Predicting Car Prices Using a Simple Linear Regression Model with Basic Data Analysis

Group 3: Do Quoc Duy, Nguyen Thai Duy, Hoang Quoc Duy, Tran Anh Duong

December 26, 2024

Abstract

Understanding the factors influencing second-hand car prices is critical in the used car market, where the absence of standardized valuation frameworks often leads to reliance on subjective judgments. This study addresses this gap by developing a Linear Regression model to predict car prices based on key features, including car specifications, brand, and service life. The methodology involves thorough data analysis to assess feature distributions and residual patterns, coupled with targeted data transformations to resolve inconsistencies and enhance model accuracy. Our analysis reveals that the dataset does not satisfy the assumptions of linear regression such as homoscedasticity and normality. Comparative evaluations reveal significant performance improvements post-processing, demonstrating the effectiveness of these techniques. This work contributes a data-driven approach to price estimation, offering practical implications for buyers, sellers, and market analysts.

1 Introduction

The pricing of second-hand cars is influenced by factors such as service life, engine performance, features, and brand reputation. Vehicles with lower mileage, superior specifications, and strong brand recognition typically command higher prices. These attributes can be quantified through numerical and categorical data, including metrics like mileage, age, onboard features, and brand classification.This study addresses the lack of standardized valuation in the used car market by developing a predictive model. Initial evaluations are conducted on raw data, followed by iterative refinements through data preprocessing to improve model accuracy.

2 The dataset

The dataset utilized in this study was sourced from Kaggle and comprises 14,242 instances across 23 columns. The target variable, `price`, is predicted using

features pertaining to car models, brands, and specifications. A summary of the dataset's key attributes is provided to outline the essential information relevant to the analysis.

	price	km	Gears	age	Previous_Owners	hp_kW	Inspection_new	Displacement_cc	Weight_kg	cons_comb
count	15915.000000	15915.000000	15915.000000	15915.000000	15915.000000	15915.000000	15915.000000	15915.000000	15915.000000	15915.000000
mean	18024.380584	32089.995708	5.937355	1.389695	1.042853	88.499340	0.247063	1428.661891	1337.700534	4.832124
std	7381.679318	36977.214964	0.704772	1.121306	0.339178	26.674341	0.431317	275.804272	199.682385	0.867530
min	4950.000000	0.000000	5.000000	0.000000	0.000000	40.000000	0.000000	890.000000	840.000000	3.000000
25%	12850.000000	1920.500000	5.000000	0.000000	1.000000	66.000000	0.000000	1229.000000	1165.000000	4.100000
50%	16900.000000	20413.000000	6.000000	1.000000	1.000000	85.000000	0.000000	1461.000000	1295.000000	4.800000
75%	21900.000000	46900.000000	6.000000	2.000000	1.000000	103.000000	0.000000	1598.000000	1472.000000	5.400000
max	74600.000000	317000.000000	8.000000	3.000000	4.000000	294.000000	1.000000	2967.000000	2471.000000	9.100000

Figure 1: Summary of the Dataset (Describe Method)

	make_model	body_type	price	vat	km	type	Fuel	Gears	Comfort_Convenience	Entertainment_Media	...	Previous_Owners	hp_kW	Inspection_new	Paint_Type	Upholstery_type	Gearing_Type	Displacement_cc	Weight_kg	Drive_chain	cons_comb
0	Audi A1	Sedans	15770	VAT deductible	56013.000000	Used	Diesel	7.0	Air conditioning,Armrest,Automatic climate con...	Bluetooth,Hands-free equipment,On-board comput...	...	2.0	66.0	1	Metallic	Cloth	Automatic	1422.0	1220.0	front	3.8
1	Audi A1	Sedans	14500	Price negotiable	80000.000000	Used	Benzine	7.0	Air conditioning,Automatic climate control,Hil...	Bluetooth,Hands-free equipment,On-board comput...	...	1.0	141.0	0	Metallic	Cloth	Automatic	1798.0	1255.0	front	5.6
2	Audi A1	Sedans	14640	VAT deductible	83450.000000	Used	Diesel	7.0	Air conditioning,Cruise control,Electrical sid...	MP3,On-board computer	...	1.0	85.0	0	Metallic	Cloth	Automatic	1598.0	1135.0	front	3.8
3	Audi A1	Sedans	14500	VAT deductible	73000.000000	Used	Diesel	6.0	Air suspension,Armrest,Auxiliary heating,Elect...	Bluetooth,CD player,Hands-free equipment,MP3,O...	...	1.0	66.0	0	Metallic	Cloth	Automatic	1422.0	1195.0	front	3.8
4	Audi A1	Sedans	16790	VAT deductible	16200.000000	Used	Diesel	7.0	Air conditioning,Armrest,Automatic climate con...	Bluetooth,CD player,Hands-free equipment,MP3,O...	...	1.0	66.0	1	Metallic	Cloth	Automatic	1422.0	1135.0	front	4.1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
16907	Renault Espace	Van	39980	VAT deductible	100.000000	Pre-registered	Diesel	6.0	Air conditioning,Armrest,Automatic climate con...	Bluetooth,Digital radio,Hands-free equipment,O...	...	1.0	118.0	0	Metallic	Part/Full Leather	Automatic	1598.0	1734.0	front	4.7
16908	Renault Espace	Van	39950	VAT deductible	1647.362609	New	Diesel	6.0	Air conditioning,Automatic climate control,Cru...	Bluetooth,Digital radio,Hands-free equipment,O...	...	1.0	147.0	0	Metallic	Part/Full Leather	Automatic	1997.0	1758.0	front	5.3
16909	Renault Espace	Van	39950	VAT deductible	1000.000000	Demonstration	Benzine	6.0	Air conditioning,Armrest,Automatic climate con...	Bluetooth,Digital radio,Hands-free equipment,M...	...	1.0	165.0	0	Metallic	Part/Full Leather	Automatic	1798.0	1734.0	front	6.8
16911	Renault Espace	Van	39885	VAT deductible	9900.000000	Used	Benzine	7.0	Air conditioning,Automatic climate control,Cru...	Bluetooth,Digital radio,Hands-free equipment,R...	...	1.0	165.0	0	Metallic	Cloth	Automatic	1798.0	1708.0	front	7.4
16912	Renault Espace	Van	39875	VAT deductible	15.000000	Pre-registered	Diesel	6.0	Air conditioning,Armrest,Automatic climate con...	Bluetooth,Hands-free equipment,On-board comput...	...	1.0	146.0	1	Metallic	Part/Full Leather	Automatic	1997.0	1734.0	front	5.3

Figure 2: First Few Entries of the Dataset

We look at the information given and see that the statistic for standard deviation, value range swings wildly. This suggests that the dataset may contain many problems and will need further exploration.

3 Simple EDA

To begin, we need to examine whether the features are linear with the target and if there is any potential multicollinearity. The heatmap and pair plot below illustrate the degree of linearity among the variables.

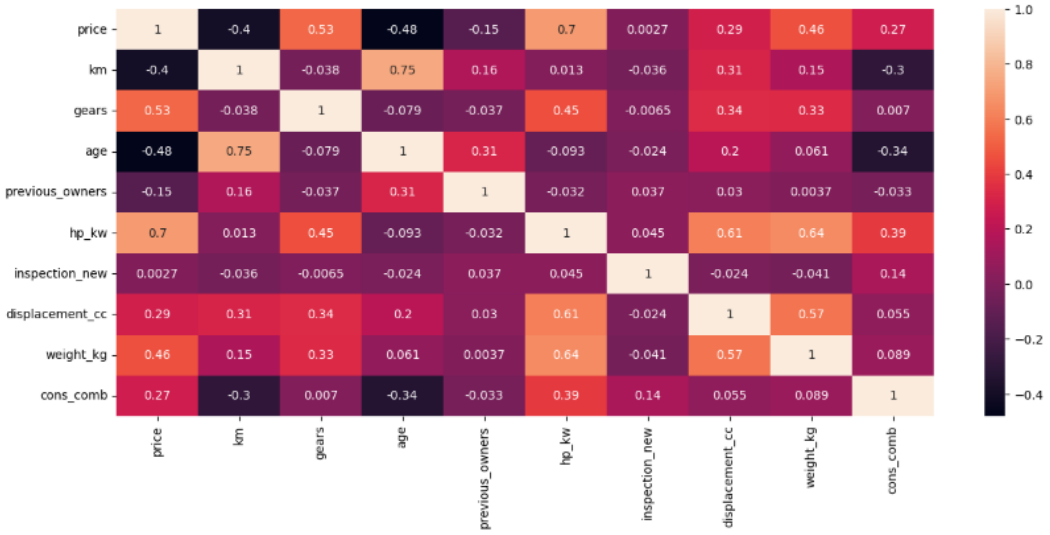


Figure 3: Heatmap showing the correlation between columns.

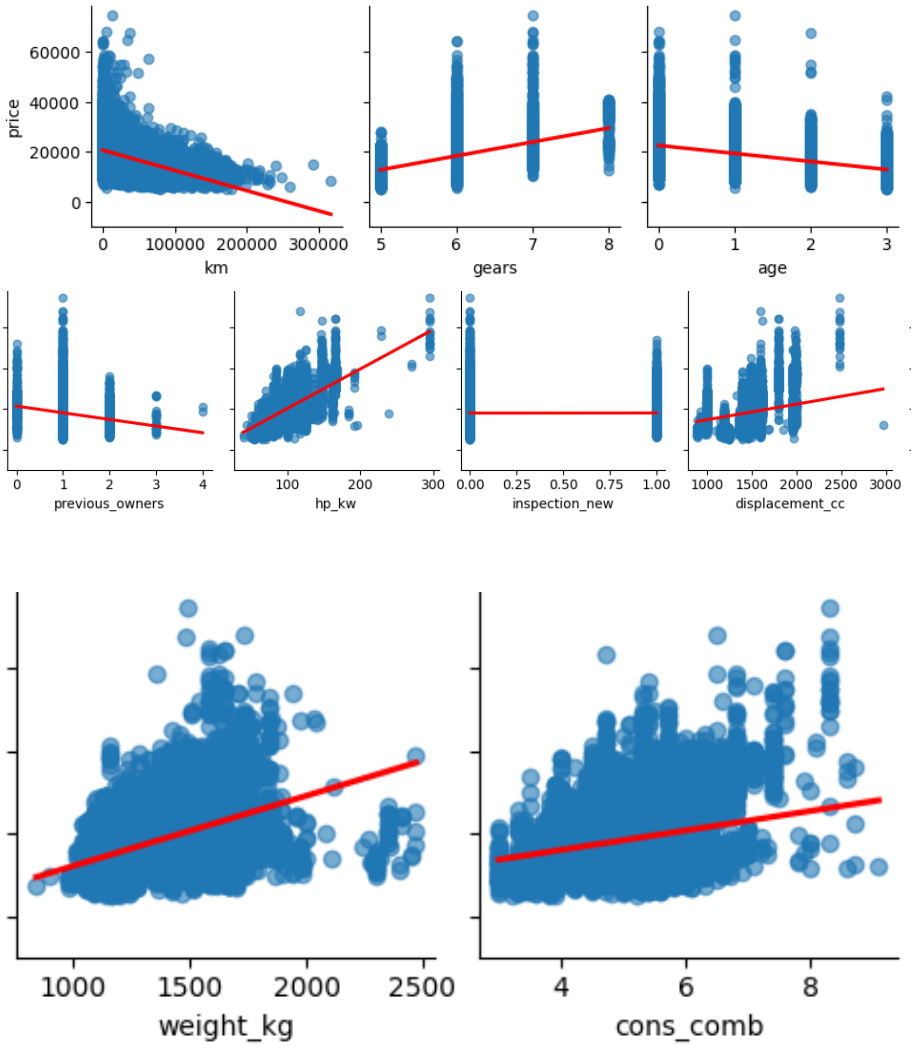
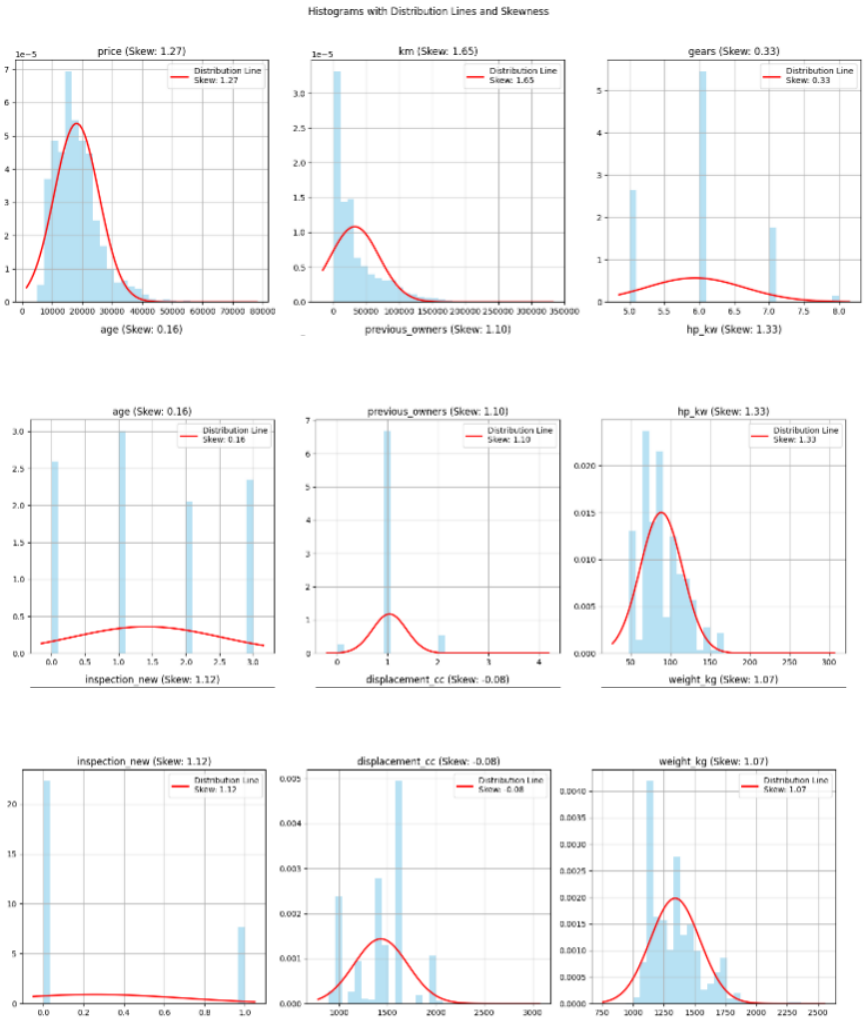


Figure 4: Pair plot illustrating the relationships between columns.

The figures reveal strong linear correlations between the target variable and most feature columns, except for the `inspection_new` column. This column

contributes minimal information to the target variable and may introduce noise; thus, it will be excluded from further analysis. Additionally, the plots indicate the presence of potential outliers in the dataset.

The heatmap also highlights instances of multicollinearity among the features, which will be addressed in subsequent steps of the analysis. Since Linear Regression assumes a normal distribution for optimal performance, it is also necessary to assess the degree of skewness in the numerical data to see whether it is worth while to apply box-cox transformation.



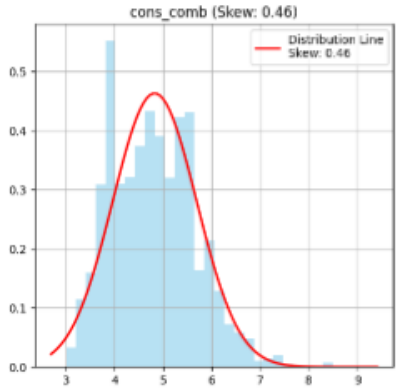


Figure 5: Histogram showing the distribution of each column

The histogram shows that some of the columns are highly skewed, which could decrease the performance of our model. We will address this by applying box-cox transformation.

4 Linear Regression analysis - First attempt

We will build a Multiple Linear Regression model to predict car prices. The model's performance will be evaluated using R-squared to assess its predictive power. In the first attempt, we will use only the original numerical data. We speculate that the model's accuracy will be relatively low.

The model can be described in matrix notation as $y = X\beta + \varepsilon$, Y has dimension 14242×1, X has dimension 14242×9, β has dimension 9×1, and ε is a scalar. Using scikit-learn, we calculate $\hat{\beta}$ and ε :

$$\begin{bmatrix} -1.66095777 \times 10^3 \\ 1.63535178 \times 10^3 \\ -2.01905036 \times 10^3 \\ 2.54851648 \\ 4.55408941 \times 10^3 \\ -1.40852243 \times 10^2 \\ -6.90776262 \times 10^2 \\ 7.81721363 \times 10^2 \\ -9.53560757 \times 10^2 \end{bmatrix}$$

$$\begin{bmatrix} 18077.021568933647 \end{bmatrix}$$

With an R-squared score of only 0.744961871699976, this aligns with our initial assumption. As shown in the plot below, the line fits the data to some extent but lacks high accuracy.

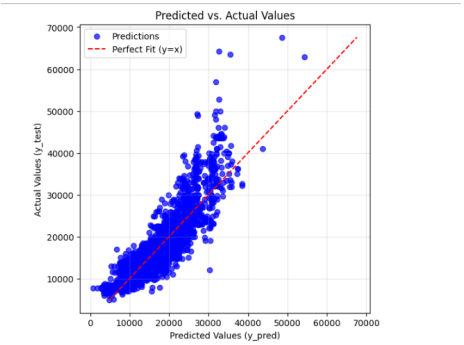


Figure 6: Plot of the ideal fit line

We can write our model as:

$$y = 18077.02 + 1660.96x_1 + 1635.35x_2 + 2019.05x_3 - 2.55x_4 + 4554.09x_5 - 140.85x_6 + 690.78x_7 + 781$$

In a linear model, there are multiple assumptions. However, we have not yet checked whether all the assumptions of our model hold. To address this, we will generate diagnostic plots to identify any potential violations of these assumptions.

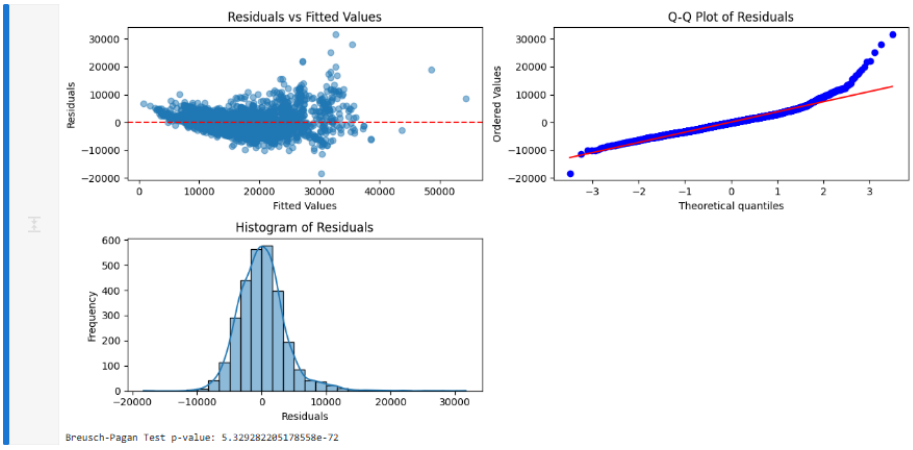


Figure 7: Visualization of normal, homogeneity, and homoscedasticity assumptions

The normal assumption and homoscedasticity assumption of our case is therefore slightly violated. We also calculate the condition number of our dataset (156185.95222972264) showing that there are multicollinearity in our data. Our data transformation process will aim to fix these in our data.

5 Data transformation process

In order to improve the performance of the model, several pre-processing steps will be applied to the data. This includes handling outliers, normalizing the data, ensuring normal distribution and addressing multicollinearity.

5.1 Ensuring Normal Distribution with Box-Cox Transformation

One of the assumptions of Linear Regression is the normalities of residual errors. The residual plot shows clear heteroscedasticity, indicating that a transformation is necessary. One way that may improve this is making our feature distribution normal, because our sample is not large enough, we want to ensure the central limit theorem. The Box-Cox transformation is particularly useful here, as it can help stabilize the variance and normalize the data. Our results are shown below.

For a positive response variable y, the Box-Cox transformation is defined as:

$$y_{(\lambda)} = \begin{cases} \frac{y^{\lambda}-1}{\lambda} & \text{if } \lambda \neq 0 \\ \ln(y) & \text{if } \lambda = 0 \end{cases}$$

We applied this to all of the numerical values except for the *inspection_new* column as we have dropped it previously. The distribution of all columns has improved significantly, as shown below.

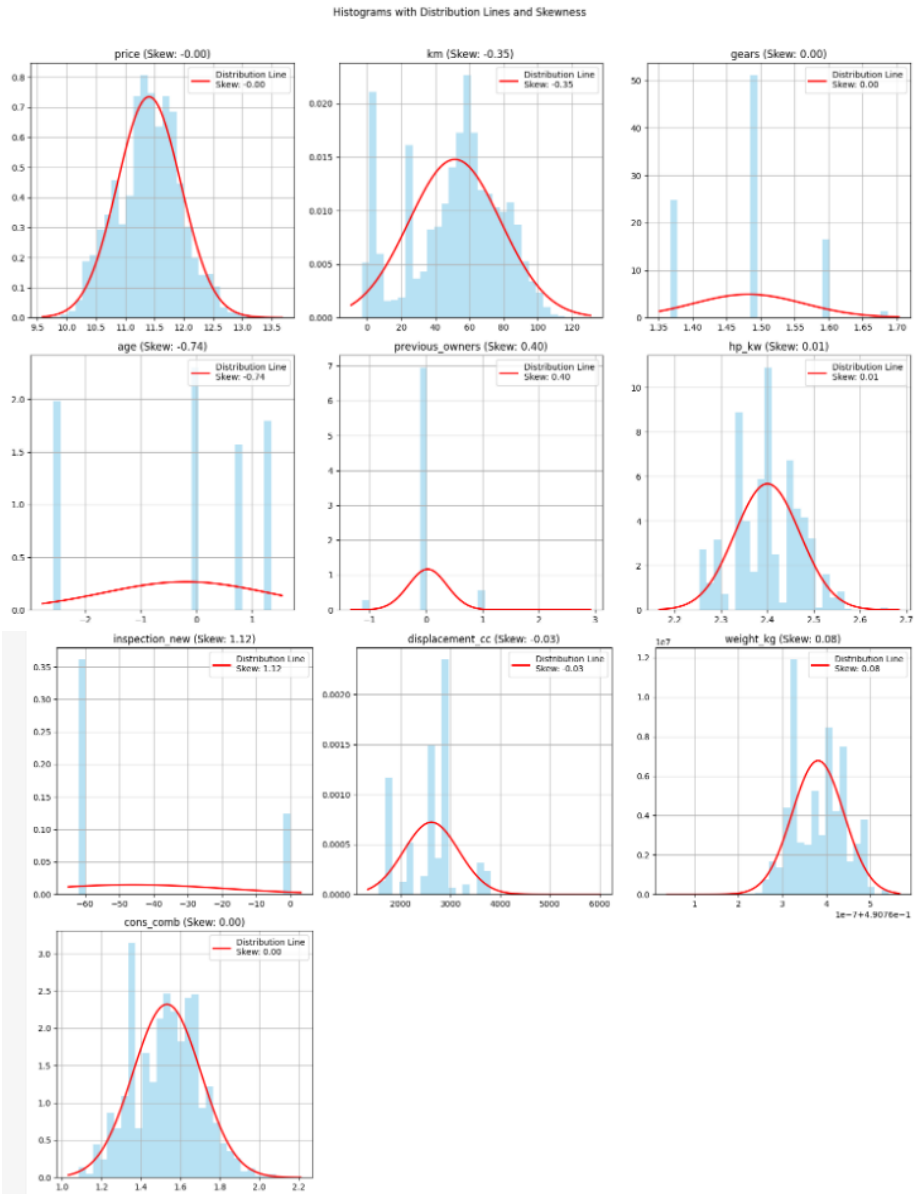


Figure 8: Distribution of data after box-cox transformation

5.2 Handling Outliers Using Robust Scaler

To address the outliers visible in the residual plot, we implement the robust scaling. This method is less sensitive to outliers than standard scaling techniques.

The Robust Scaling is:

$$X_{\text{scaled}} = \frac{X - \text{median}(X)}{\text{IQR}(X)}$$

where $\text{IQR}(X)$ represents the interquartile range (75th percentile - 25th percentile). We will apply this to all of the X variables in our data-set. Using scikit-learn, we arrive at:

	km	gears	age	previous_owners	hp_kw	inspection_new	displacement_cc	weight_kg	cons_comb
0	0.504012	0.816285	0.399746	0.955899	-0.585414	1.0	-0.105914	-0.344557	-0.850009
1	0.724738	0.816285	0.233813	0.000000	1.041085	0.0	0.925798	-0.214442	0.558541
2	0.752391	0.816285	0.399746	0.000000	0.000000	0.0	0.374150	-0.712877	-0.850009
3	0.665889	0.000000	0.399746	0.000000	-0.585414	0.0	-0.105914	-0.444671	-0.850009
4	-0.111489	0.816285	0.399746	0.000000	-0.585414	1.0	-0.105914	-0.712877	-0.573065
...	...	...	...	...	...	...	...	...	...
14237	-1.226664	0.000000	-0.766187	0.000000	0.693373	0.0	0.374150	0.843782	-0.076428
14238	-0.807970	0.000000	-0.766187	0.000000	1.119703	0.0	1.480523	0.875130	0.359253
14239	-0.908772	0.000000	-0.766187	0.000000	1.332355	0.0	0.925798	0.843782	1.259400
14240	-0.301783	0.816285	-0.766187	0.000000	1.332355	0.0	0.925798	0.808300	1.563719
14241	-1.370270	0.000000	-0.766187	0.000000	1.106895	1.0	1.480523	0.843782	0.359253

14242 rows × 9 columns

Figure 9: Enter Caption

We are sure that all of the outliers are being dealt with as Robust Scaler will shrink the data based on their IQR.

5.3 Categorical Variable Encoding

For categorical variables, we implement one-hot encoding. For a categorical variable with k categories, the encoding function is:

$$\text{OneHot}_i(x) = \begin{cases} 1 & \text{if } x \text{ belongs to category } i \\ 0 & \text{otherwise} \end{cases}$$

for $i \in \{1, \ldots, k\}$ The categorical data includes: "make_model", "body_type", "type", "fuel",**"comfort_convenience"**, **"entertainment_media"**, "extras", "safety_security", "paint_type", "upholstery_type", "gearing_type", "drive_chain" . We suspect that the number of dimensions will increase manifold as **"comfort_convenience"**, **"entertainment_media"** have various unique values. This may be a potential drawback when applying Linear Regression with too many dimensions. We will deal with this through dimensionality reduction.

5.4 Dimensionality Reduction with PCA

Principal Component Analysis (PCA) can help reduce multicollinearity and improve model efficiency. We decide to reduce the dimensions of the data as we have not addressed the problem of multicollinearity and too many dimensions. The transformation is given by:

$$X_{\text{PCA}} = XW$$

where:

- X is the original data matrix
- W is the matrix of principal component loadings

The variance explained by the k-th principal component is:

$$\text{Var}(PC_k) = \frac{\lambda_k}{\sum_{i=1}^{p} \lambda_i}$$

where λ_k is the k-th eigenvalue of the covariance matrix.

We chose the number of components to be 50 as the initial number of columns was 131. This reduction should not be too aggressive, since we want to keep the relevant features. This is our result after applying PCA using Scipy.

```
PCA Components:
[[ 1.85770079e-04  1.60419846e-01 -2.13990964e-02 ...  4.84480300e-02
   1.72382436e-05 -1.79954824e-03]
 [ 3.42630189e-01  1.58642368e-01  1.86041325e-01 ...  3.23616044e-01
   2.02433553e-04  1.41878076e-03]
 [ 3.56436509e-01 -1.01734702e-01  2.33078808e-01 ...  6.43728729e-02
   8.88080993e-05  3.68962667e-04]
 ...
 [ 9.57664157e-02  4.34445022e-02 -2.08500826e-02 ...  1.04354925e-02
   7.95314942e-06  3.91689158e-03]
 [ 3.35576714e-02 -4.05534175e-02 -1.37860794e-02 ... -4.43466530e-02
   1.48555650e-03  6.38196969e-03]
 [ 2.64368462e-02  5.02528474e-02 -9.84145244e-03 ...  5.44726655e-02
   5.88968239e-04  4.13490007e-03]]

Explained Variance Ratio:
[0.16657155 0.09112949 0.06014973 0.04673691 0.03393893 0.02687073
 0.02498255 0.02095698 0.01918413 0.01799965 0.0166909  0.01604149
 0.01454613 0.01412257 0.01285803 0.0124644  0.01148966 0.01099334
 0.01079613 0.01051626 0.01030471 0.00996746 0.00967672 0.00942003
 0.00903379 0.00896742 0.00858098 0.00804127 0.00789962 0.00777712
 0.00735276 0.00727531 0.00709098 0.00699349 0.00682945 0.00671728
 0.00665606 0.0064719  0.00620796 0.00619709 0.00607758 0.00590573
 0.00575458 0.00567002 0.00563189 0.00541423 0.00537768 0.00526794
 0.00513294 0.00491129]
```

Figure 10: The stats of PCA process

5.5 Implementation Strategy

Given the patterns observed in the residual plot, we recommend the following implementation order:

1. Apply the Box-Cox transformation to address heteroscedasticity
2. Use Robust Scaling to handle outliers
3. Apply one-hot encoding to categorical variables
4. Implement PCA if necessary for dimensionality reduction

6 Linear Regression analysis - Improved

Using the same metric, the model can be described in matrix notation as
$y = X\beta + \varepsilon$,
Y has dimension 14242×1, X has dimension 14242×50, β has dimension

50×1, and ε is a scalar. Using scikit-learn, I calculate $\hat{\beta}$ and ε :

$$\begin{bmatrix} 0.17126037 & 0.02103046 & -0.22426291 & 0.35209152 & 0.01538385 & -0.02312338 \\ 0.05194741 & -0.09037494 & 0.04368393 & -0.04104001 & -0.09044943 & -0.00954266 \\ 0.01935881 & -0.02886389 & 0.00813226 & -0.08311531 & 0.04990344 & 0.01428426 \\ -0.02916373 & -0.07604571 & -0.04569225 & 0.00422782 & -0.03399831 & -0.01341111 \\ 0.06542989 & 0.07326259 & 0.03378372 & 0.0169327 & 0.03661927 & -0.05480266 \\ -0.00302443 & -0.05669495 & -0.0118762 & -0.04834824 & 0.00587085 & -0.04518211 \\ 0.02014047 & -0.06375252 & -0.0094627 & -0.03002523 & 0.07823402 & 0.01767747 \\ 0.04556617 & -0.01160372 & 0.03785738 & -0.01854641 & 0.01495663 & 0.03842133 \\ 0.03159593 & -0.00702328 & & & & \end{bmatrix}$$

$$\begin{bmatrix} 11.406005161593802 \end{bmatrix}$$

We can now be assured that all of the assumptions are correct as shown:

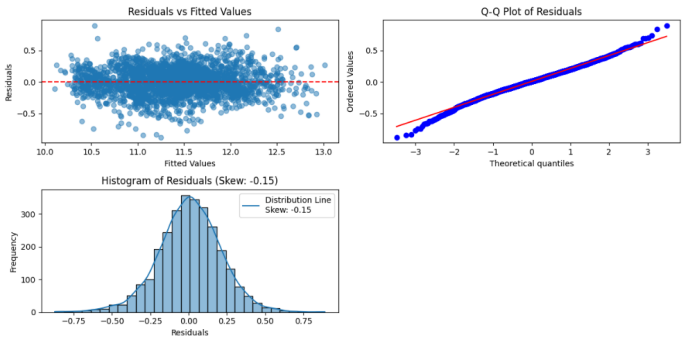


Figure 11: Assumptions of normal and homosedasticity resolved

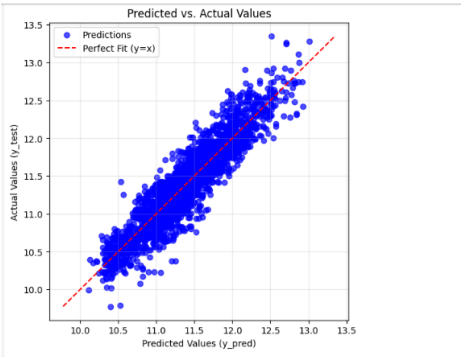


Figure 12: Ideal Fit line after processing

Based on residual diagnostics and analysis, the assumptions of linearity, homoscedasticity, and normality of residuals appear to be satisfied. Independence

and multicollinearity are also addressed through PCA. The model is suitable for inference and prediction, with no major violations detected in the evaluated assumptions.

We also see that our model performs significantly better as the accuracy has increased to 0.8562403817125805.

We also perform cross-validation and see that the performance is fairly consistent among different samples. We do not do hypothesis testing as this will be way above the level needed for this course.

7 Conclusion and Further Discussion

This study demonstrated the importance of data preprocessing in enhancing the performance of Linear Regression models. By addressing skewness, outliers, and multicollinearity, the model's accuracy improved significantly. The refined R-squared value of 0.8562 indicates that the model effectively captures the relationship between car attributes and prices.

Further Discussion

While the model's performance is robust, certain limitations remain:

- The dataset may not encompass all factors influencing car prices, such as market trends and geographic location.
- High-dimensional data from one-hot encoding may lead to overfitting in smaller datasets.
- Future work could explore non-linear models or ensemble techniques for further accuracy improvements.

8 Works Cited

- *https://github.com/AlvinJiaozhu/Linear-Regression-Model-Basketball/blob/master/Project*