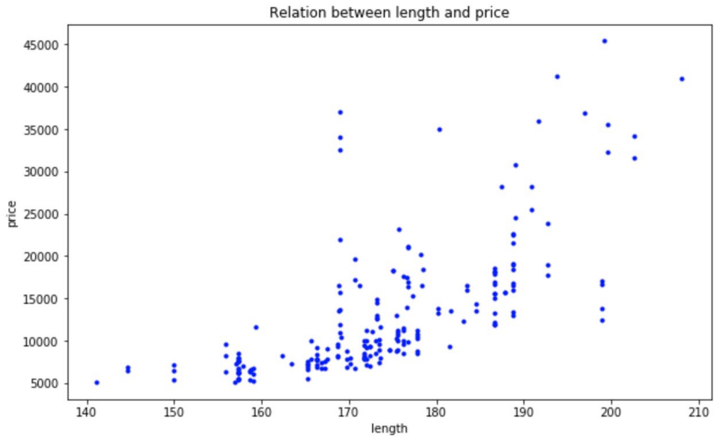


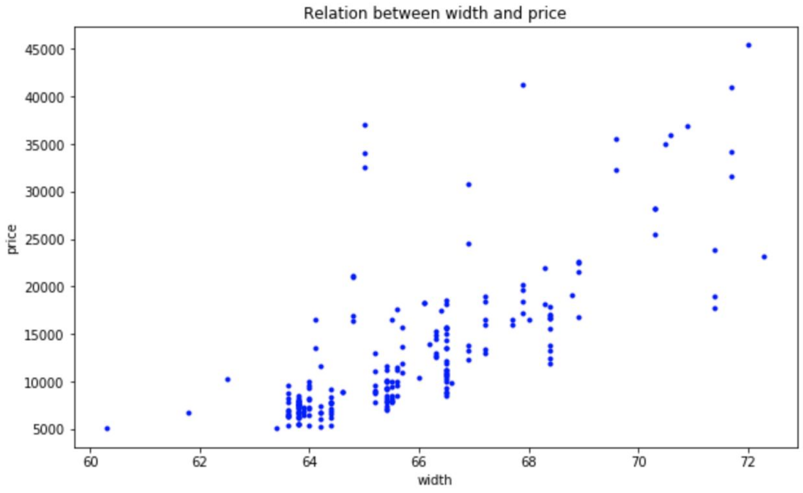
Through the heatmap, unlike city-mpg,highway-mpg are negatively correlated; engine_size, horsepower, curb_weight, length and width have positively high correlation with car prices. Furthermore, let's make scatter plot for each feature to better understand this relation:





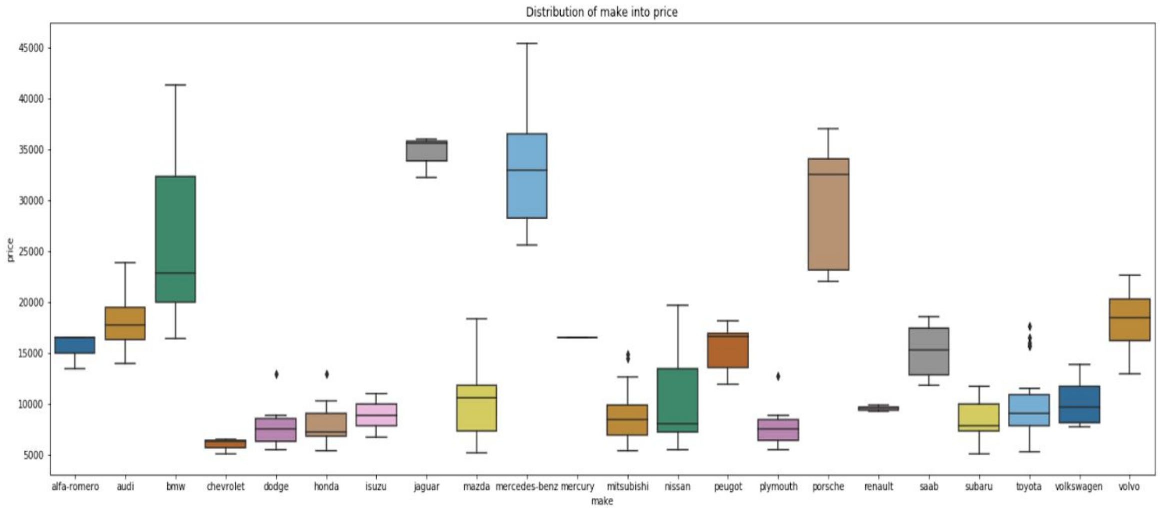






Engine size and price have a nice linear relationship with little to no outliers, this is a strong predictor. Secondly, the relation between horsepower and price has a good distribution but we also have some outliers, curb weight, length, and width are all fairly similar, a bit more sporadic than size, but still good indicators of price. We will need to remove some of these features in the feature engineering part to deal with abnormal distribution of target to avoid overfitting in our model.

In the next step of EDA, let's make a bivariate analysis of each categorical feature with our target to understand better these features through the box plot of the distribution of make into price with x-axis is the cars' makers and y-axis is the price of cars:



Findings we obtain from the box plot above and other plots from the .ipynb notebook:

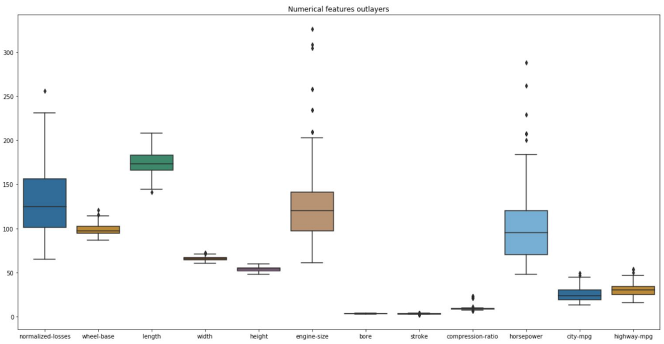
- Mercedes-Benz ,BMW, Jaguar, Porsche produce expensive cars which are more than 25000.
- Chevrolet, Dodge, Honda, Mitsubishi, Nissan, Plymouth, Subaru, Toyota produce budget models with lower prices.
- Most of the car companies produce cars in range below 25000.
- Hardtop models are expensive in prices followed by convertible and sedan body styles.
- Cars with higher prices have more cylinders, and ohcv or dohcv engines
- Turbo models have higher prices than for the standard model, but there exist outliers with std with high prices too.
- Convertible has only standard edition with expensive cars.
- Hatchback and sedan turbo models are available below 20000.
- Rwd wheel drive vehicles have expensive prices.

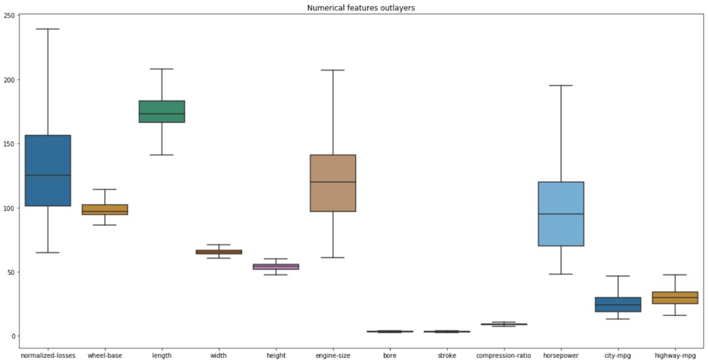
Feature Engineering:

For dealing with outliers in numeric features, we create a boxplot for each numerical feature to visualize and remove outliers by using Boxplot Interquartile Range(IQR) method.

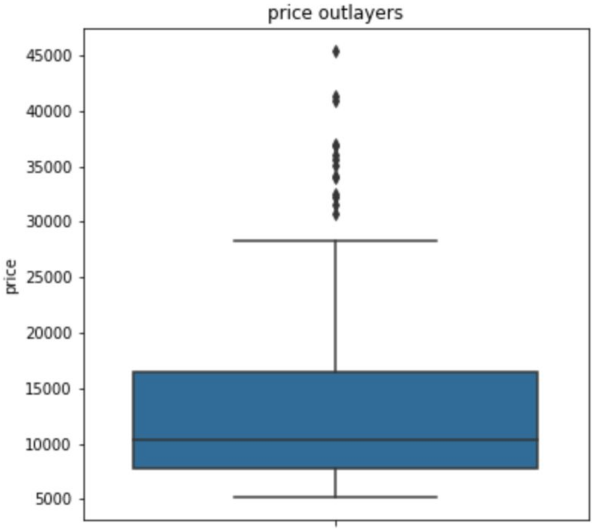
Specifically, IQR is the difference between Q3(25th percentile) and Q1(75th percentile), values above (Q3+1.5 IQR) and values below (Q1-1.5 IQR) are considered outliers, in a box plot we can find all this values and use then to remove outliers from our data.

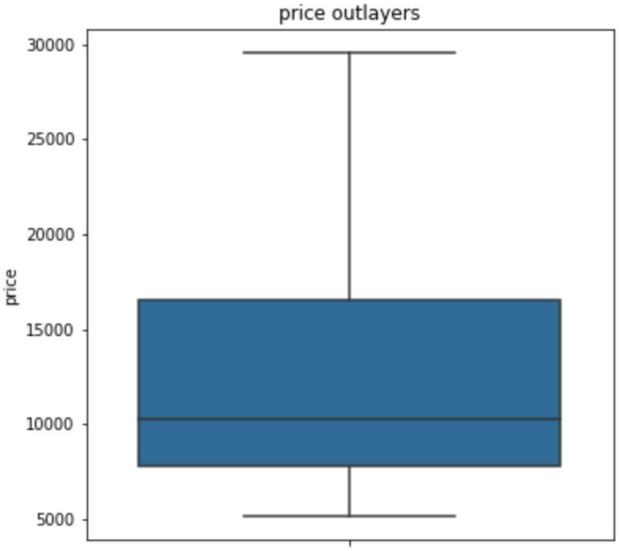
The result below is the box plot of numerical features before and after removing the outliers:



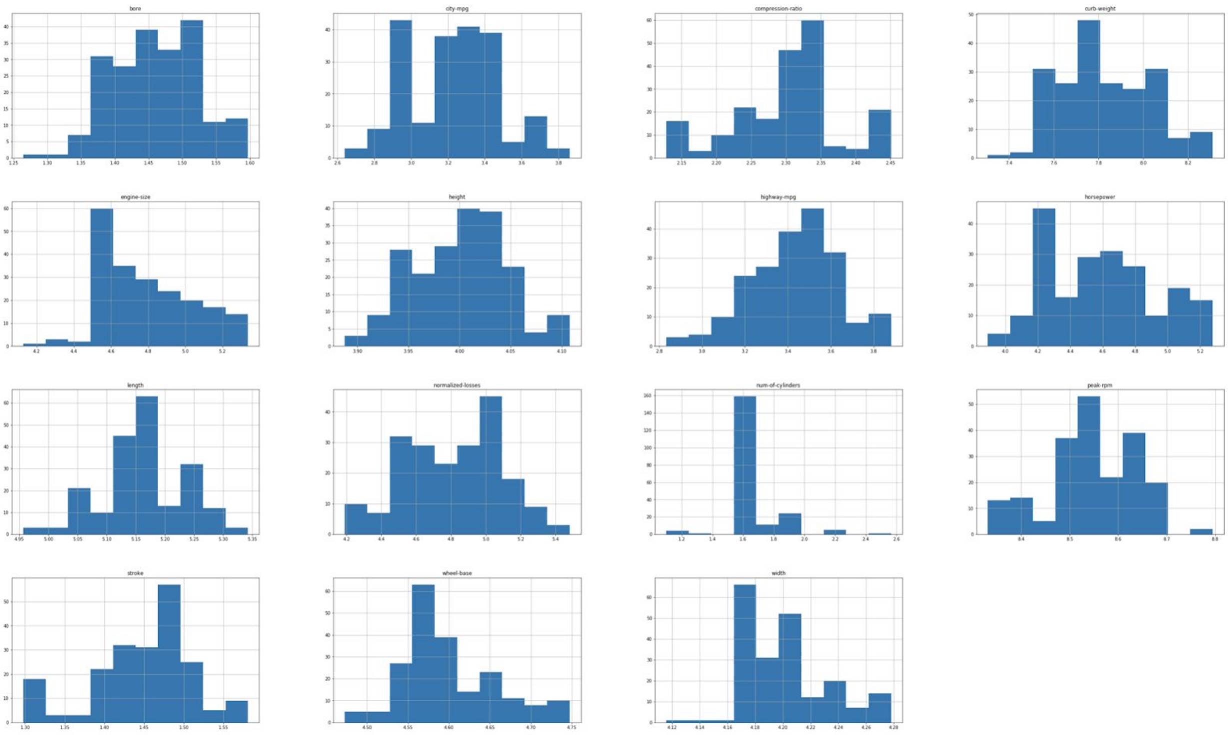


Applying IQR method to deal with outliers in our target ‘price’, we obtain the result after removing outliers above 30000:

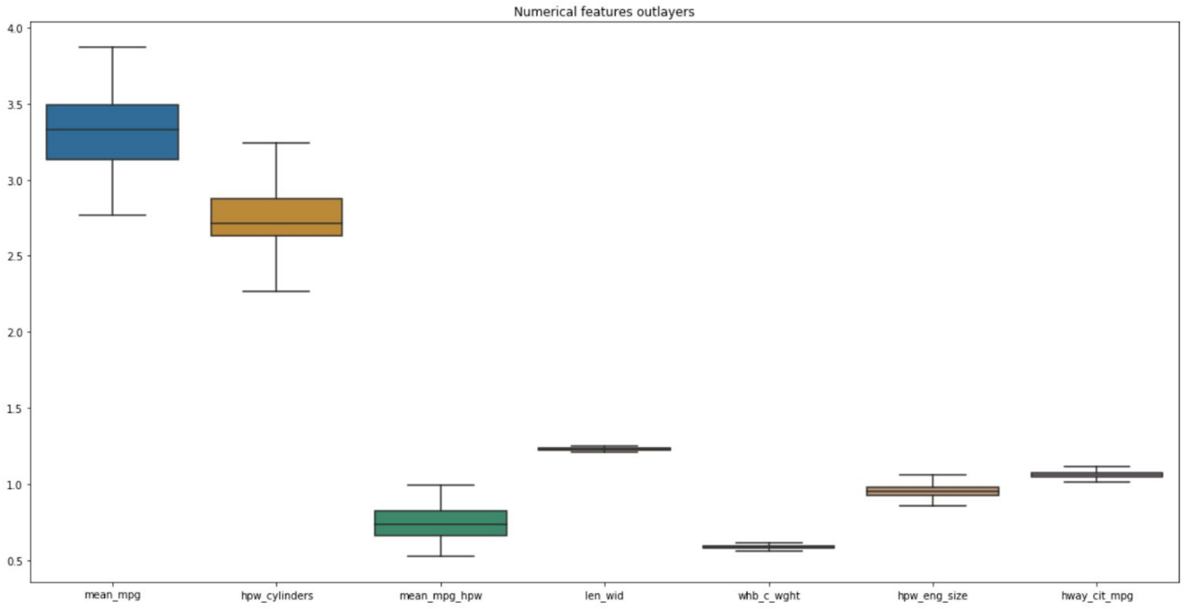




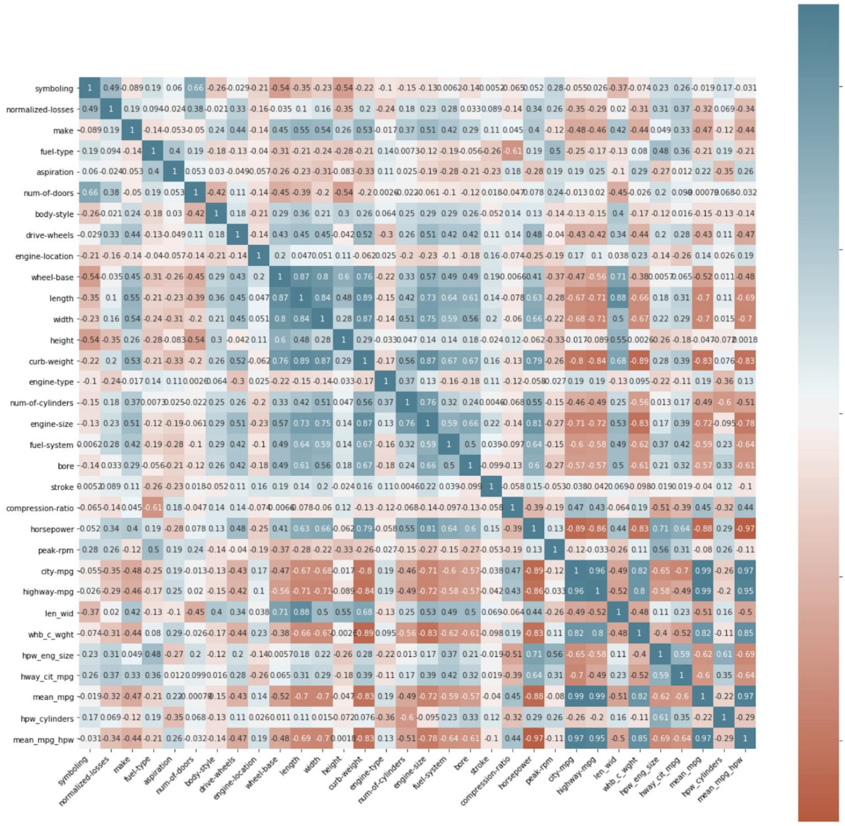
Next step, we normalize our numerical features using log transformation. We do not have normal distribution in all of our numerical features, thus we use the numpy function log1p which applies log(1+x) to all elements of the column to handle that. Let's visualize the new numerical distributions:

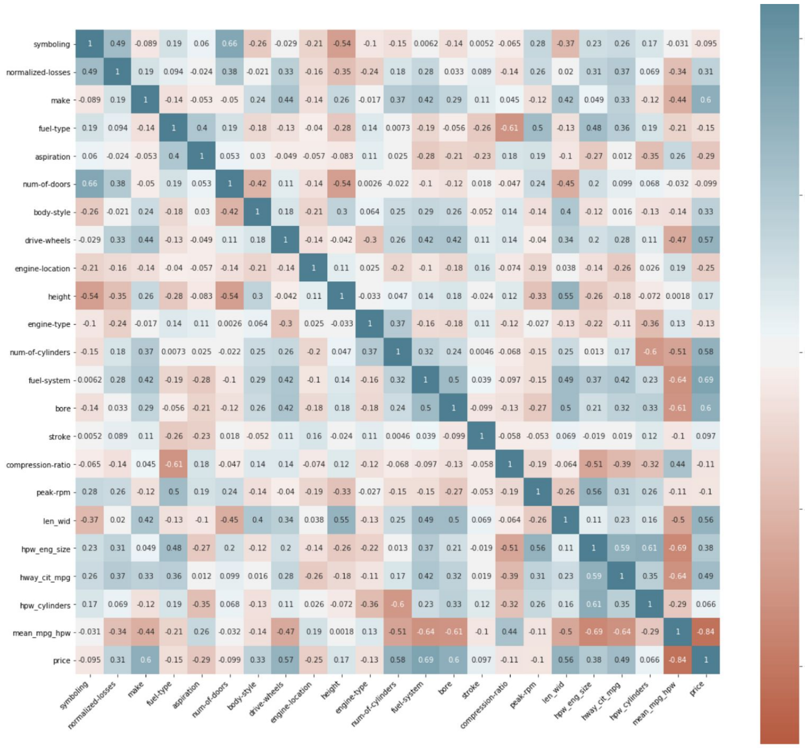


Now with the first outliers removed and our first continuous features normalized, we will create new features, treat outliers, do log transformation and do a correlation analysis to remove high correlated features in order to reduce the dimensionality of the data. Specifically, we remove some of high correlations by creating new features with relation between high correlated features:



Lastly, we have a great number of features and clearly have some features that are highly correlated with each other, we will select and remove them using a high correlation matrix. The purpose of removing high correlated features is to have less features to process and reduce chances of overfitting. The heatmaps below are before and after we remove high correlated features:


symboling
normalized-losses
make
fuel-type
aspiration
num-of-doors
body-style
drive-wheels
engine-location
wheel-base
length
width
height
curb-weight
engine-type
num-of-cylinders
engine-size
fuel-system
bore
stroke
compression-ratio
horsepower
peak-rpm
city-mpg
highway-mpg
len_wid
whb_c_wght
hpw_eng_size
hway_cit_mpg
mean_mpg
hpw_cylinders
mean_mpg_hpw


symboling
normalized-losses
make
fuel-type
aspiration
num-of-doors
body-style
drive-wheels
engine-location
height
engine-type
num-of-cylinders
fuel-system
bore
stroke
compression-ratio
peak-rpm
len_wid
hpw_eng_size
hway_cit_mpg
hpw_cylinders
mean_mpg_hpw
price