Car Price Report

HarvardX Data Science Professional Certificate: PH125.9x Capstone Project 2

Mateus Spencer

2024-04-19

1. Introduction/Overview

1.1. Project Overview

This project aims to explore, clean, and analyze a dataset of used car sales in the UK. The dataset, sourced from Kaggle: 100,000 UK Used Car Data set, contains information about 100,000 used cars from various makes including Audi, BMW, Ford, Hyundai, Mercedes, Skoda, Toyota, Vauxhall, and Volkswagen.

The code for the project can be found on GitHub

The primary goal of this project is to build and evaluate predictive models that can effectively predict car sales. This involves several steps, starting with initial data exploration to understand the structure and characteristics of the data, followed by data cleaning to handle missing values, outliers, and any inconsistencies in the data. Feature engineering is then performed to create new variables that can improve the performance of the predictive models. The data is then split into training and testing sets, and various machine learning models are built and evaluated using cross-validation. The performance of each model is assessed using appropriate metrics, and the best model is selected based on these metrics. This report documents the entire process, providing a detailed overview of the steps taken and the results obtained. It serves as a comprehensive guide to the project, offering insights into the data and the predictive models built.

2. Data Exploration

This section is dedicated to exploring the dataset, which is a critical step in understanding the data we're working with. We will only be using the data files regarding the makes of the cars and not the cclass or focus datasets as they are in their respective make datasets.

We first check if all the datasets have the same columns and column names.

```
## [1] "tax(£)"
```

As we see one of them has a different column name, we will change it to match the others so we don't get two columns that should actually be the same when we join the datasets.

Before combining the datasets, we will add a column to each dataset to identify the make of the car as this might be beneficial for the model performance.

Now we can combine all the datasets into one.

Before cleaning the data we will create a final holdout test so that we can use it to test our final selected model on a completly sepparate and independent dataset.

2.1. Data Overview

The dataset contains 99187 rows and 10 columns. Here we can look at the first few rows of the dataset.

model	year	price	transmission	mileage	fuelType	tax	mpg	engineSize	make
A1	2017	12500	Manual	15735	Petrol	150	55.4	1.4	Audi
A6	2016	16500	Automatic	36203	Diesel	20	64.2	2.0	Audi
A1	2016	11000	Manual	29946	Petrol	30	55.4	1.4	Audi
A4	2017	16800	Automatic	25952	Diesel	145	67.3	2.0	Audi
A3	2019	17300	Manual	1998	Petrol	145	49.6	1.0	Audi
A1	2016	13900	Automatic	32260	Petrol	30	58.9	1.4	Audi

Here is a brief description of each variable in the dataset.

	Data Type	Description
model	character	Model of the car
year	numeric	Year of manufacture
price	numeric	Price of the car
transmission	character	Type of transmission
mileage	numeric	Mileage of the car
fuelType	character	Type of fuel used
tax	numeric	Tax on the car
mpg	numeric	Miles per gallon
engineSize	numeric	Engine size
make	character	Make of the car

2.2. Data Cleaning

In this section, we will clean our dataset by handling missing values, removing duplicates, and converting data types if necessary.

There are 0 missing values, and 1475 duplicate rows so we need to adress these issues.

We will remove the duplicate rows that might have appeared during the data gathering process.

Next, we convert some of the variables into factors, namely: model, make, transmission and fuelType.

We are now ready to move on to the next step in the data analysis process.

2.3. Exploratory Data Analysis

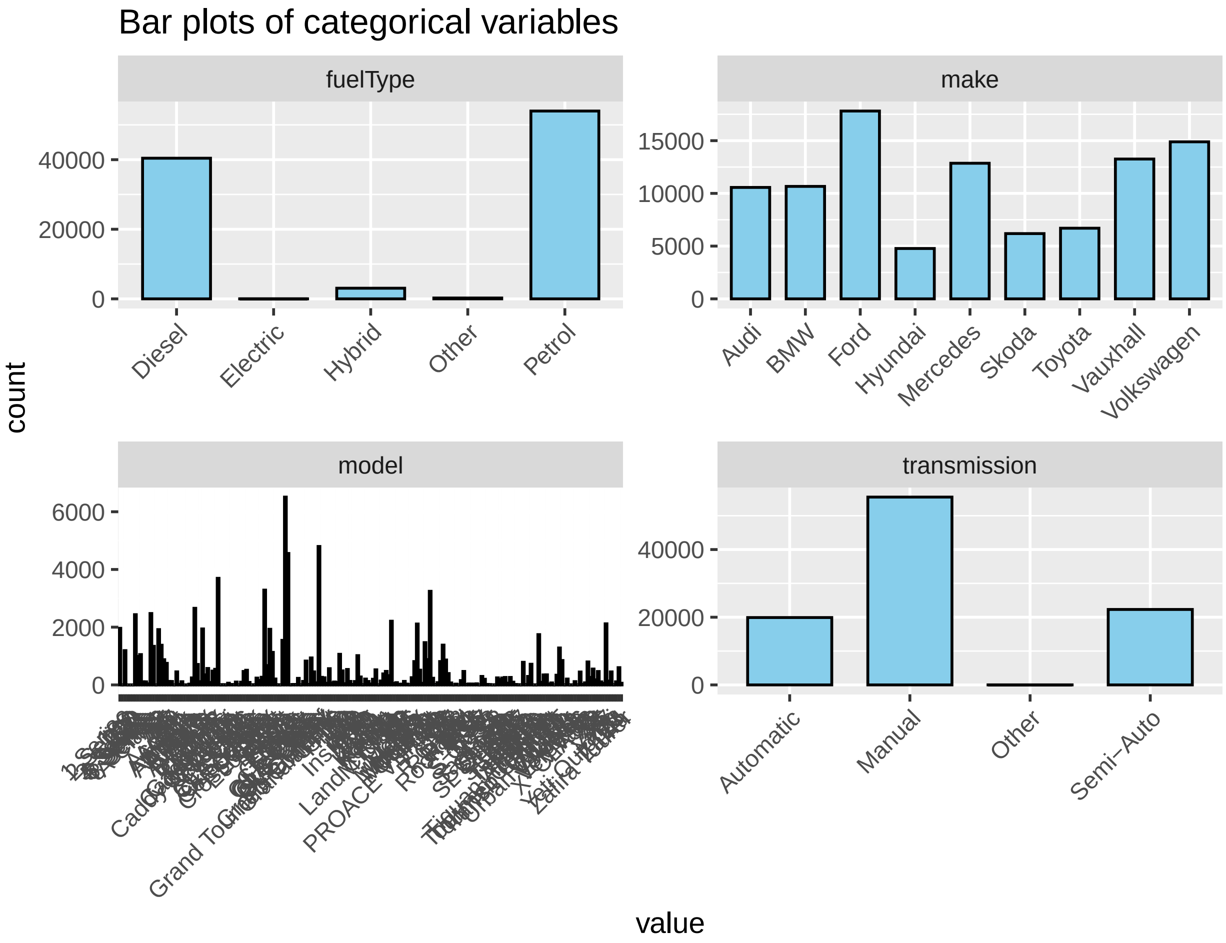
Lets try to understand the underlying structure of the data, identify outliers and anomalies, discover patterns, spot relationships among variables and test assumptions.

2.3.1. Univariate Analysis

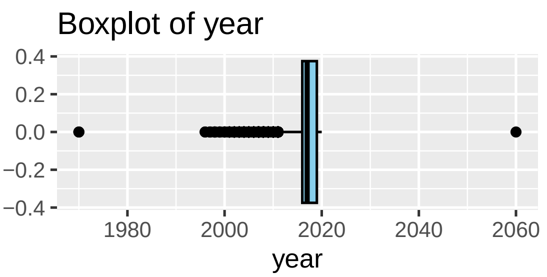
We can start by looking at the distribution of the categorical variables.

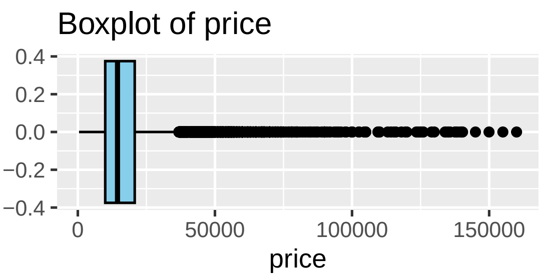
- There are 9 unique car makes and 195 unique models in the dataset.
- There are 4 types of transmission: Manual, Automatic, Semi-Auto, Other and 5 types of fuel: Petrol, Diesel, Hybrid, Other, Electric.

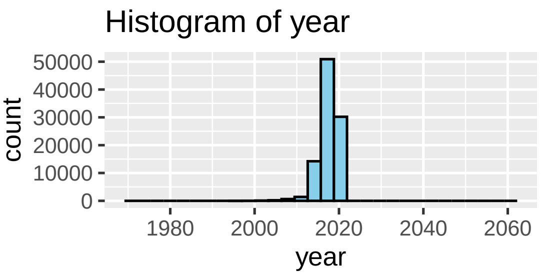
Here we can inspect them visually.

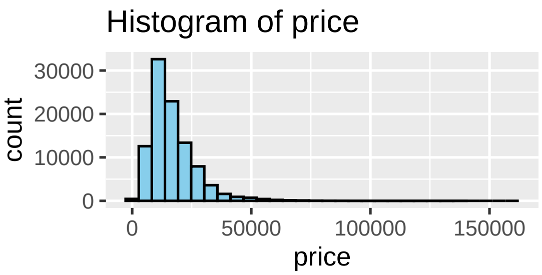


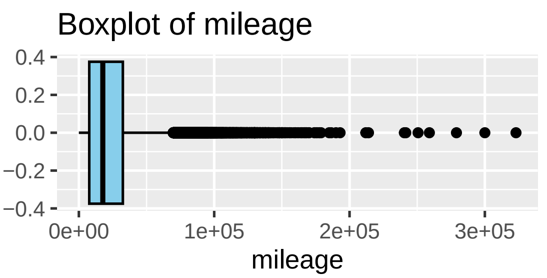
For the numerical variables we will plot histograms and boxplots to visualize the distribution of the data and identify any outliers.

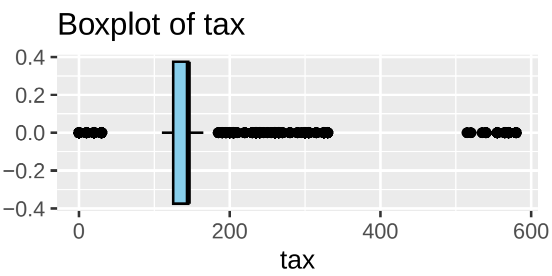

Boxplot of year

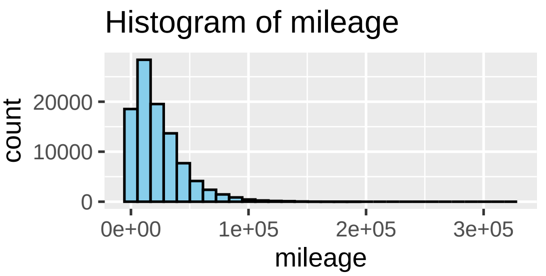

Boxplot of price

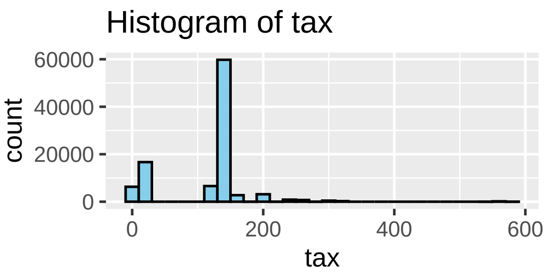

Histogram of year

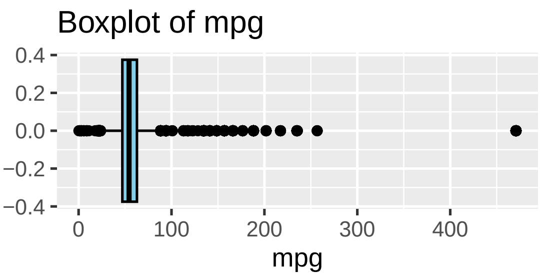

Histogram of price

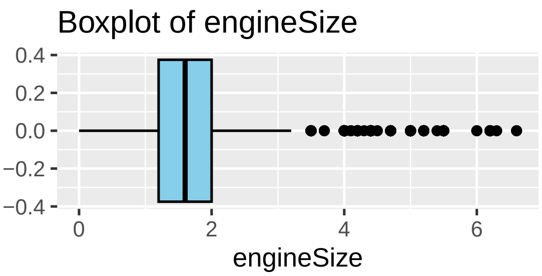

Boxplot of mileage

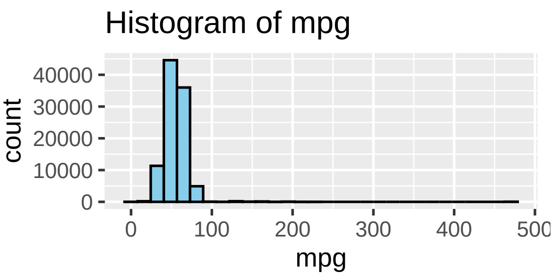

Boxplot of tax

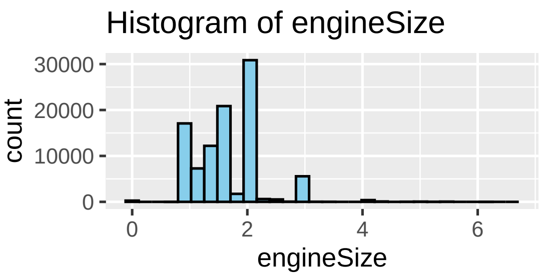

Histogram of mileage


Histogram of tax


Boxplot of mpg


Boxplot of engineSize


Histogram of mpg

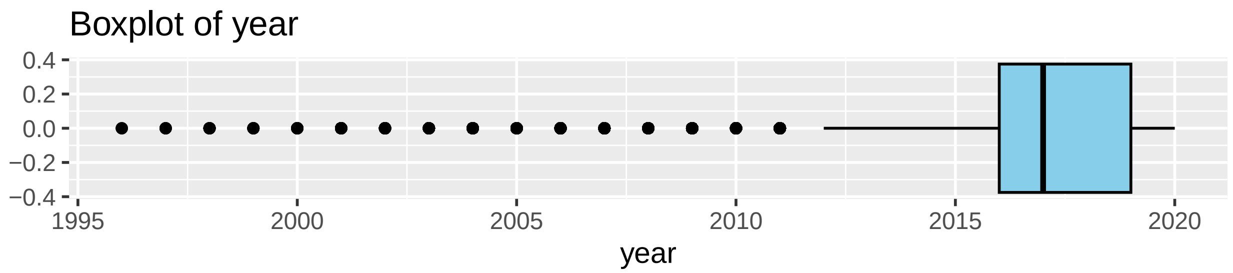

Histogram of engineSize

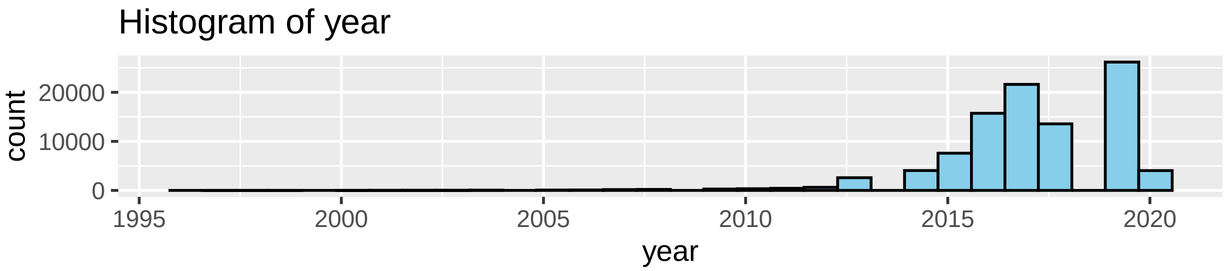
Here is a table that summarizes some of the attributes of the numerical variables.

Variable	Mean	Median	Mode	SD	Min	Max	Skewness	Kurtosis
year	2017.066870	2017.0	2019.0	2.122993	1970.0	2060.0	-1.9118596	16.720860
price	16773.487555	14470.0	9995.0	9868.552222	450.0	159999.0	2.3638117	15.464609
mileage	23219.475499	17682.5	10.0	21060.882302	1.0	323000.0	1.7795983	8.490749
tax	120.142408	145.0	145.0	63.357250	0.0	580.0	0.0360537	6.940103
mpg	55.205623	54.3	60.1	16.181659	0.3	470.8	8.9729167	204.523316
engineSize	1.664913	1.6	2.0	0.558574	0.0	6.6	1.3121561	7.415400

We see there are some outliers so we will adress each variable individually and clean the data.

Year There is at least one clear error in the year since one car is from the future 2060, so we will remove entries whose year is greater than 2024 and also the outliers with years below 1990.

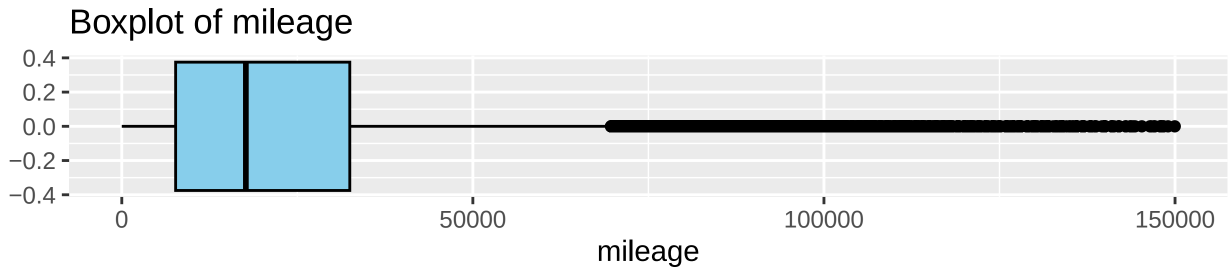


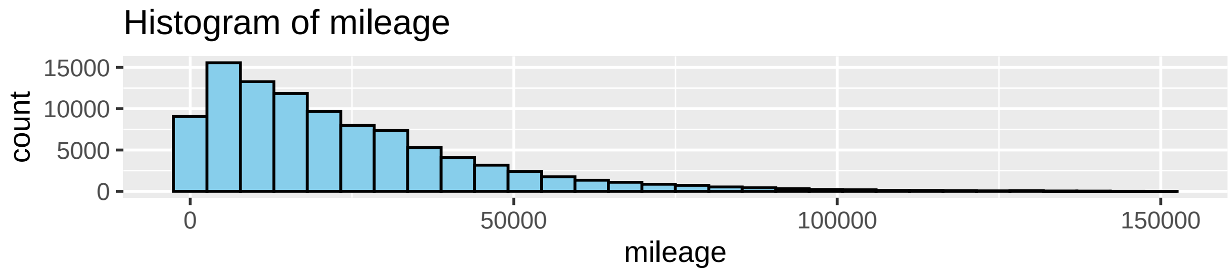


We removerd 3 rows from the dataset.

Mileage In regards to mileage the mean is 2.3219475×10^4, which is affected by the presence of an outlier value of 3.23×10^5 miles.

We will therefore remove these outliers (greater than 150000) to get a more accurate representation of the data.

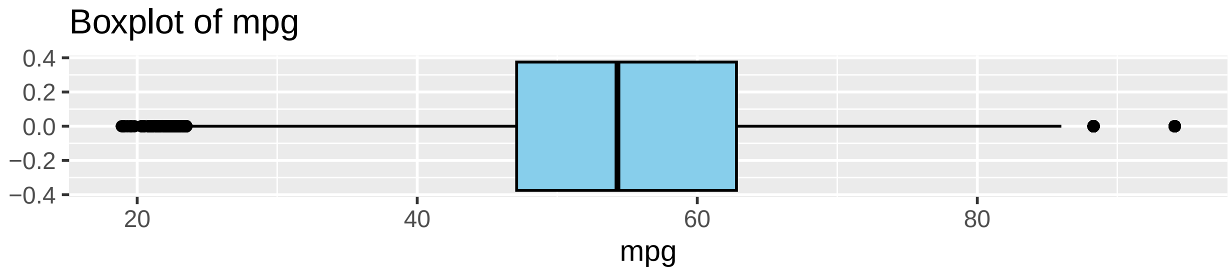


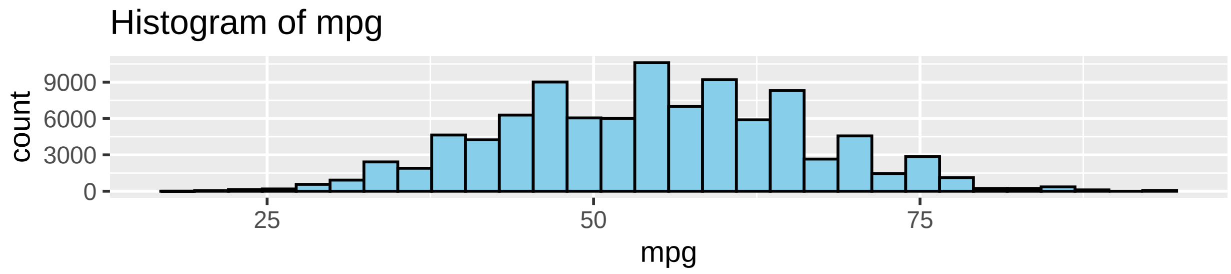


We removerd 50 rows from the dataset.

Miles per Gallon (MPG) Besides the clear outlier with mpg of 470.8, the distribution of the mpg is quite skewed to the right.

We will remove the outliers with mpg greater than 100 and lower than 15.

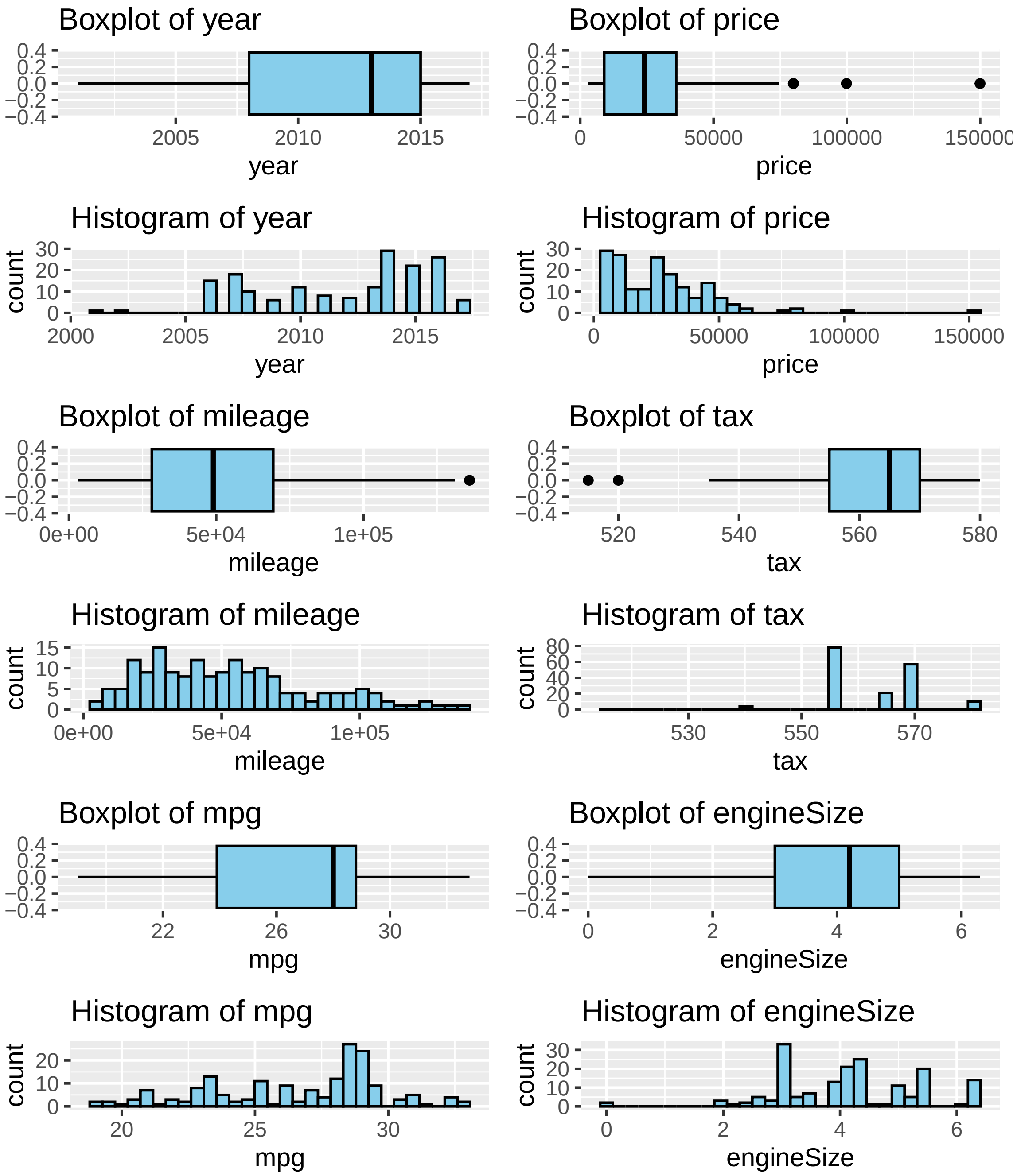




We removerd 0 rows from the dataset.

Tax It looks there are two outlier clusters: tax values over 350 and lower than 100.

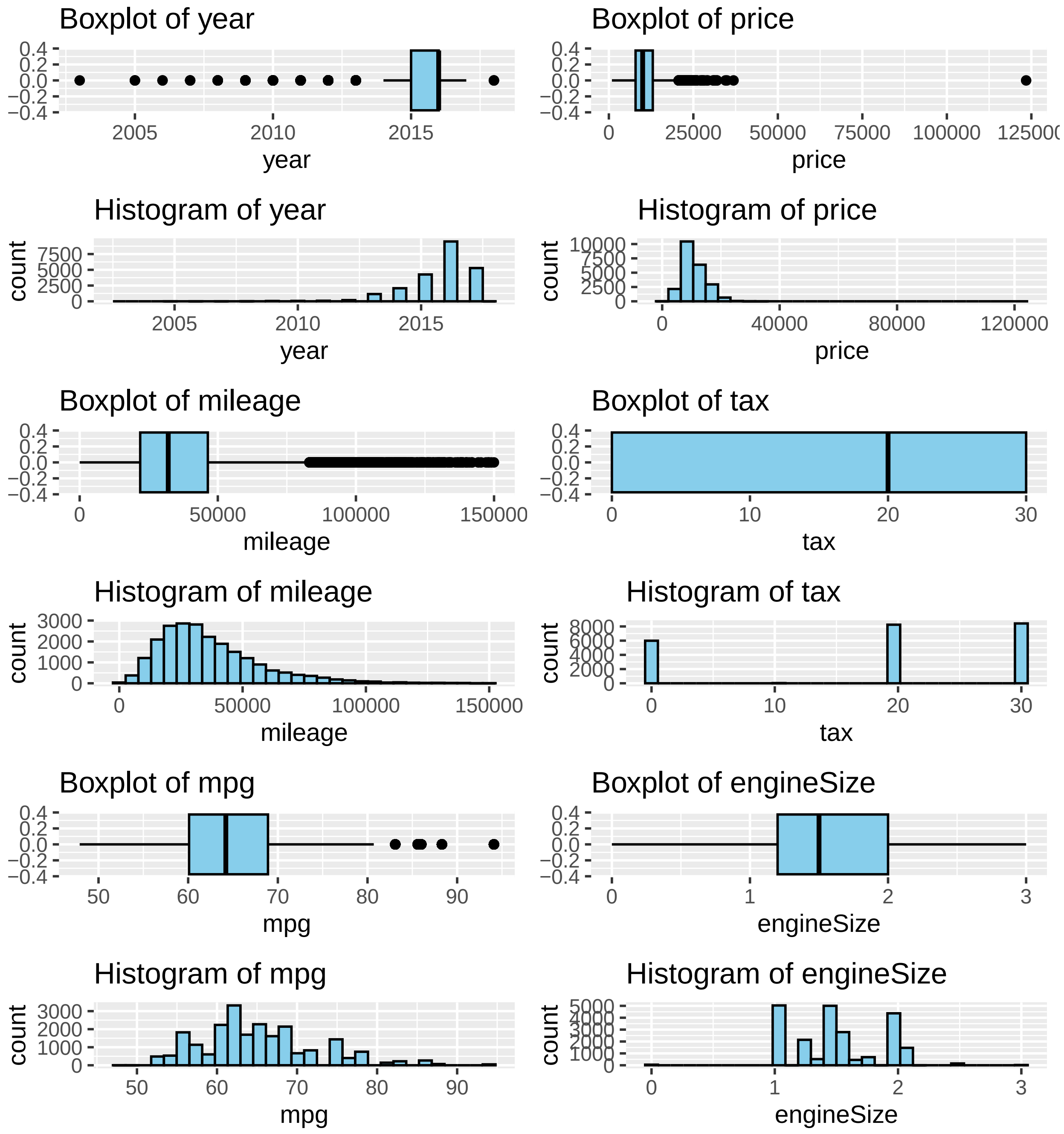
Lets plot the distributions of the observations from the high tax value group.



It looks like that these cars also have quite high values for enginesize and price, with most values above the average (most likely luxury cars).

We will remove these 173 outliers.

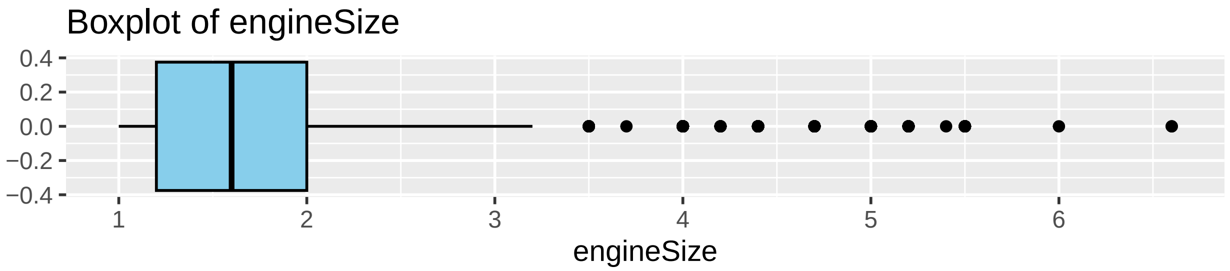
Now looking at the low tax value group.

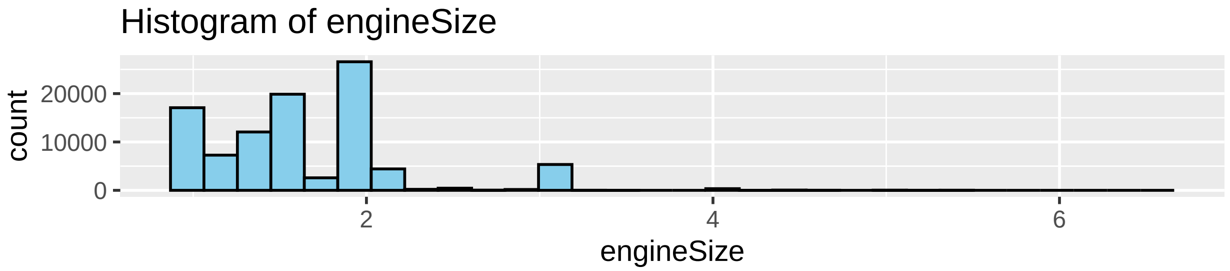


First of all we see there is quite a large number of cars in this cluster: 22677. Excluding a clear outlier, these cars have a small price and the other values are not particularly different from others besides the low value of tax.

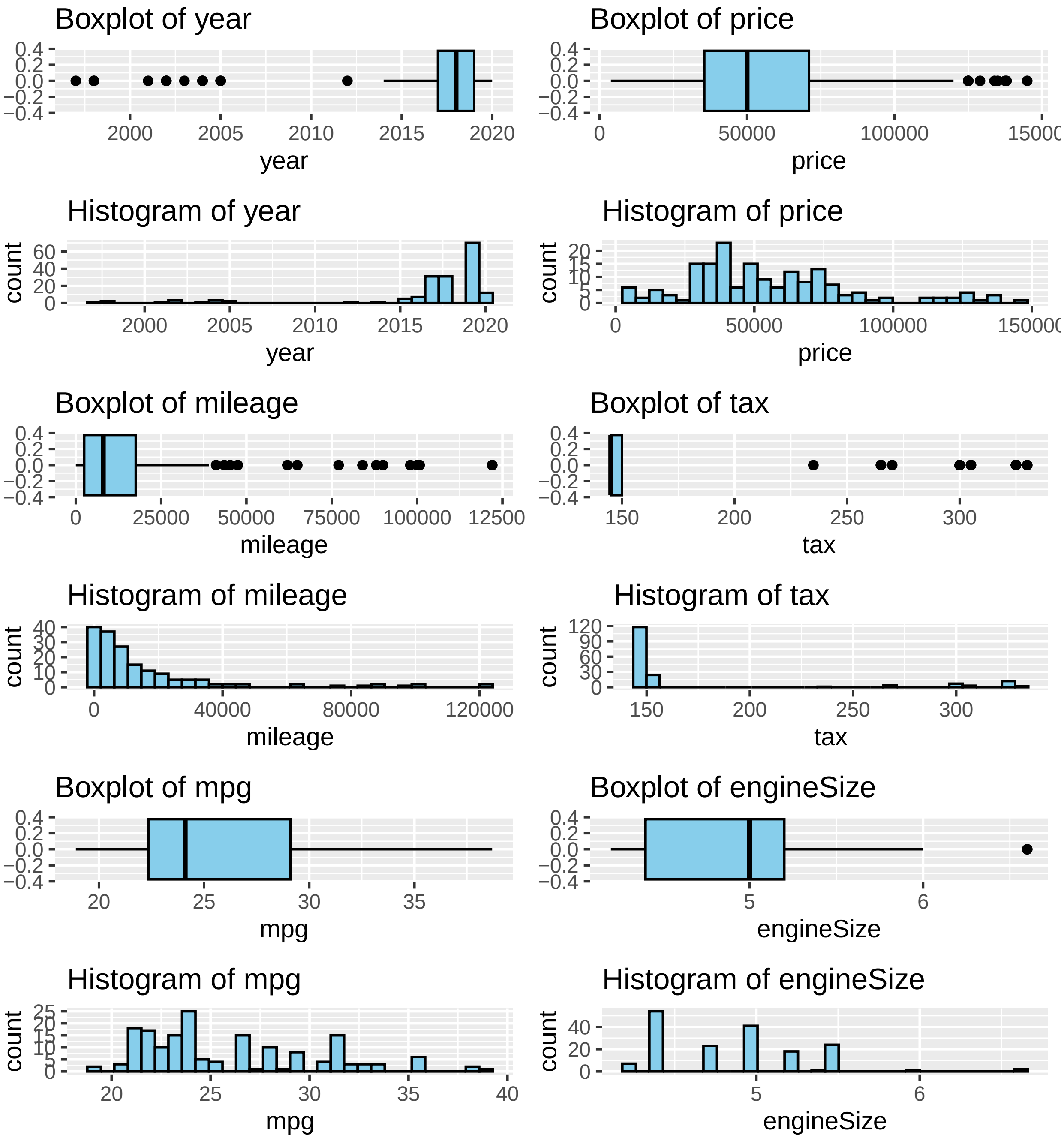
Therefore we will keep these and only remove the outlier with a price of 1.23456×10^5 for a relatively low tax.

Engine Size The Histogram shows us that there are some cars with a reported engine sizze of 0, which is not possible so we will remove them.

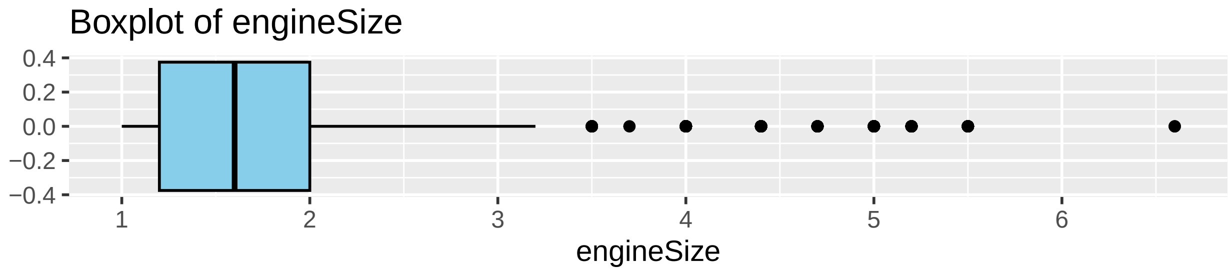


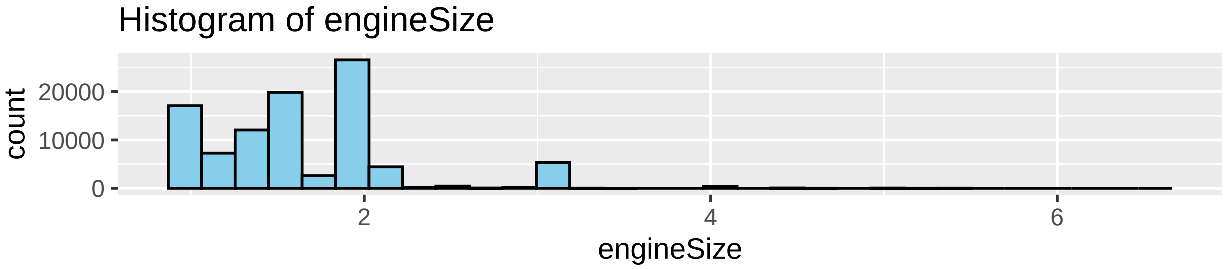


Now looking at the bigger engine sizes we see that there are a small number of cars with engine sizes above 4.



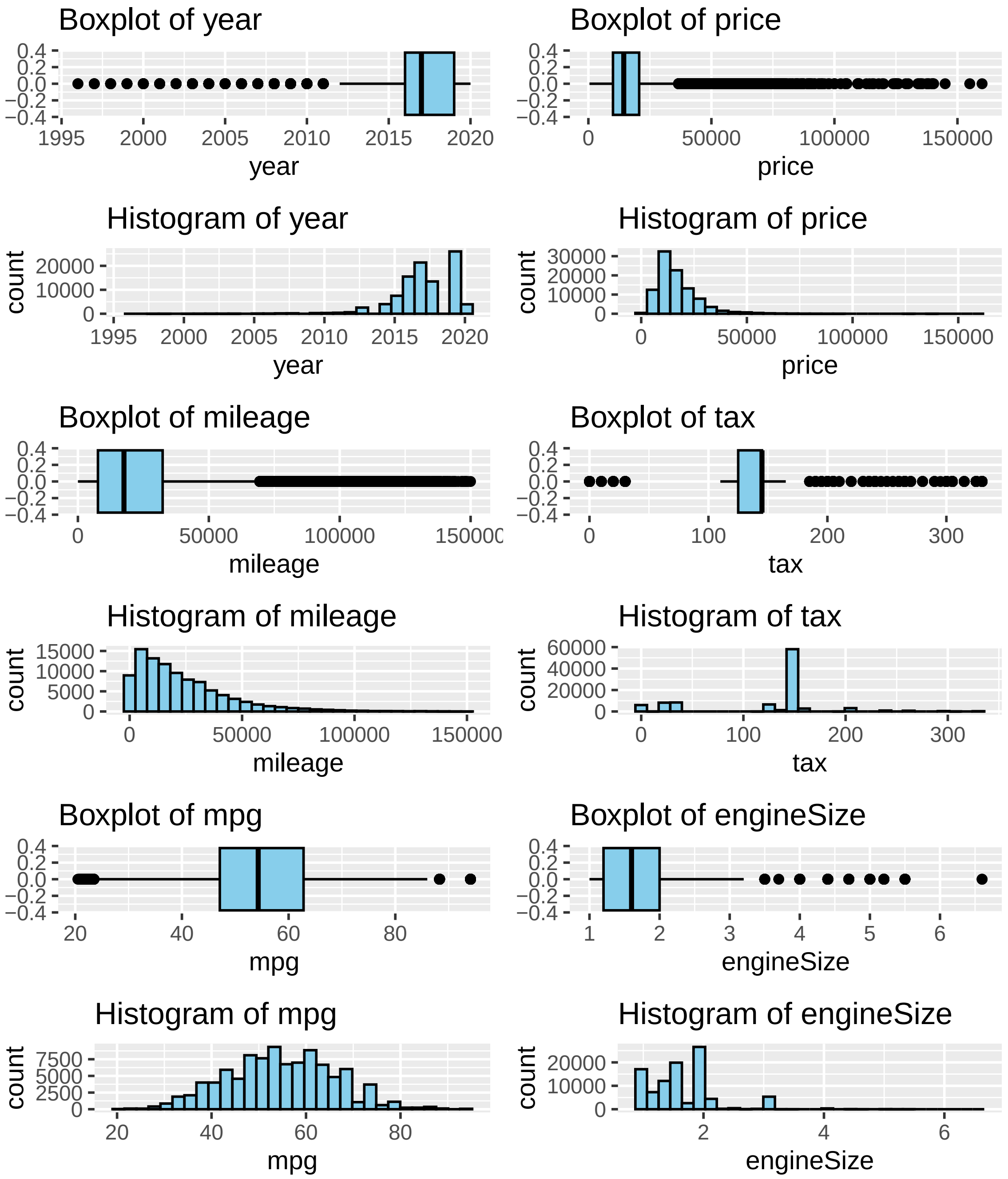
As we can see all of these cars have a high tax, but most of them are around 150 tax, the rest are outliers with a very high tax, so we eill remove just these.





Price The distribution, like the mileage is a long tail distribution, which we could try to normalize with a log transformation to improve model performnce but will leave it as is for now.

After Outlier Removal We have removed 1083 outliers from the dataset which represent 1.11% of the original data.


Boxplot of year

Boxplot of price

Histogram of year

Histogram of price

Boxplot of mileage

Boxplot of tax

Histogram of mileage

Histogram of tax

Boxplot of mpg

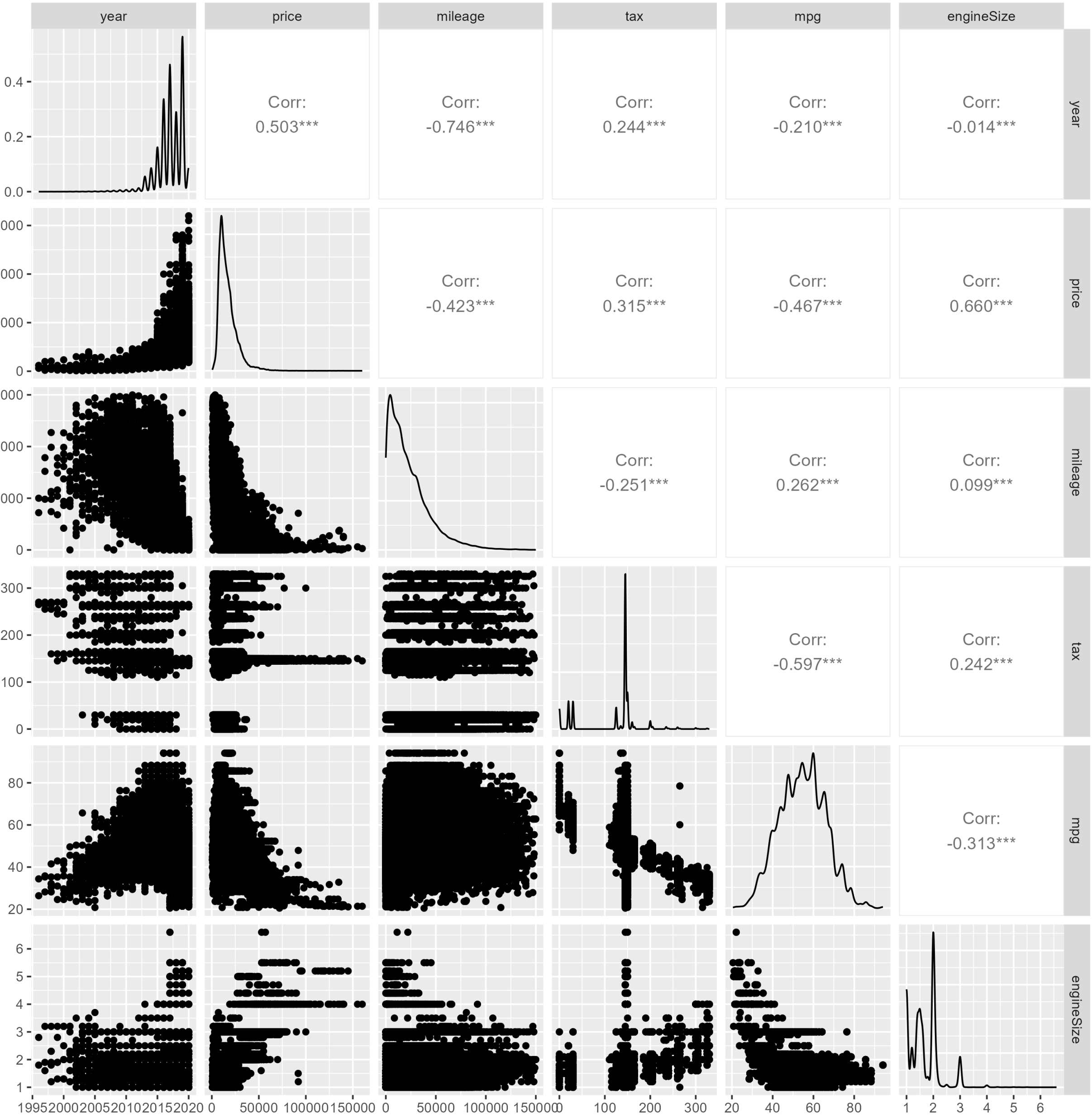
Boxplot of engineSize

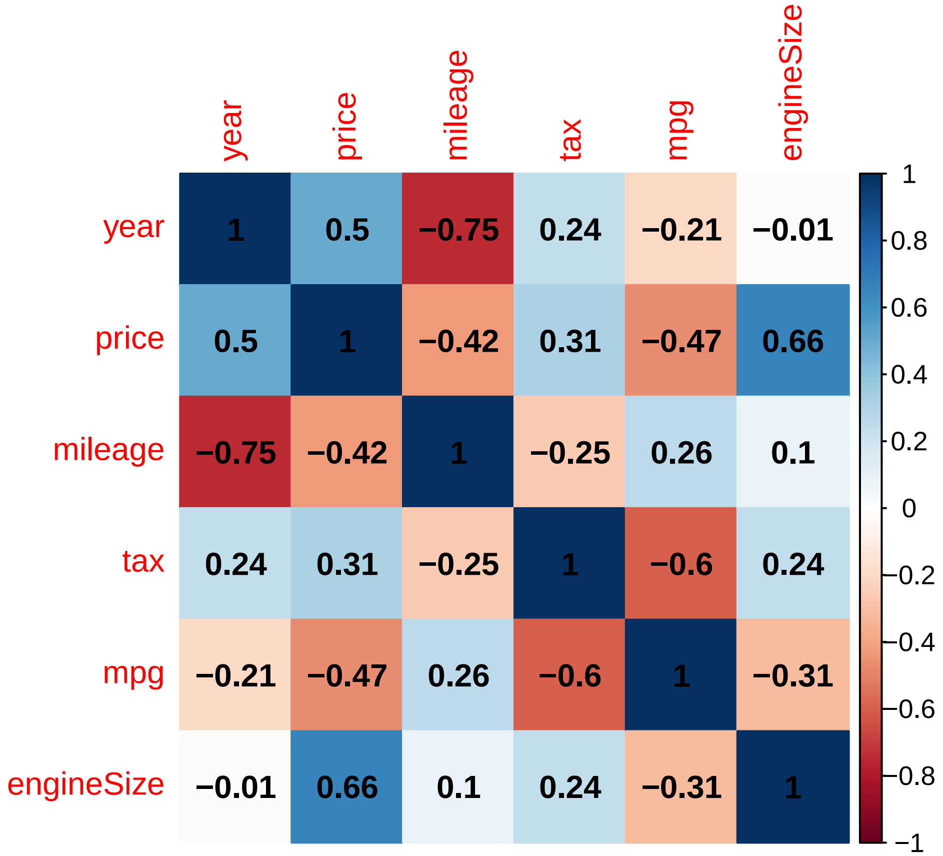
Histogram of mpg

Histogram of engineSize

2.3.2. Bivariate/Multivariate Analysis

We can also look at the relationships between variables and see how they affect each other.


year
price
mileage
tax
mpg
engineSize
Corr:
0.503***
Corr:
-0.746***
Corr:
0.244***
Corr:
-0.210***
Corr:
-0.014***
Corr:
-0.423***
Corr:
0.315***
Corr:
-0.467***
Corr:
0.660***
Corr:
-0.251***
Corr:
0.262***
Corr:
0.099***
Corr:
-0.597***
Corr:
0.242***
Corr:
-0.313***
0.4
0.2
0.0
150000
100000
50000
0
150000
100000
50000
0
300
200
100
0
80
60
40
20
6
5
4
3
2
1
1995
2000
2005
2010
2015
2020
0
50000
100000
150000
0
50000
100000
150000
0
100
200
300
20
40
60
80
1
2
3
4
5
6



From the heatmap, we can say the following about the price:

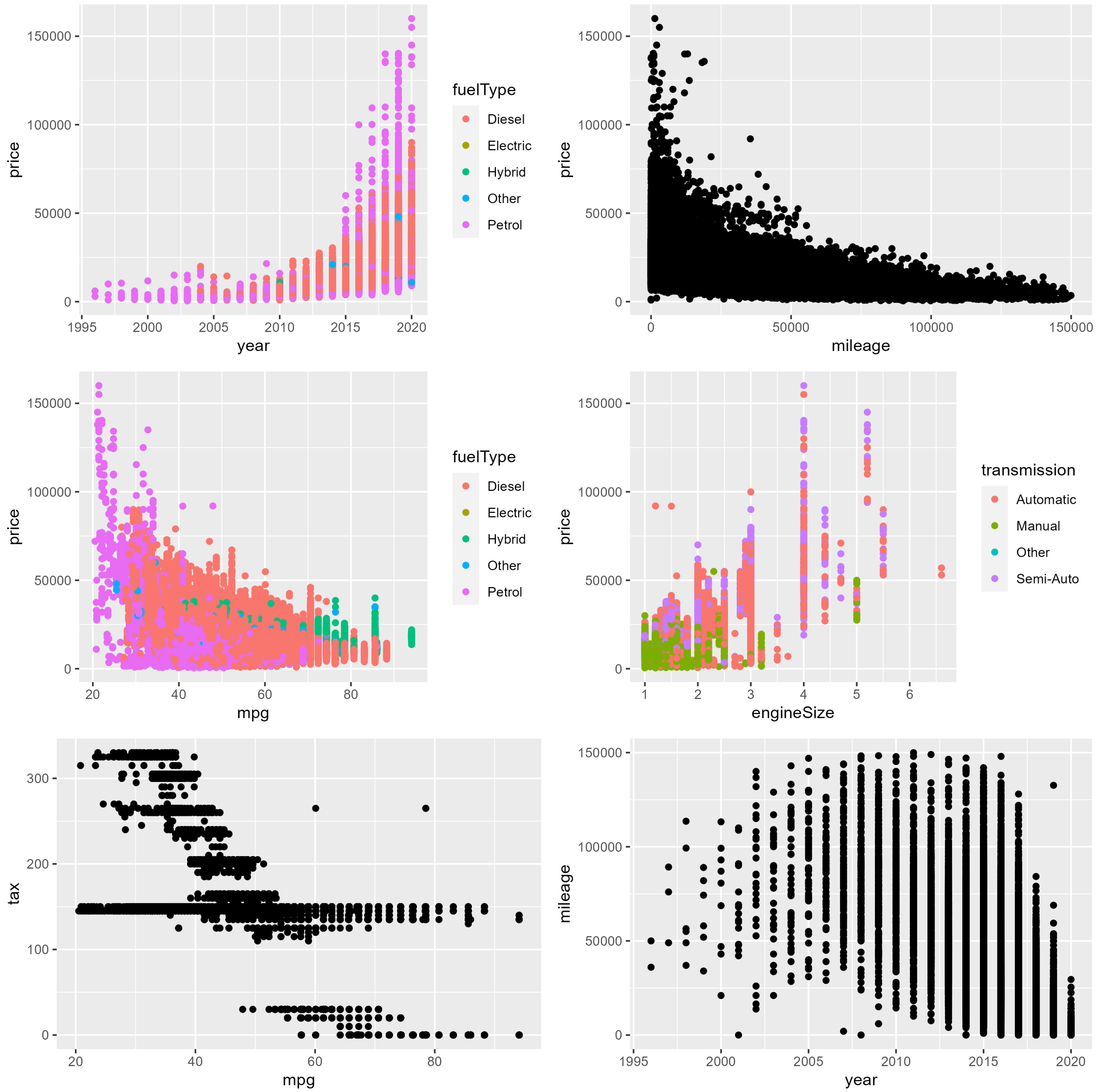
- There is a moderate positive correlation (0.5) between the price and the year. This means that cars with a higher year tend to be more expensive.
- There is a moderate negative correlation (-0.42) between the price and the mileage. This means that cars with higher mileage tend to be cheaper.
- There is a small positive correlation (0.31) between the price and the tax. This means that cars with higher tax tend to be more expensive.
- There is a moderate negative correlation (-0.47) between the price and the miles per gallon. This means that cars with higher mpg tend to be cheaper.
- There is a significant positive correlation (0.66) between the price and the engine size. This means that cars with higher engine size tend to be more expensive.

In regards to other relations between variables:

- There is a significant negative correlation (-0.74) between the mileage and the year. This means that cars with a lower (older) year tend to have more mileage.
- There is a significant negative correlation (-0.6) between the miles per gallon and the tax. This means that cars with high mpg tend to have lower tax.

We should investigate Multivariate outliers for the features with high correlations:

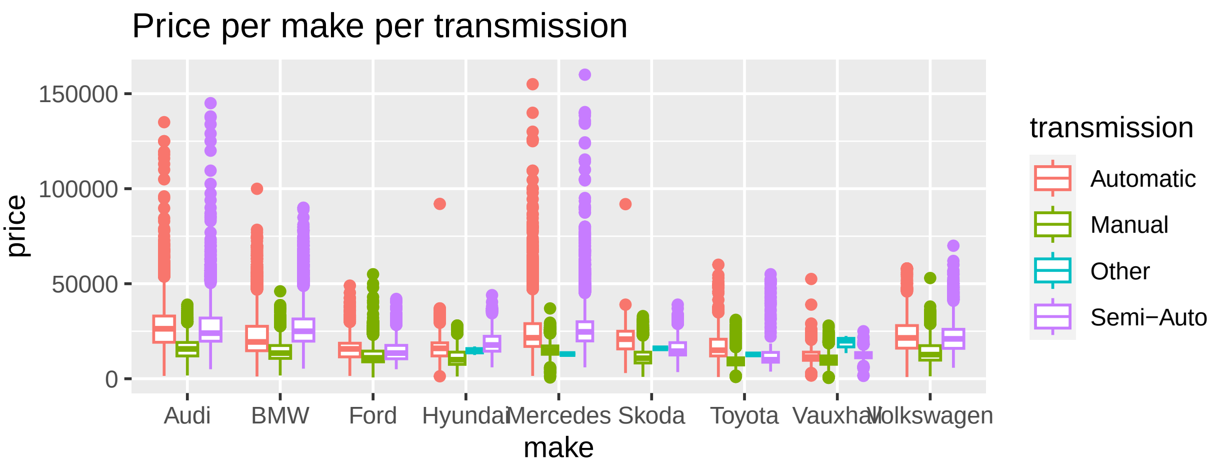
- price-year
- price-mileage
- price-mpg
- price-enginesize
- tax-mpg
- mileage-year

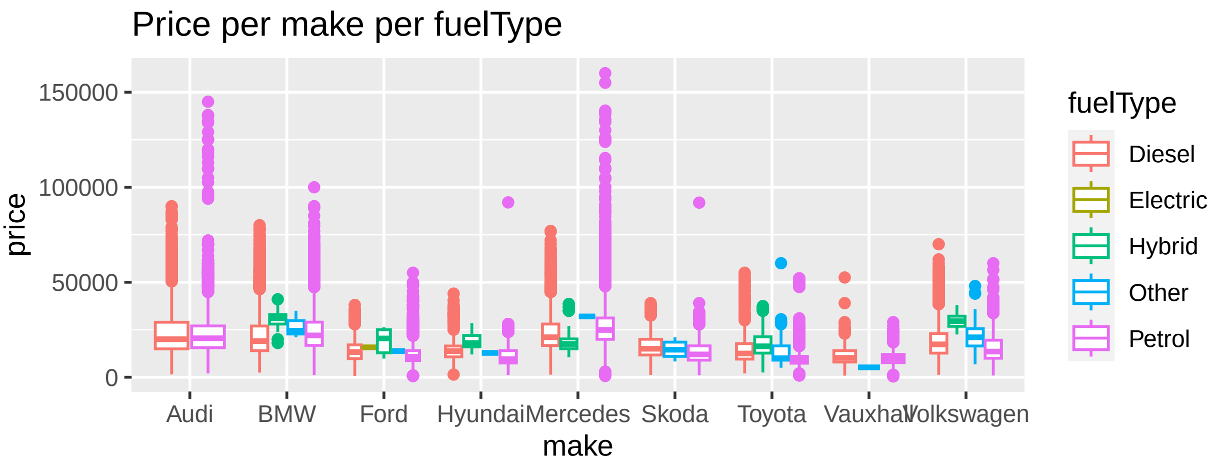


There don't seem to be many outliers left, so we will only remove the two outliers that have a mpg of 60 or higher but a tax of about 250.

We have now removed 1086 outliers from the dataset which represent 1.11% of the original data.

We can now look at the relationships between the categorical variables and the price of the cars.



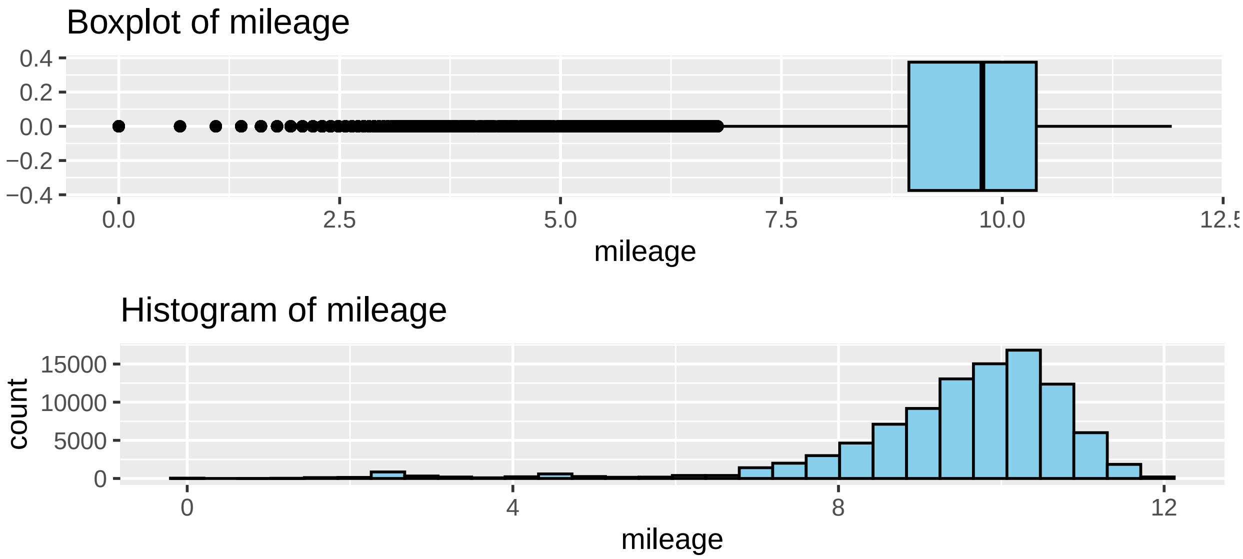


2.4. Feature Engineering & Data Preparation

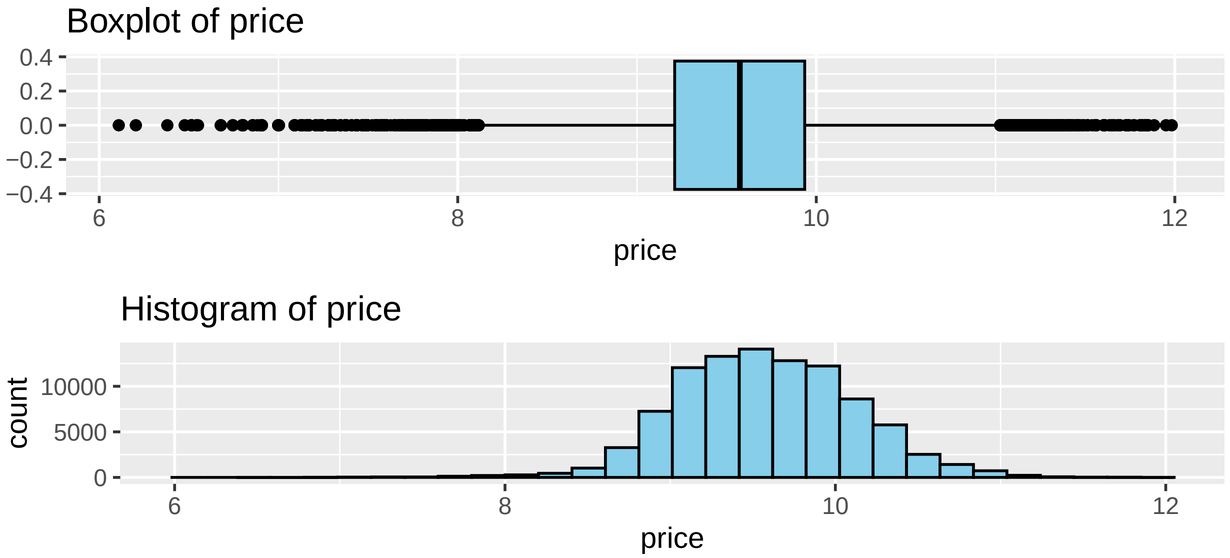
First we'll start by encoding the categorical variables into numerical values so that they can be used in the models.

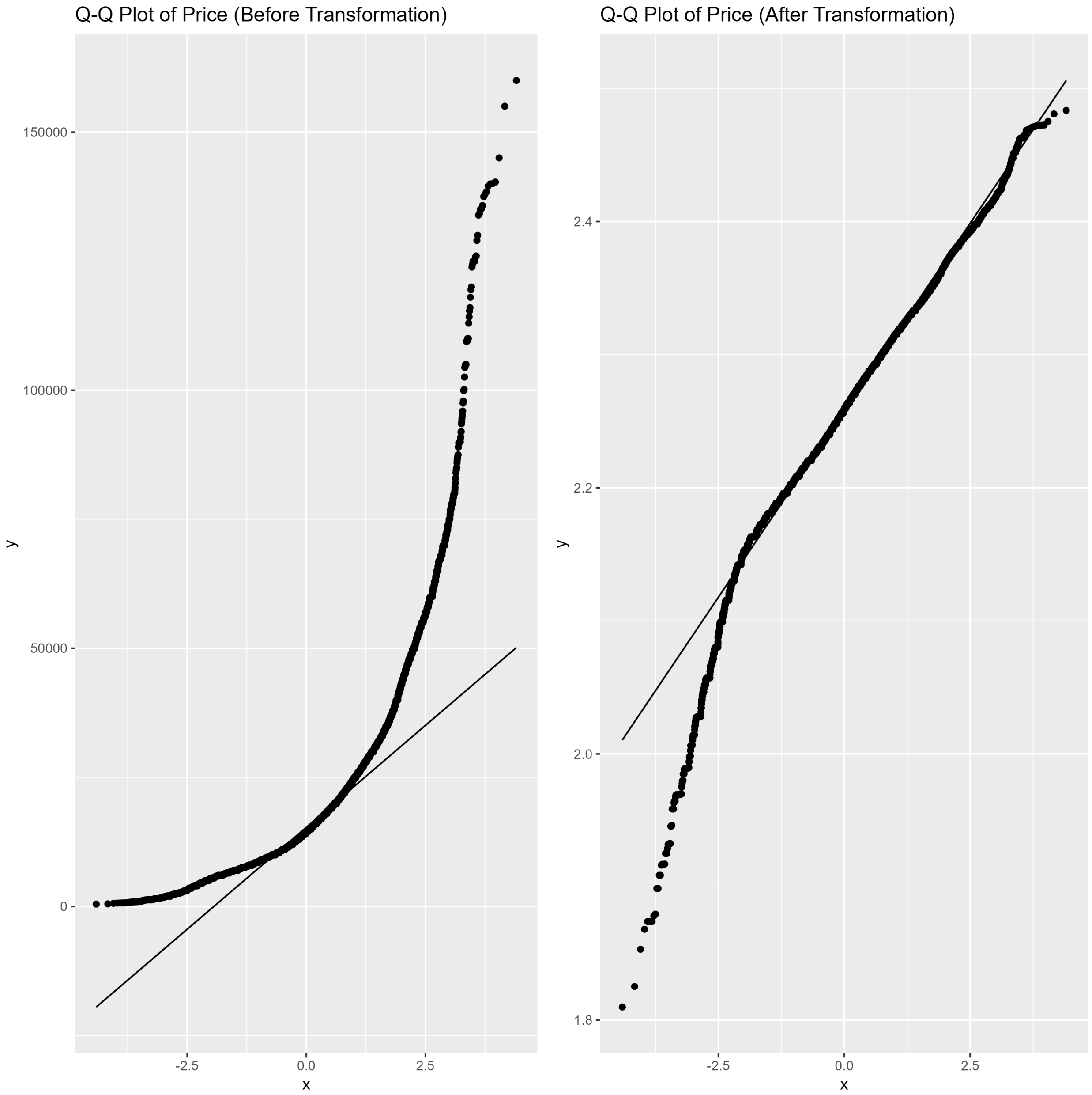
model	year	price	transmission	mileage	fuelType	tax	mpg	engineSize	make
13	2017	12500	2	15735	5	150	55.4	1.4	1
18	2016	16500	1	36203	1	20	64.2	2.0	1
13	2016	11000	2	29946	5	30	55.4	1.4	1
16	2017	16800	1	25952	1	145	67.3	2.0	1
15	2019	17300	2	1998	5	145	49.6	1.0	1
13	2016	13900	1	32260	5	30	58.9	1.4	1

As mentioned before we will try to perform a log transformation to normalize the values of mileage and price to improve performance in models that assume normality.



Clearly the log transformation has not improved the distribution of the mileage variable. We will keep it as it was.





For the price variable, the log transformation has improved the distribution and made it more normal. We will keep this transformation.

We are now ready to start building the price prediction models.

2.5. Modeling Approaches

We will be building several models to predict the price of the cars and evaluate the performance of each model using appropriate metrics such as RMSE, R2 and MAE and select the best model based on these metrics.

- RMSE: Root Mean Squared Error is the square root of the average of the squared differences between the predicted and actual values. The smaller the RMSE, the better the model's performance.

- R2: R-squared is a measure of how well the model fits the data. It is a value between 0 and 1, with 1 indicating a perfect fit.
- MAE: Mean Absolute Error is the average of the absolute differences between the predicted and actual values. Like RMSE, the smaller the MAE, the better the model's performance.

Due to the nature of our dataset and the some of the linear relations between variables we observed we'll start by trying linear models and then progress towards more complex.

3. Model Building

We will start by splitting the data into training and testing sets and then build and evaluate the models, starting with a baseline linear regression model and then moving on to more complex models such as Elastic Net and LightGBM.

3.1. Data Splitting: Split the data into training and testing sets.

We will split the data into training and testing sets using an 80/20 split.

3.2. Model 1: Baseline linear regression model

Linear regression is one of the simplest and most widely used statistical techniques for predictive modeling. It aims to model the relationship between a scalar dependent variable y (price) and one or more independent variables (features of the cars) denoted X. The relationship is modeled through a linear function and the unknown model parameters are estimated from the data. This is generally done by minimizing the sum of the squares of the differences between the observed responses in the dataset and those predicted by the linear approximation.

Linear regression benefits from being straightforward to understand and interpret, and it's particularly useful when there is a linear relationship between the inputs and the output. However, it's often too simplistic to capture complex patterns in data unless transformations or interactions are included.

	Model	RMSE	R2	MAE
RMSE	Baseline Linear Regression	4512.403	0.7954028	2909.566

The baseline linear regression model provides a good starting point for predicting car prices, but it may not capture all the complexities in the data.

3.3. Model 2: Regularization - Elastic Net Model

The Elastic Net is a regularized regression method that linearly combines the L1 and L2 penalties of the Lasso and Ridge methods. The model solves a regularized version of the least squares, where the objective function is augmented by adding penalty terms that constrain the size of the coefficients:

- L1 penalty (Lasso): Encourages sparsity which can be useful for feature selection if some features are irrelevant.
- L2 penalty (Ridge): Shrinks the coefficients of correlated predictors towards each other, thus stabilizing the solution.

Elastic Net is particularly useful when there are multiple features that are correlated with each other. The combination of L1 and L2 penalty functions allows Elastic Net to inherit some of Ridge's stability under correlated data and Lasso's ability to select sparse features.

	Model	RMSE	R2	MAE
RMSE	Elastic Net	4498.808	0.7955297	2908.64

However, the Elastic Net model did not improve on the performance of the baseline linear regression model. We will now try a more complex model, the LightGBM model, to see if it can provide better predictions.

3.4. Model 3: Gradient Boosting - LightGBM Model

Gradient boosting is a machine learning technique for regression and classification problems, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees. It builds the model in a stage-wise fashion like other boosting methods do, and it generalizes them by allowing optimization of an arbitrary differentiable loss function.

LightGBM (Light Gradient Boosting Machine) is an efficient and scalable implementation of gradient boosting framework by Microsoft. It uses tree-based learning algorithms designed for speed and performance. LightGBM extends the gradient boosting model by introducing two key innovations:

- Gradient-based One-Side Sampling (GOSS): A technique to filter out the data instances to find a split value, focusing more on those instances that produce larger gradients.
- Exclusive Feature Bundling (EFB): A method to reduce the number of features by combining mutually exclusive features, thus significantly decreasing the number of data dimensions without sacrificing much accuracy.

LightGBM constructs trees leaf-wise (best-first), rather than level-wise like other boosting methods. This makes the model more efficient and generally leads to better model fit as it grows the tree more with the most promising regions.

	Model	RMSE	R2	MAE
RMSE	LightGBM	1989.388	0.9585881	1238.79

As we can see this is quite a good result. We will now compare the results of the three models.

4. Model Evaluation & Results

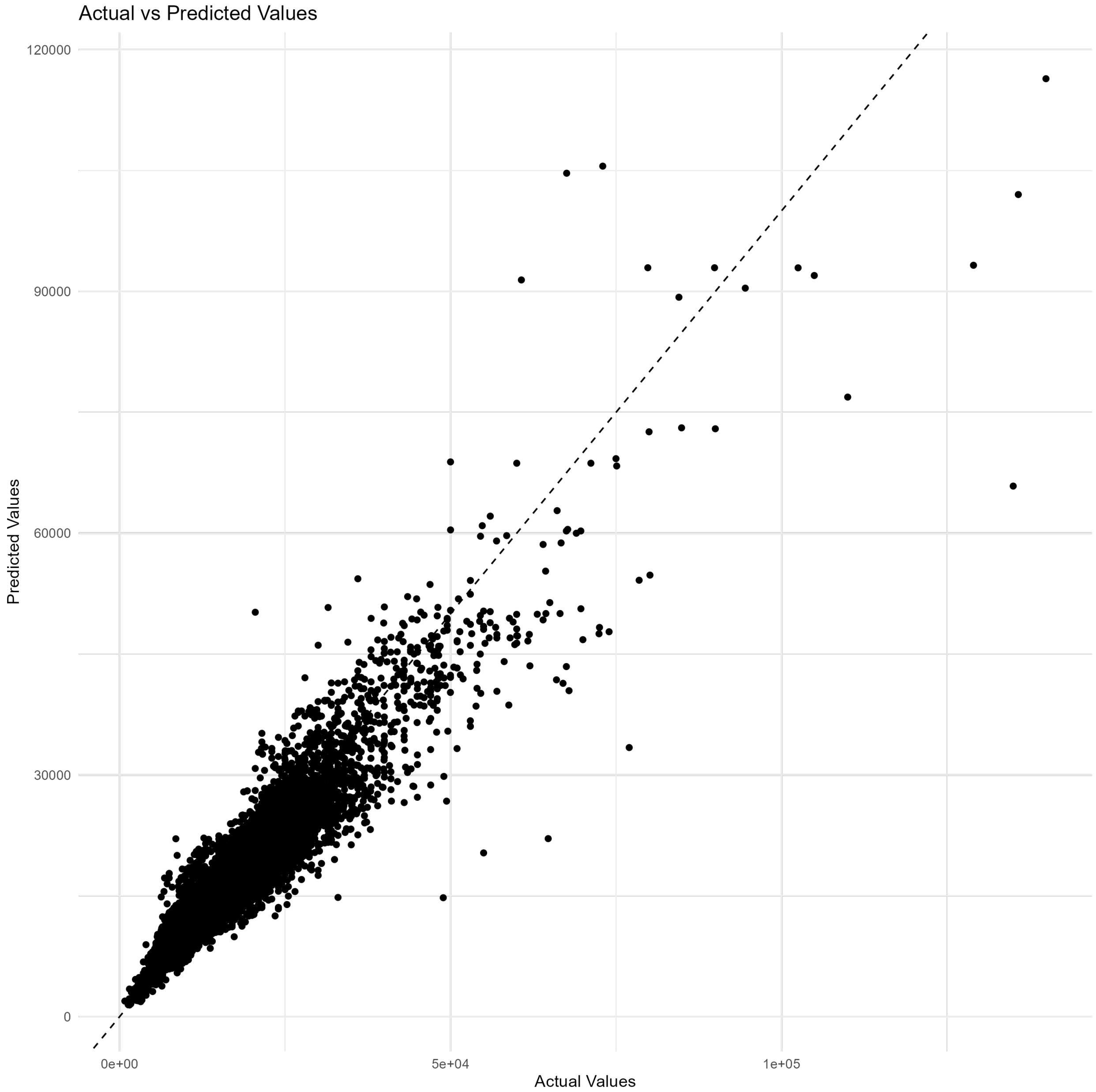
4.1. Results

So we can conclude that the LightGBM model is the best performing model, with the lowest RMSE and highest R-squared value. Let's test it on the holdout set to see how it performs on unseen data.

	Model	RMSE	R2	MAE
RMSE	LightGBM - Holdout Set	3498.203	0.8770959	2233.49

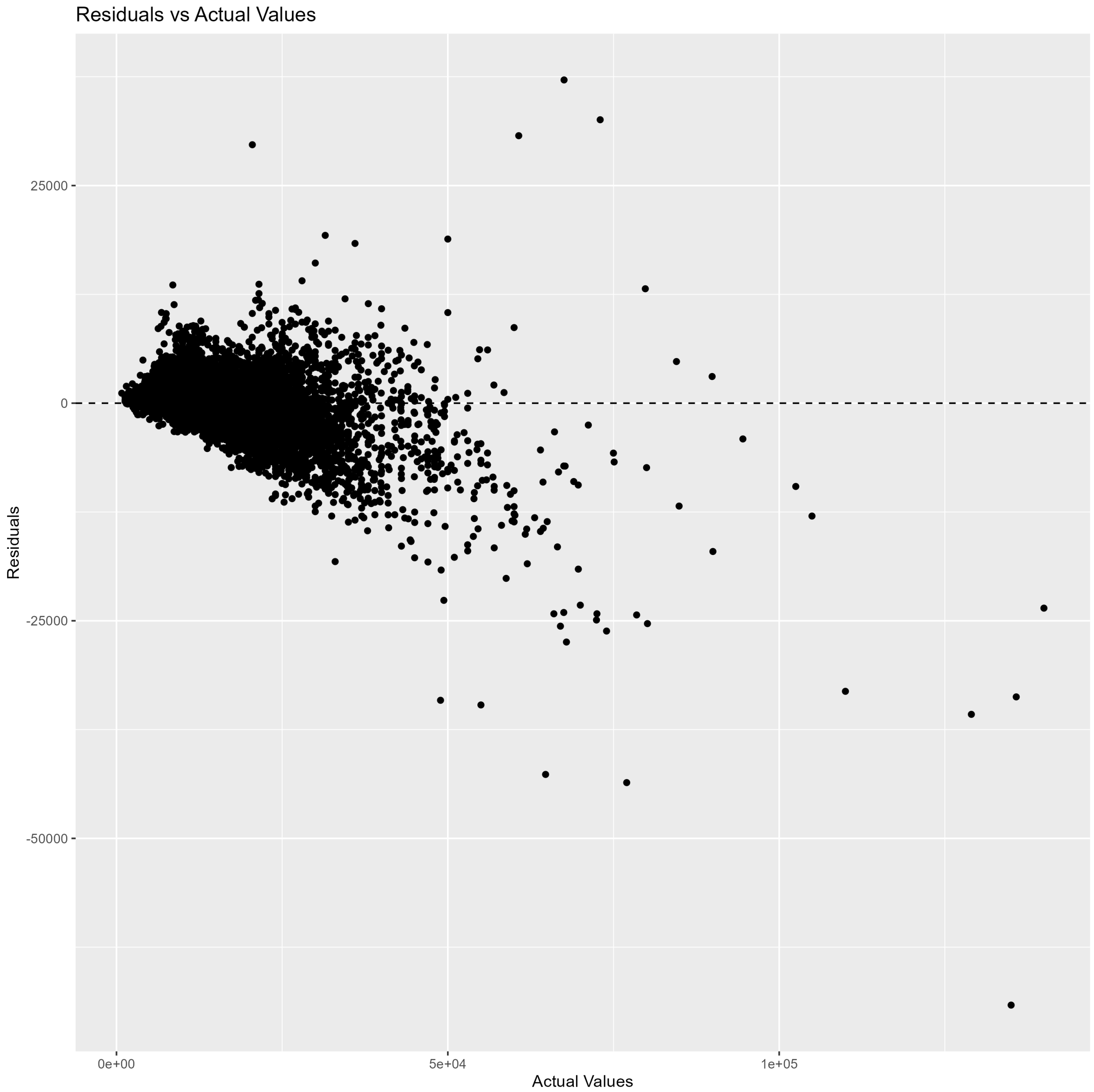
Although not as great as the test set, the LightGBM model still performs well on the holdout set.

Actual vs Predicted Values Plot This plot compares the predicted values against the actual values, showing how closely the predictions align with reality; closer points to the diagonal line indicate more accurate predictions. Divergence from the diagonal line highlights prediction errors.

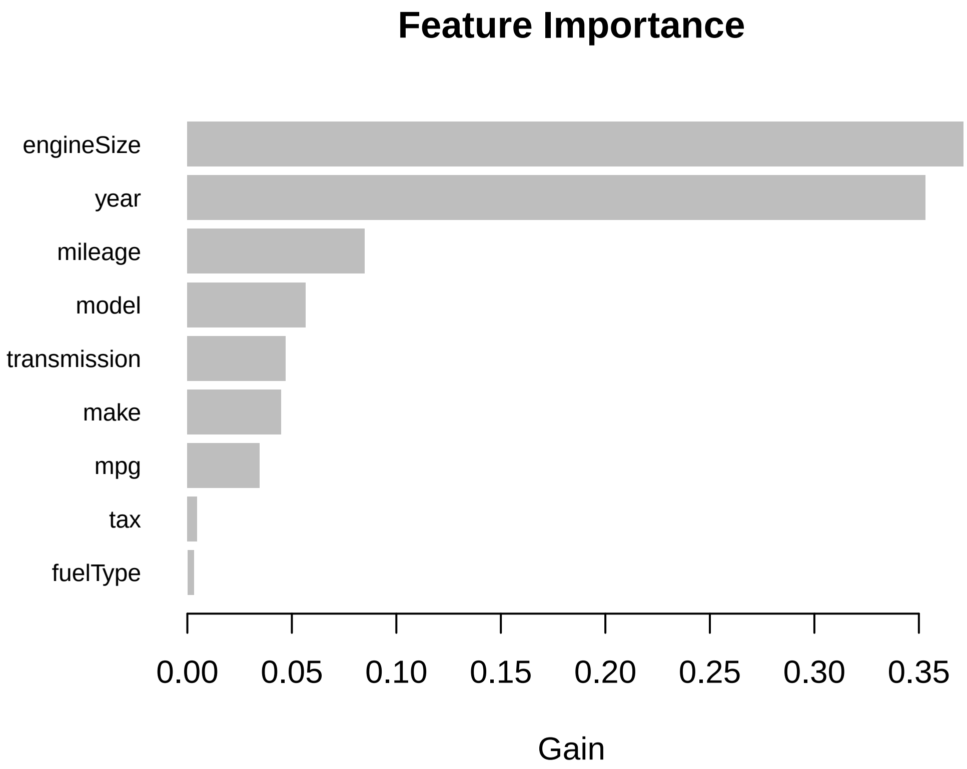


As we see, the LightGBM model has a good fit with the actual values, with most of the points close to the diagonal line. However we see also that the bigger the price the bigger the error.

Residuals vs Actual Values Plot Residuals (differences between actual and predicted values) are plotted against actual values to identify patterns; residuals clustering around zero suggest better model accuracy. Systematic patterns or trends in the residuals may indicate model biases or heteroscedasticity, requiring further model adjustments.



Variable Importance Plot This plot ranks the features based on their importance in the LightGBM model, highlighting which features most influence the model's predictions. Features higher on the plot have a greater impact on the model, guiding feature selection and model refinement.



5. Conclusion

5.1. Summary of the Report

In this report, we analyzed a dataset of used car sales. We explored the relationships between various features and the car price, and built predictive models to estimate the price of a car based on its features. Our models included a baseline linear regression model, an Elastic Net model and a LightGBM model. The LightGBM model performed the best, with the lowest RMSE and highest R-squared value.

5.2. Potential Impact

The results of this analysis could be useful for both buyers and sellers of used cars. Buyers could use our model to estimate the fair price for a used car based on its features, which could help them negotiate a better deal. Sellers could use our model to set a competitive price for their car that reflects its value. Additionally, our analysis could be useful for car dealerships and online marketplaces that deal in used cars.

5.3. Limitations

Our analysis has several limitations. First, our dataset only includes used cars, so our findings may not apply to new cars. Second, our dataset may not be representative of all used cars, as it only includes cars that were listed for sale on a specific website. Third, our models assume that the relationships between the features and the car price are linear, which may not be the case in reality. Finally, our models do not account for factors that could affect the car price but were not included in our dataset, such as the car's condition or the seller's negotiation skills.

5.4. Future Work

In future work, I could improve this analysis in several ways. I could collect more data to make our dataset more representative of all used cars. Trying different models that do not assume a linear relationship between the features and the car price, such as neural network models might also improve results. Including more features in our models, such as the car's condition or the seller's negotiation skills would also improve the prediction capabilities of the models.

6. References

[1] "Introduction to Data Science - Data Analysis and Prediction Algorithms with R", Dr. Rafael A. Irizarry link

[2] "An Introduction to Statistical Learning with Applications in R", Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani link

[3] "LightGBM: A Highly Efficient Gradient Boosting Decision Tree", Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, Tie-Yan Liu link

[4] "Elastic Net Regularization", Hui Zou, Trevor Hastie link