- Pickl.AI

Used-Car Price Predtion

Developing a Machine Learning Model to predict the selling price of a used car

By Roshan Kumar

- Pickl.AI

Client Overview & Challenge

Our client is a well-established used car dealership renowned for its commitment to customer satisfaction and quality. Operating across diverse regions, they understand the importance of accurate pricing to ensure a positive customer experience and successful sales transactions. To this end, they have generously provided us with a comprehensive dataset containing information about the used cars they have sold.

Solution

- Pickl.AI

The automotive industry stands to revolutionize how used cars are priced and sold through predictive analytics. By harnessing the power of data and machine learning, we can unlock valuable insights that empower buyers, sellers, and dealerships to make informed decisions, resulting in a more efficient and transparent marketplace.

The Key Benefits it provides are Empowering Buyers and Sellers with a fair estimate of a vehicle's value based on a comprehensive analysis of historical data, market trends, and specific features. It can be used to determine otimized Pricing Strategies where Dealerships can strategically price their inventory by accurately anticipating future value, leading to faster inventory turnover and improved profitability.

AUsed car price prediction revolutionizes the automotive market by fostering transparency, efficiency, and informed decision-making. Through data-driven insights, buyers, sellers, and industry players can navigate the used car market with confidence, ultimately shaping the future of automotive transactions.

\

- Pickl.AI

Dataset Overview

Data quality is paramount for accurate predictions in this problem statement. High-quality data ensures that the classification model receives reliable and relevant information, leading to more accurate results. Poor data quality, such as missing values or erroneous entries, can introduce biases and errors that undermine the model's effectiveness.

	mon_year	KM_driven	Fuel_Type	Horse_Power	Color	Transmission	Engine	Doors	Cylinders	Gears	Sport_Model	selling_price
0	Oct_2006	49805.0	Diesel	90	Metallic	Manual	2000	3	4	5	0	14310.0
1	Oct_2006	77313.0	Diesel	90	Metallic	Manual	2000	3	4	5	0	14575.0
2	Sept_2006	44214.0	Diesel	90	Metallic	Manual	2000	3	4	5	0	14787.0
3	Jul_2006	50880.0	Diesel	90	Non-Metallic	Manual	2000	3	4	5	0	15847.0
4	Mar_2006	40810.0	Diesel	90	Non-Metallic	Manual	2000	3	4	5	0	14575.0

Data Fields and Overview

- Pickl.AI

Data Description

- `mon_year`: The month and year in which the car was first registered.
- `KM_driven`: The number of kilometers driven by the car.
- `Fuel_Type`: The type of fuel used by the car, either Diesel or Petrol.
- `Horse_Power`: The horsepower of the car's engine.
- `Color`: The color of the car, either Metallic or Non-Metallic.
- `Transmission`: The type of transmission used by the car, either Manual or Automatic.
- `Engine`: The size of the car's engine in cubic centimeters (cc).
- `Doors`: The number of doors the car has.
- `Cylinders`: The number of cylinders in the car's engine.
- `Gears`: The number of gears in the car's transmission.
- `Sport_Model`: A binary variable indicating whether the car is a sport model or not.
- `selling_price`: The price at which the car was sold.

DataSet Description

- Pickl.AI

	KM_driven	Horse_Power	Engine	Doors	Cylinders	Gears	Sport_Model	selling_price
count	1436.000000	1436.000000	1436.00000	1436.000000	1436.0	1436.000000	1436.000000	1436.000000
mean	72645.248607	101.502089	1576.85585	4.033426	4.0	5.026462	0.300139	11374.681755
std	39756.831763	14.981080	424.38677	0.952677	0.0	0.188510	0.458478	3844.583866
min	1.000000	69.000000	1300.00000	2.000000	4.0	3.000000	0.000000	4611.000000
25%	45580.000000	90.000000	1400.00000	3.000000	4.0	5.000000	0.000000	8957.000000
50%	67193.000000	110.000000	1600.00000	4.000000	4.0	5.000000	0.000000	10494.000000
75%	92242.000000	110.000000	1600.00000	5.000000	4.0	5.000000	1.000000	12667.000000
max	257580.000000	192.000000	16000.00000	5.000000	4.0	6.000000	1.000000	34450.000000

- Pickl.AI

Methodology

- Data Cleaning
- Data Preparation
- Feature Analysis
- Model Selection
- Model Validatuon
- Hyper-Parameter Tuning
- Model Performance

TRANSORG ANALYTICS (PICKL.AI)

Data Analysis

- Pickl.AI

Missing Data Analysis

```
Checking for Null Values:

mon_year         0
KM_driven        0
Fuel_Type        0
Horse_Power      0
Color            0
Transmission     0
Engine           0
Doors            0
Cylinders        0
Gears            0
Sport_Model      0
selling_price    0
dtype: int64
```

```
Unique values in each column:
mon_year : 77
KM_driven : 1263
Fuel_Type : 3
Horse_Power : 12
Color : 2
Transmission : 2
Engine : 13
Doors : 4
Cylinders : 1
Gears : 4
Sport_Model : 2
selling_price : 236
```

- Our data contained **1,436 entries**
- **No** records with **missing values**.
- All the records were unique(No duplicates were present)

TRANSORG ANALYTICS (PICKL.AI)

Data Cleaning

Data Cleaning

In the context of used-cars price prediction, accurate and clean data is essential for training robust prediction models. If the data contains errors or inconsistencies, the model may learn from flawed patterns, leading to poor performance and unreliable predictions.

Data cleaning improves the overall quality of the dataset by removing duplicate records, correcting errors, and addressing missing values. High-quality data supports more accurate analysis and produces better results.

Data Cleaning

After Checking the Dataset:

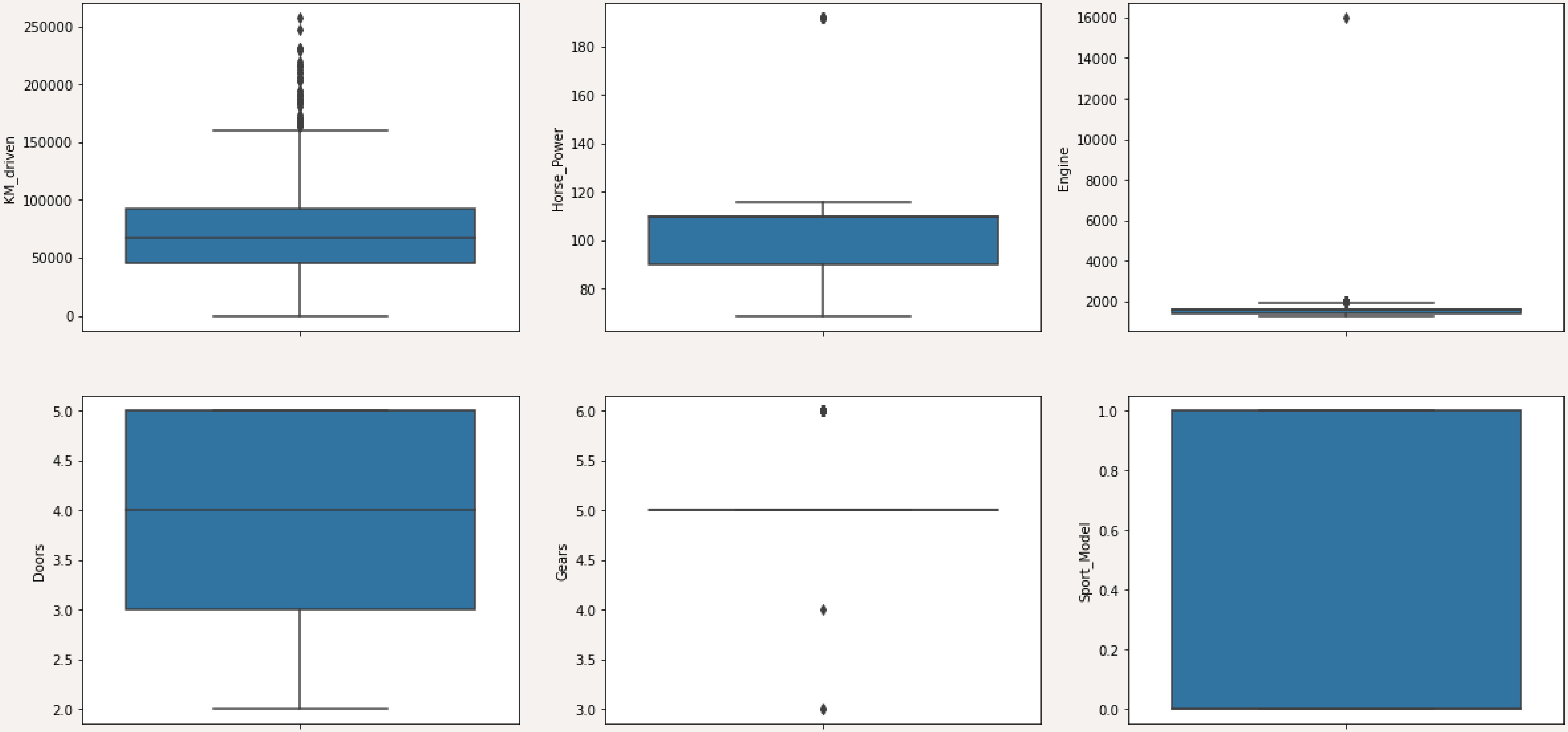
- 'Cylinder' was a redundant column
- Split the column 'mon_year' into 'month' and 'year'

	month	year	KM_driven	Fuel_Type	Horse_Power	Color	Transmission	Engine	Doors	Gears	Sport_Model	selling_price
0	Oct	2006	49805.0	Diesel	90	Metallic	Manual	2000	3	5	0	14310.0
1	Oct	2006	77313.0	Diesel	90	Metallic	Manual	2000	3	5	0	14575.0
2	Sept	2006	44214.0	Diesel	90	Metallic	Manual	2000	3	5	0	14787.0
3	Jul	2006	50880.0	Diesel	90	Non-Metallic	Manual	2000	3	5	0	15847.0
4	Mar	2006	40810.0	Diesel	90	Non-Metallic	Manual	2000	3	5	0	14575.0

TRANSORG ANALYTICS (PICKL.AI)

Data Visualization

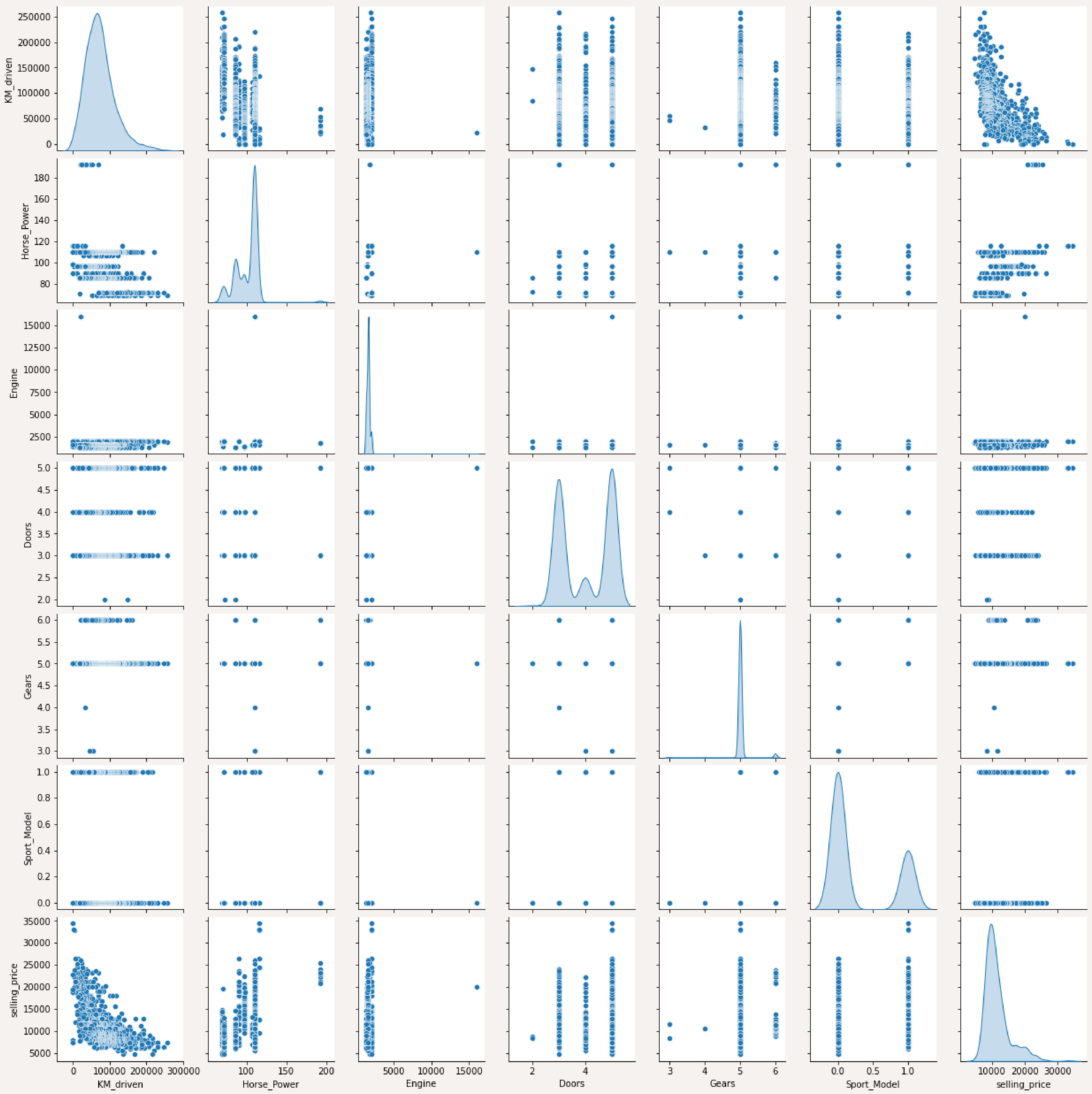
Box Plot Analysis

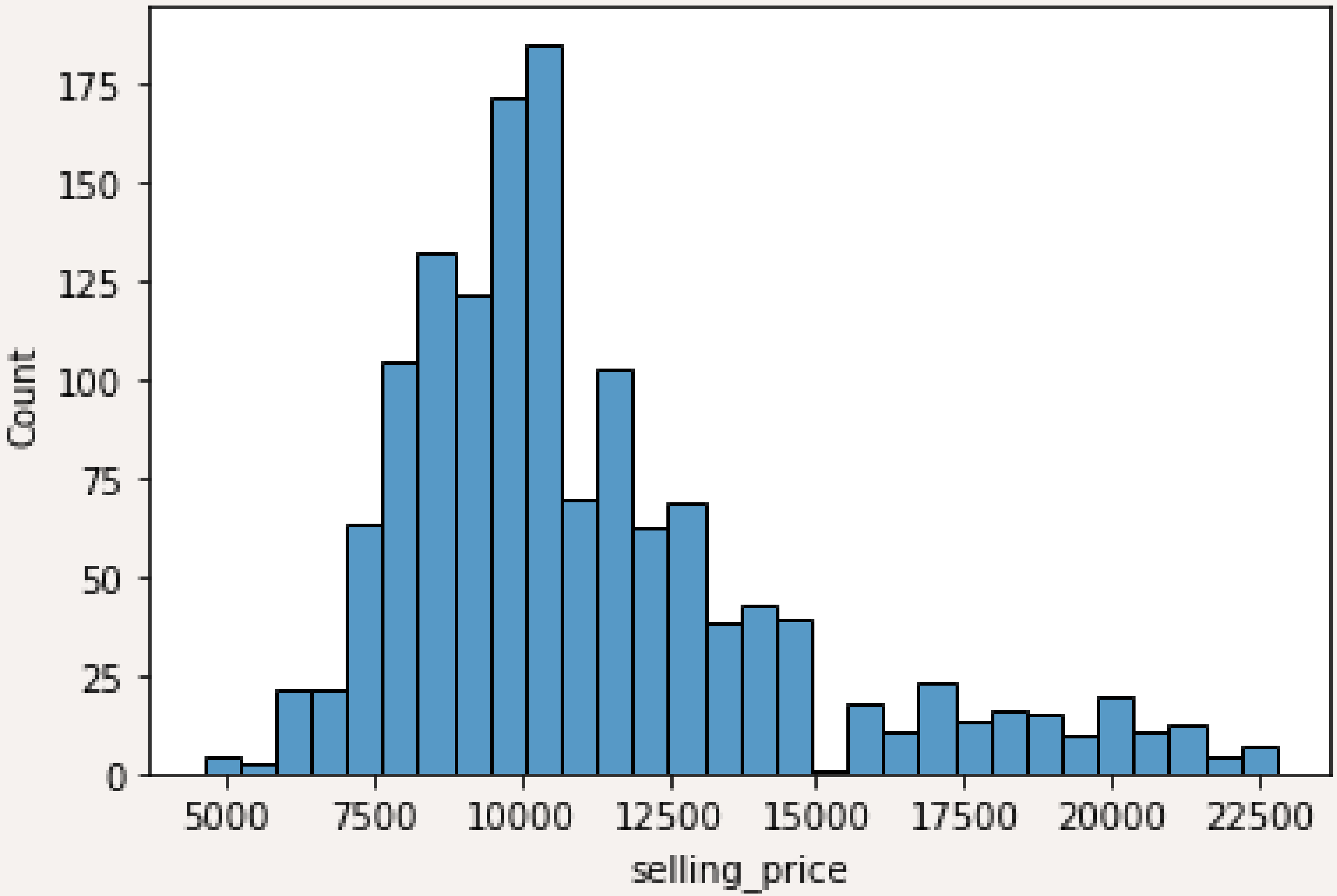



Features vs Class
KM_driven distribution
Horse_Power distribution
Engine distribution
Doors distribution
Gears distribution
selling_price distribution
0 10 20 30 40 50 60 70 80
0 50000 100000 150000 200000 250000
0 100 200 300 400 500 600 700 800
80 100 120 140 160 180
0 100 200 300 400 500 600 700 800
2500 5000 7500 10000 12500 15000
0 100 200 300 400 500 600 700
2.0 2.5 3.0 3.5 4.0 4.5 5.0
0 200 400 600 800 1000 1200 1400
3.0 3.5 4.0 4.5 5.0 5.5 6.0
0 25 50 75 100 125 150 175
5000 10000 15000 20000 25000 30000 35000

Pair Plot Visualization

In a pair plot, each variable in the dataset is compared with every other variable, resulting in a grid of scatterplots. The diagonal of the grid usually contains histograms or density plots for each individual variable, showing their distributions. The off-diagonal plots are scatterplots that illustrate how pairs of variables interact.



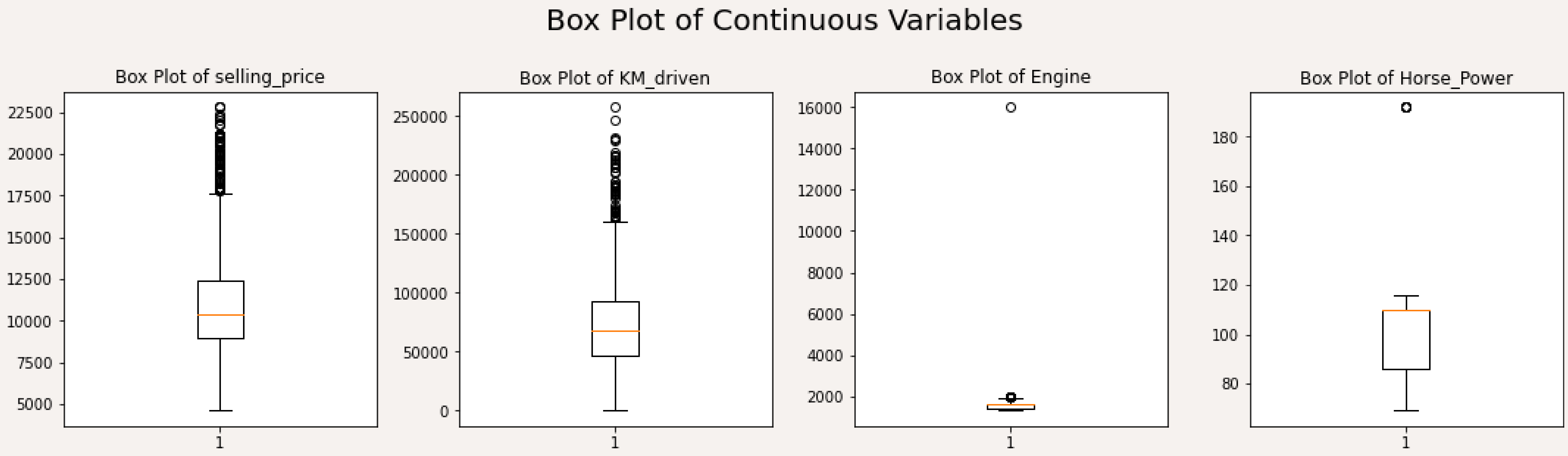


Count
selling_price
0
25
50
75
100
125
150
175
5000
7500
10000
12500
15000
17500
20000
22500

TRANSORG ANALYTICS (PICKL.AI)

Outlier Analysis

Outlier Analysis



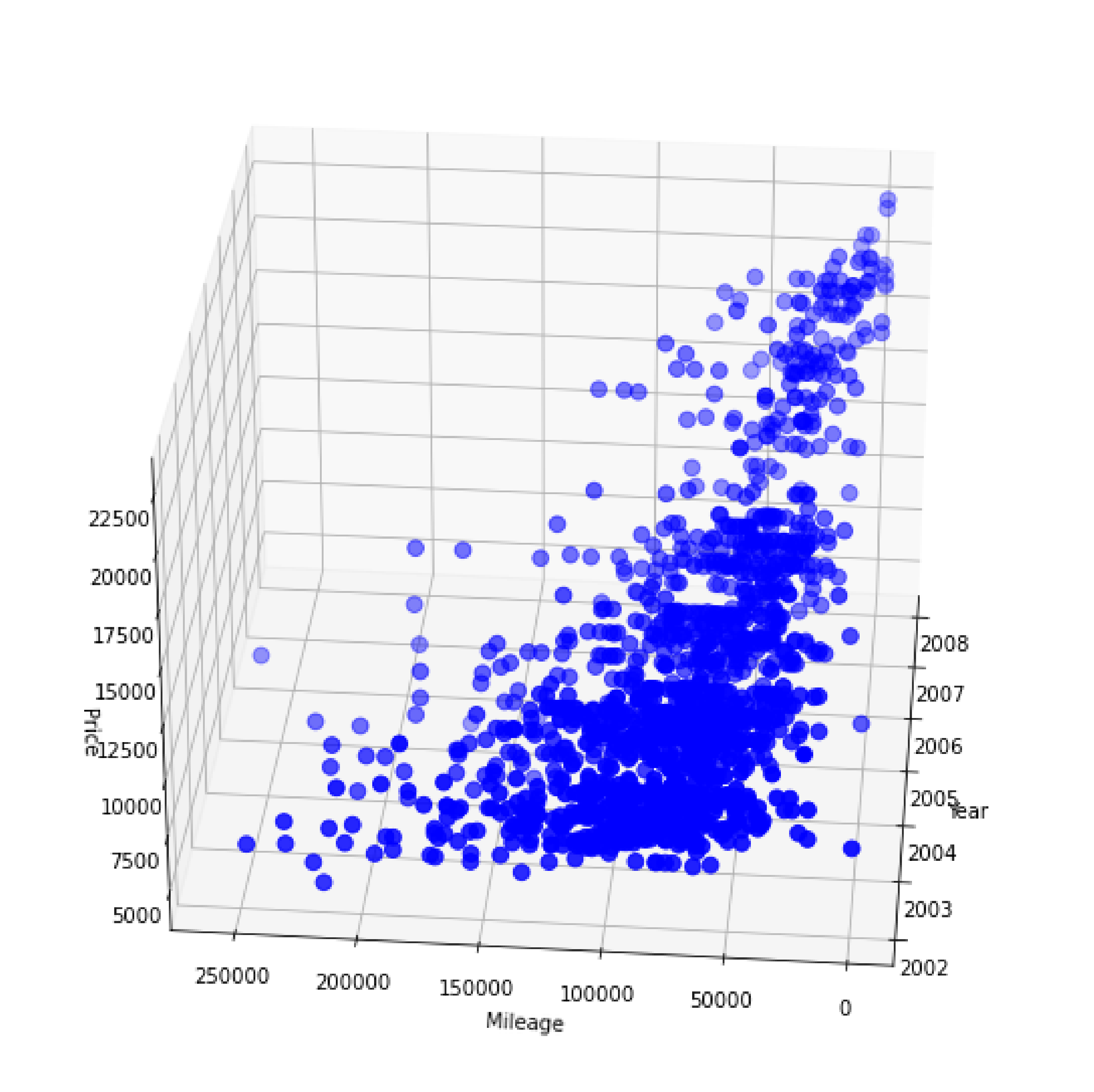
- Number of Outliers in selling_price and 'KM_driven is more.
- 'Engine' and 'Horse_Power have less outliers.

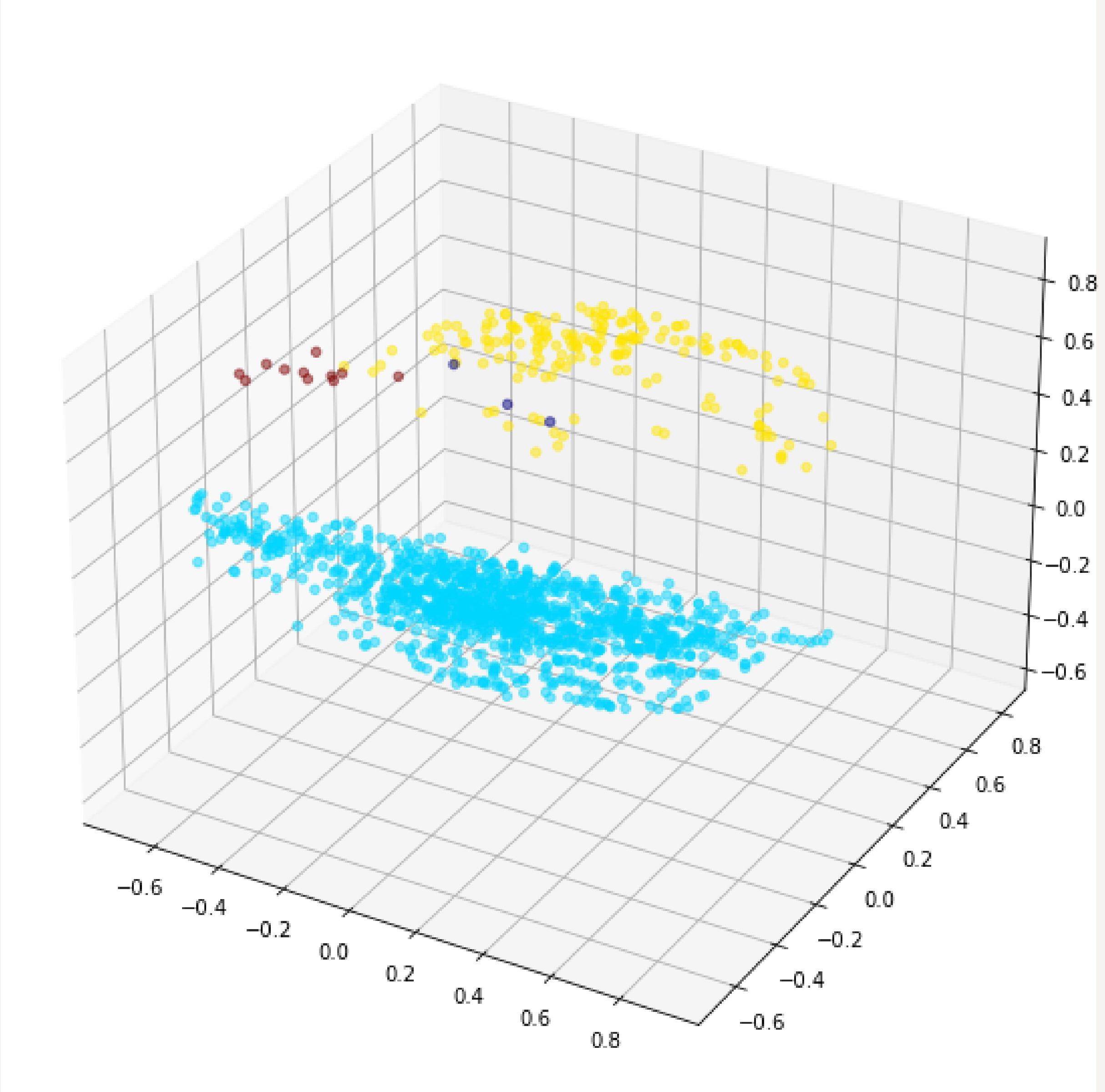
Z-Score Analysis

```
selling_price
The score threshold is: 3
The indices of the outliers:
(array([ 14,  15,  16,  49,  53,  68,  89,  91, 109, 110, 111, 112, 113,
        114, 115, 116, 119, 125, 138, 141, 147, 154, 171, 174, 178, 179],
       dtype=int64),)
Number of outliers is: 26
--------------------------------------------------------------------------------
KM_driven
The score threshold is: 3
The indices of the outliers:
(array([ 186,  378,  379,  603,  604,  605,  606,  607, 1044, 1045, 1046,
        1047, 1048, 1049, 1050, 1051, 1052, 1053], dtype=int64),)
Number of outliers is: 18
--------------------------------------------------------------------------------
Engine
The score threshold is: 3
The indices of the outliers:
(array([80], dtype=int64),)
Number of outliers is: 1
--------------------------------------------------------------------------------
Horse_Power
The score threshold is: 3
The indices of the outliers:
(array([  8,  10,  11,  12,  13,  14,  15,  16,  49,  53, 141], dtype=int64),)
Number of outliers is: 11
--------------------------------------------------------------------------------
```

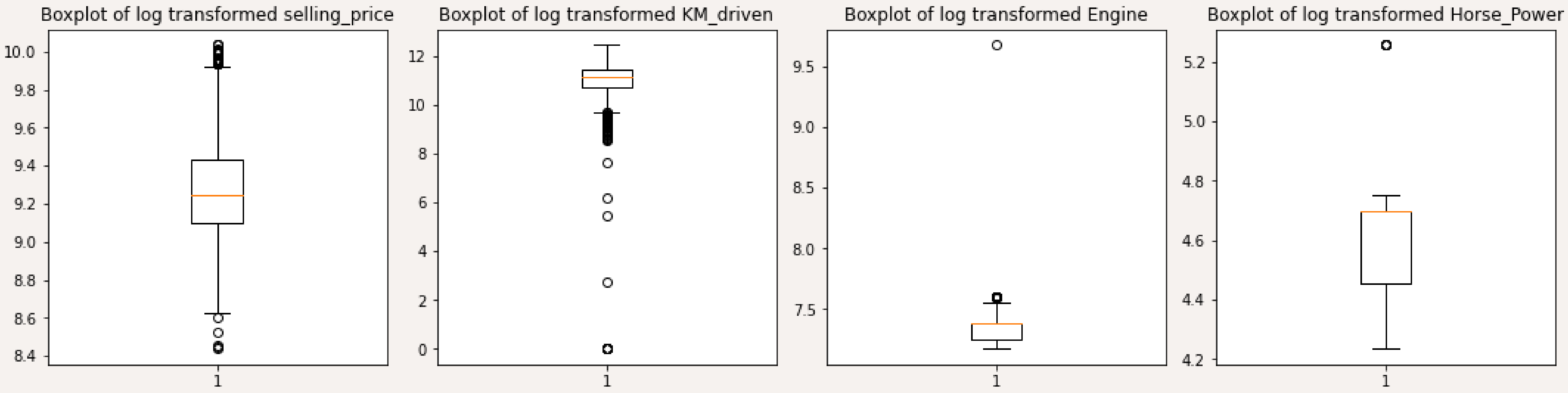
```
selling_price
The score threshold is: 2
Number of outliers is: 62
--------------------------------------------------------------------------------
KM_driven
The score threshold is: 2
Number of outliers is: 27
--------------------------------------------------------------------------------
Engine
The score threshold is: 2
Number of outliers is: 1
--------------------------------------------------------------------------------
Horse_Power
The score threshold is: 2
Number of outliers is: 11
--------------------------------------------------------------------------------
```

DBSCAN


22500
20000
17500
15000
12500
10000
7500
5000
Price
2008
2007
2006
2005
2004
2003
2002
Year
250000
200000
150000
100000
50000
0
Mileage

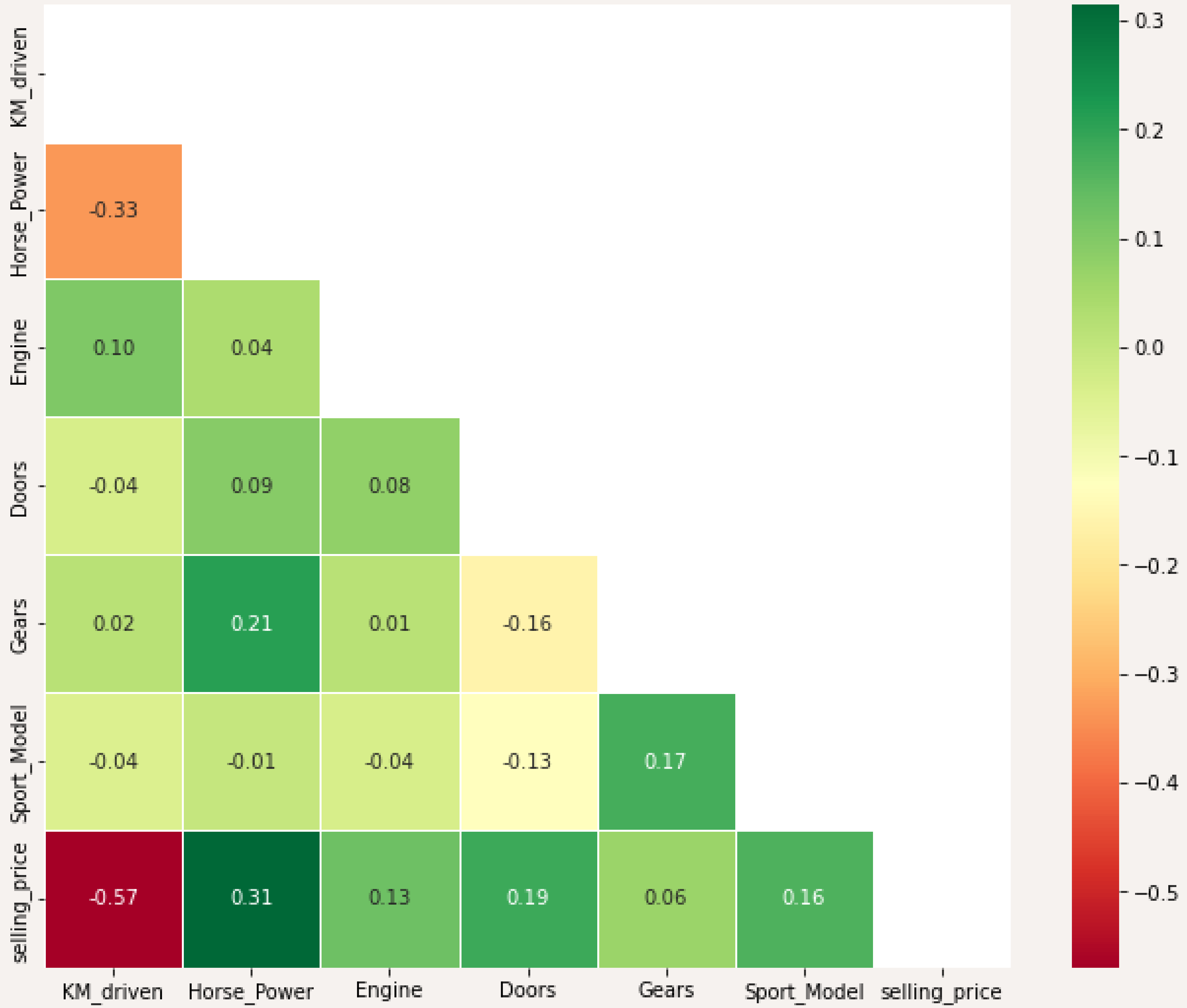

0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
−0.6
−0.4
−0.2
0.0
0.2
0.4
0.6
0.8

Boxplot of log transformed data



Co-relation Matrix

- A correlation matrix is a tabular representation of the correlation coefficients between multiple variables in a dataset.
- Correlation coefficients quantify the strength and direction of the linear relationship between pairs of variables.

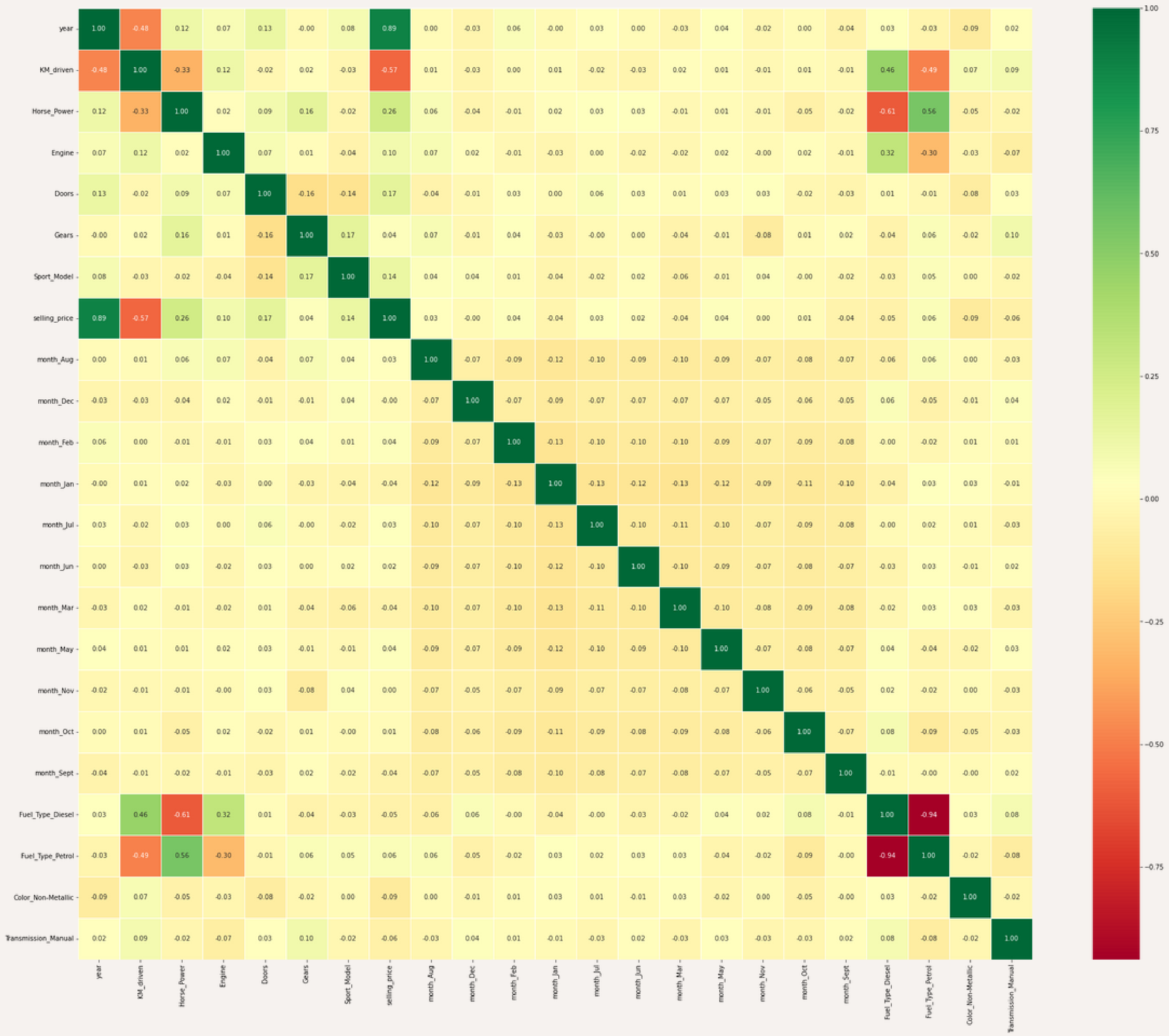


Encoding the Data

Encoding Categorical Variables

- Categorical Variables: 'Month', 'Fuel_Type' , 'Color' and 'Transmission'.

A Machine Learning model understands only the numerical data hence coulumn of categorical data should be converted into numerical using the technique of `One-Hot-Encoding`.



TRANSORG ANALYTICS (PICKL.AI)

Model Building

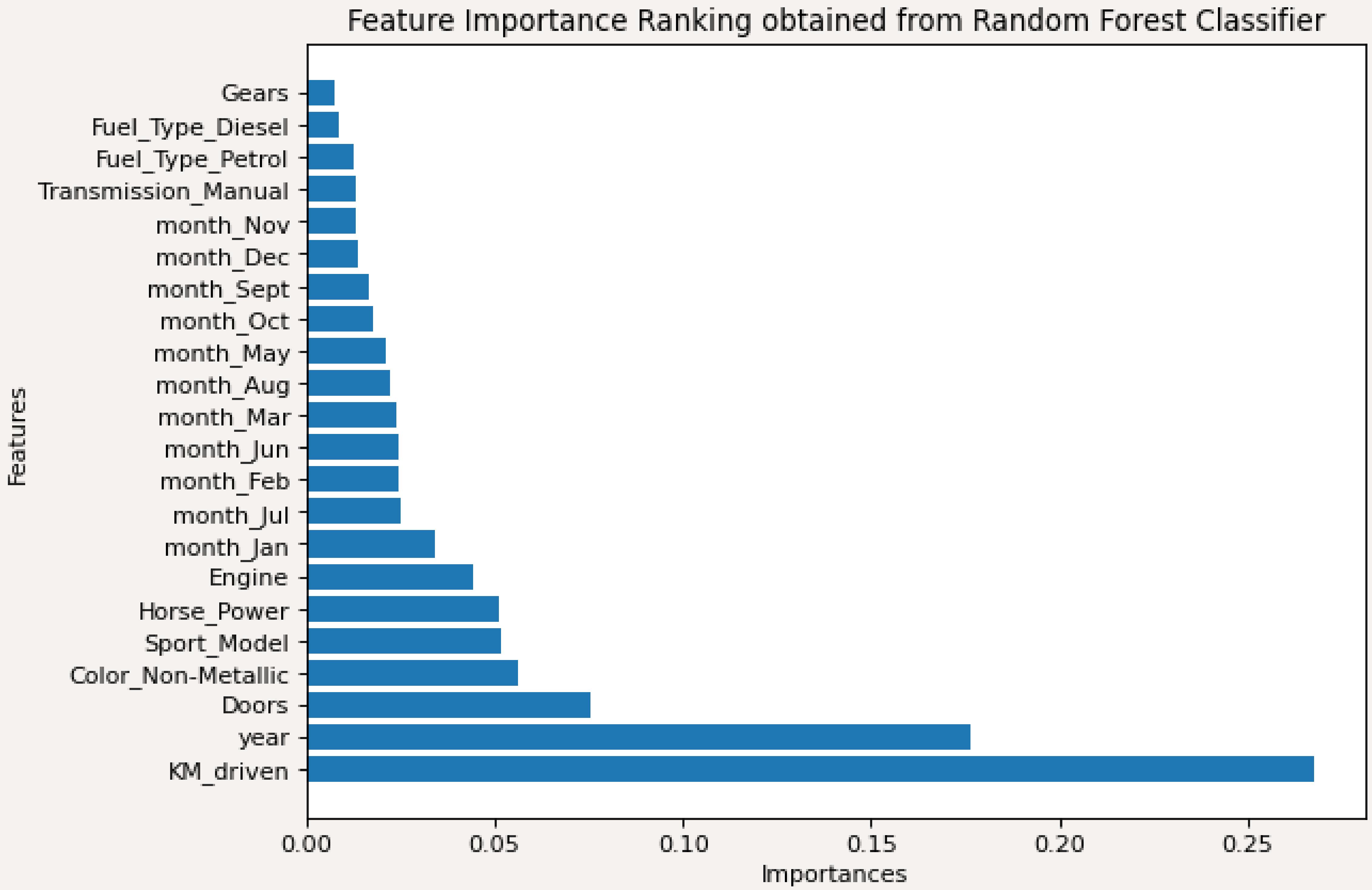
Comparing Performance of Different Models

```
Linear Regression
MAE: 1083.5171256218794, R2: 0.8580328256205721, RMSE: 1458.894233275079

Logistic Regression
MAE: 2642.6111111111113, R2: -0.15663307692252348, RMSE: 4164.162245678394

Decision Tree
MAE: 1141.0590277777778, R2: 0.8401366735787478, RMSE: 1548.118716064042

Random Forest
MAE: 861.806076388889, R2: 0.9198007073569624, RMSE: 1096.516338646036

Gradient Boost
MAE: 852.3116804214494, R2: 0.924166010689108, RMSE: 1066.2567410413183

XGBoost
MAE: 900.2789849175347, R2: 0.910456031352357, RMSE: 1158.6386394635279
```

Performance on Validation Dataset

```
Linear Regression
MAE: 1098.6463919212638, R2: 0.8156882233107704, RMSE: 1497.9579811929802

Logistic Regression
MAE: 2261.340425531915, R2: 0.0984574547068735, RMSE: 3312.9600721584784

Decision Tree
MAE: 1294.2021276595744, R2: 0.7540660413290835, RMSE: 1730.343551137906

Random Forest
MAE: 1034.9301394799054, R2: 0.8312239557038477, RMSE: 1433.436604192441

Gradient Boost
MAE: 978.3386471158499, R2: 0.8446310896454284, RMSE: 1375.324401410957

XGBoost
MAE: 1059.0373864140072, R2: 0.8270315717702572, RMSE: 1451.130626052124
```

Feature Selection



Tuning in different paramerters

```
Linear Regression
MAE: 1010.6762297140081, R2: 0.8100667240836125, RMSE: 1293.6863557547806

Logistic Regression
MAE: 1943.5851063829787, R2: 0.13664095214024763, RMSE: 2758.190070415563

Decision Tree
MAE: 1061.973404255319, R2: 0.7663332166831205, RMSE: 1434.917586347707

Random Forest
MAE: 868.661763086795, R2: 0.8461574916036146, RMSE: 1164.3046837342229

Gradient Boost
MAE: 793.3371644571711, R2: 0.8807852328846479, RMSE: 1024.9283254256925

XGBoost
MAE: 934.1140171348626, R2: 0.8338037611621505, RMSE: 1210.1496063390196
```

Model Fine-Tuning

```
Linear Regression
Best Params: {}
MAE: 1010.6762297140081, R2: 0.8100667240836125, RMSE: 1293.6863557547806

Logistic Regression
Best Params: {}
MAE: 1943.5851063829787, R2: 0.13664095214024763, RMSE: 2758.190070415563

Decision Tree
Best Params: {'max_depth': 5, 'min_samples_leaf': 4, 'min_samples_split': 5}
MAE: 893.9189497745165, R2: 0.8412032588382623, RMSE: 1182.9033482745797

Random Forest
Best Params: {'max_depth': 10, 'max_features': 'auto', 'min_samples_leaf': 2, 'min_samples_split': 10, 'n_estimators': 100}
MAE: 790.123517942525, R2: 0.871620873326082, RMSE: 1063.593505535338

Gradient Boost
Best Params: {'learning_rate': 0.05, 'max_depth': 3, 'min_samples_leaf': 2, 'min_samples_split': 10, 'n_estimators': 200}
MAE: 791.8409393831059, R2: 0.8811197371256095, RMSE: 1023.4893941875746

XGBoost
Best Params: {'learning_rate': 0.1, 'max_depth': 3, 'min_child_weight': 3, 'n_estimators': 100}
MAE: 803.0501492547651, R2: 0.8764646185065403, RMSE: 1043.33589388734
```