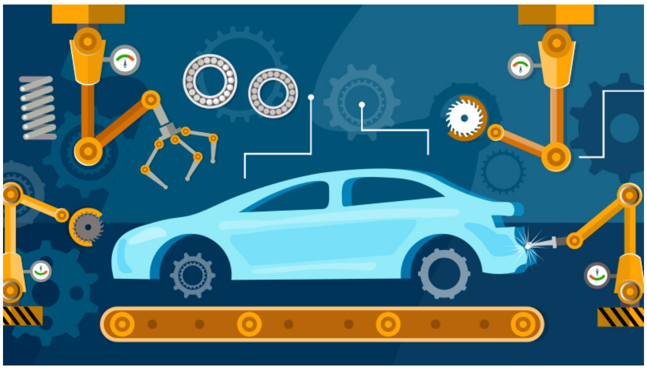
28462000 - Bourez Nicolas
25862000 - Loncour Romain

LSTAT2120
2023-2024

LSTAT2120 - Project report

1. Introduction



The automotive industry thrives on the continuous evolution and interplay of numerous factors influencing a car's market value. Predicting the price of a vehicle stands as a critical aspect for both consumers and manufacturers alike. In this report, we delve into the application of linear regression techniques to a comprehensive car dataset, which contains key specifications about cars. The primary objective is to explore how these diverse features contribute to determining the price of an automobile. Through the utilization of linear regression, we aim to model and analyze the relationships between these parameters and the final market price, thereby shedding light on the significant predictors influencing a car's valuation.[4]

Our dataset contains 200 different cars, described by 25 features :

- **Quantitative features:** wheelbase, carlength, carwidth, carheight, curbweight, enginesize, boreratio, stroke, compressionrate, horse power, peakrpm, citympg, highwaympg
- **Qualitative features:** car_ID, symboling, car name, fueltype, aspiration, doornumber, carbody, drivewheel, enginelocation, enginetype, cylindernumber, fuelsystem

A more detailed description is available in the appendix A.

In order to fit the best model, we will need to perform diverse preprocessing tasks. We will start by cleaning the dataset, as it is currently not ready-to-use. Then, a complete analysis of the data will be performed in order to spot what we have to improve. After this, we will select the most effecient model and the most relevant features using feature selection techniques in order to improve our model's predictions. Finally, we will showcase our results and how good our model is at predicting the price of a car.

2. Datacleaning

2.1. Keeping only the brand

We decided to drop the name of the model of the cars, only keeping the brand. Otherwise, this column would be irrelevant in the prediction model. If the dataset had been bigger, we could have taken a different decision but car models usually appeared only once which made them useless for our purposes. Some brands were also misspelled so we cleaned the typos.

`nissan teana` → `nissan`

`toyouta` → `toyota`

2.2. One-hot encoding

When preparing data for machine learning models, it's crucial to ensure consistency in the input format. This involves separating categorical from continuous attributes. Categorical columns are transformed into dummy columns, creating a new column for each unique category value. Each row or data point then indicates the presence of a specific category by having a value of 1 in the corresponding column. This process essentially encodes categorical information into a format that machine learning models can effectively understand and utilize.

fuelsystem
mpfi
2bbl
mpfi

$\Longrightarrow$

2bbl	idi	mapfi
0	0	1
1	0	0
0	0	0

3. Analysis of the data

3.1. Variance and mean of the numerical features

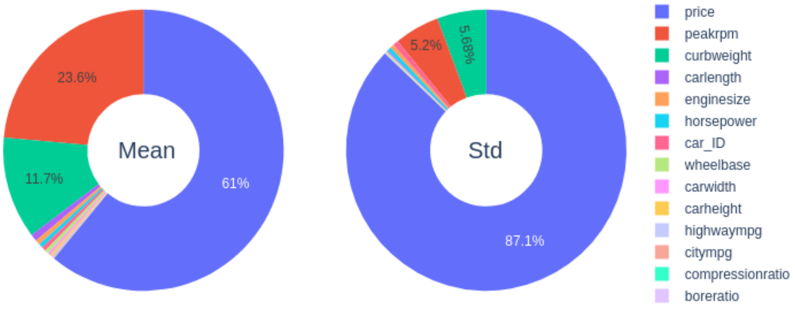


Figure 1: Piecharts of the mean and std of each features

When we look at the graphs in Figure 1, it is obvious that there is a significant difference of scale between numerical features. Three features stand out : the `price`, the `peakrpm` and the `curbweight`. Together they account for ≈ 98% of the whole numerical space. This suggests that standardizing our data is really important if we want to perform our linear regression efficiently. Standardizing allows for a fair representation of every feature.

3.2. Skewness & Kurtosis of the numerical features

As we can see in Figure 2, the **skewness** and the **kurtosis** are more diverse than the mean and variance.

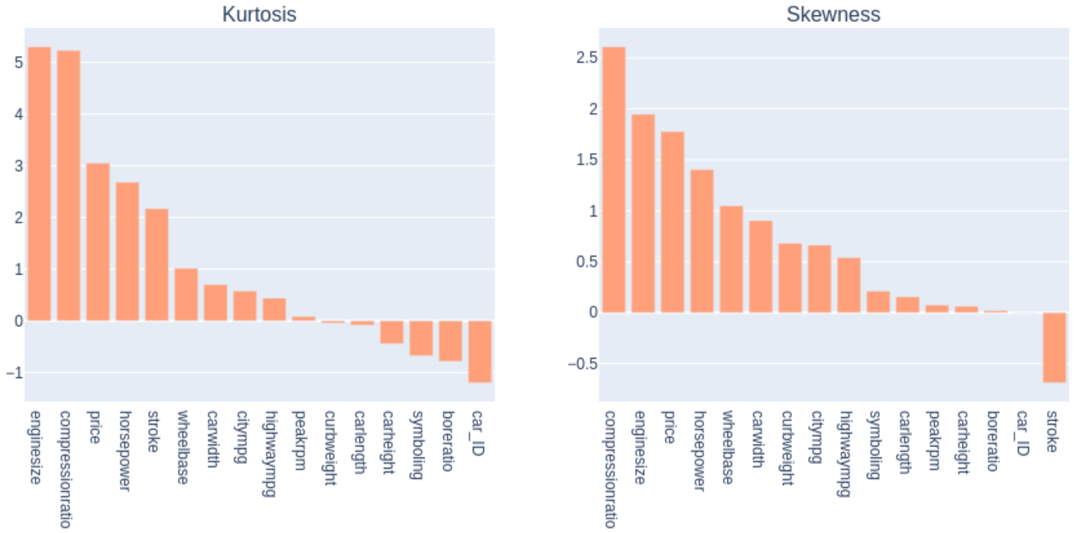


Figure 2: Skewness & Kurtosis of our dataset

Interpretation

> *In statistics, skewness is a degree of asymmetry observed in a probability distribution that deviates from the symmetrical normal distribution (bell curve) in a given set of data.[3]*

If the values of a specific independent variable (feature) are skewed, depending on the model, they may violate model assumptions or may reduce the ability to interpret feature importance so it is necessary to check for it in our features.

- **Skewness** $\in [\mathbf{-0,5,0.5}]$ identicates that data values according to a particular feature tend to follow a normal distribution. That would, in our case, be the features `car_ID`, `boreratio`, `carheight`, `peakrpm`, `carlength` and `symboling`. Note that `car_ID` has a skewness of 0, that means that the data is perfectly distributed on the right and left-hand side of the mean (with the mean being equal to the median in this case).

- **Skewness** $\in [\mathbf{-1,-0.5}[\cup]\mathbf{0.5,1}]$ identicates that data values start to be slightly skewed according to a particular feature. The following features land in this category: `highwaympg, citympg, curbweight, carwidth, wheelbase` and `stroke`

- **Skewness** $\notin [\mathbf{-1,1}]$ identicates that data values according to a particular feature are highly skewed, like `horsepower, price, enginesize` and `compressionratio`. We might be more careful about these features and apply some transformation on them.

> *Kurtosis is a statistical measure that quantifies the shape of a probability distribution. It provides information about the tails and peakedness of the distribution compared to a normal distribution.*

We can see that the general shape of the kurtosis graph follows the skewness'one. However, we might need to be careful about the feature `stroke` because it belongs to the 5 more relevant features according to the *kurtosis* criterion.

We can verify our interpretations by looking at the boxplots in the following section.

3.3. Analysis of the box plots

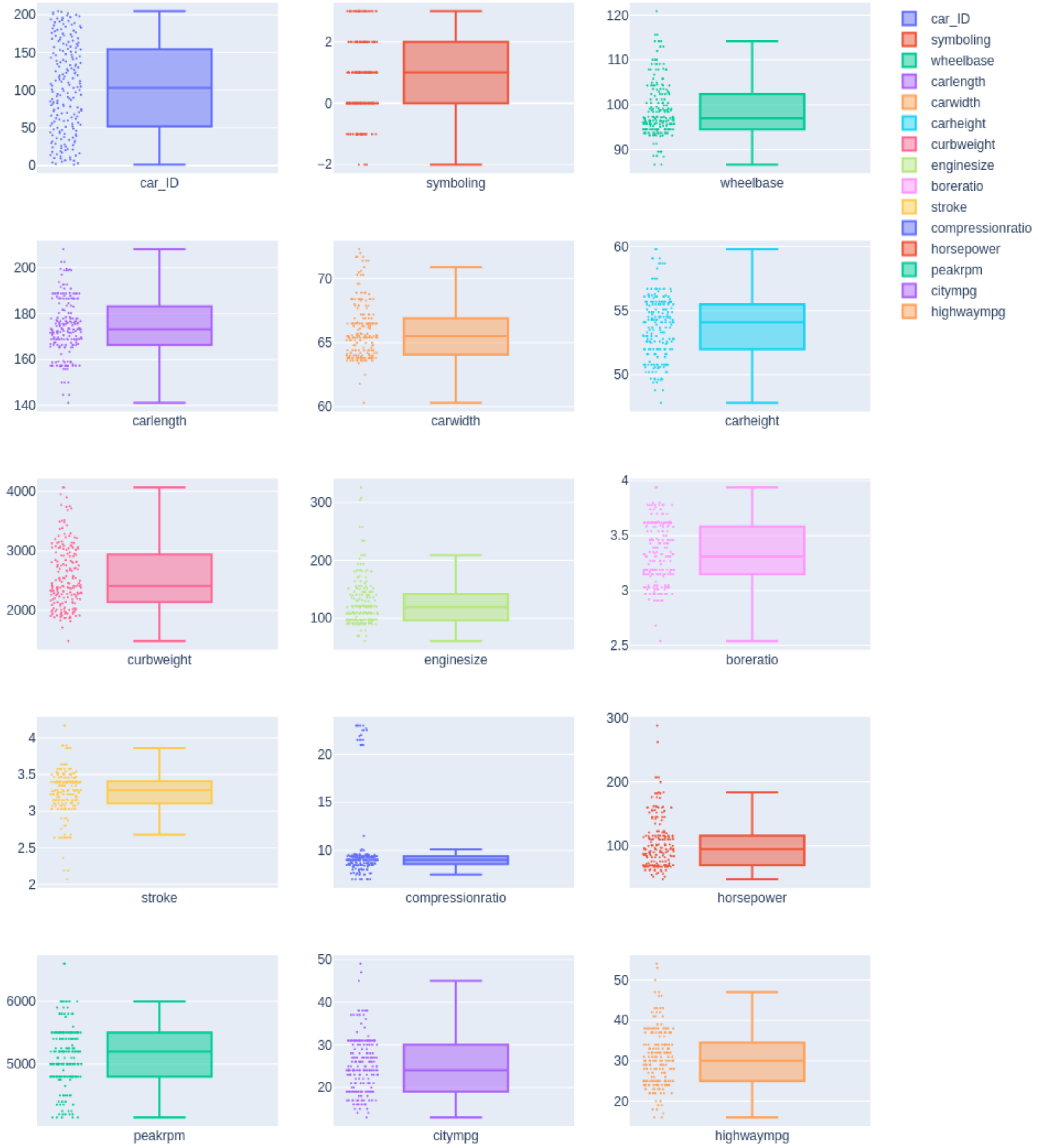


Figure 3: Boxplots of the different features

The shape of the boxplots and the distribution of the data in Figure 3 seem to confirm our previous analyses :

- Less skewed features such as `car_ID`, `boreratio`, `carheight`, `peakrpm`, `carlength` and `symboling` have boxplots that fit all the available data and graph space. It seems like they do not have any (or very few) outliers. They all have a smooth gaussian distribution shape, except for `car_ID` which has a constant probability density across its range (uniform distribution) and no extreme outliers. This justifies its kurtosis of -1.2.
- High-skewness features like `horsepower, price, enginesize` and `compressionratio` don't fit all the available space and seem to have a lot of outliers.
- The feature `compressionratio` seem to be different from the others. Indeed, its data is split into two clusters. It behaves like a categorical feature with *high_compressionratio* (> 20) and *low_compressionratio* (< 10).
- The low skweness of the feature `stroke` is justified by the fact that the tail of its distribution is spreading on the left side, identicating a lot of outliers for low data values.

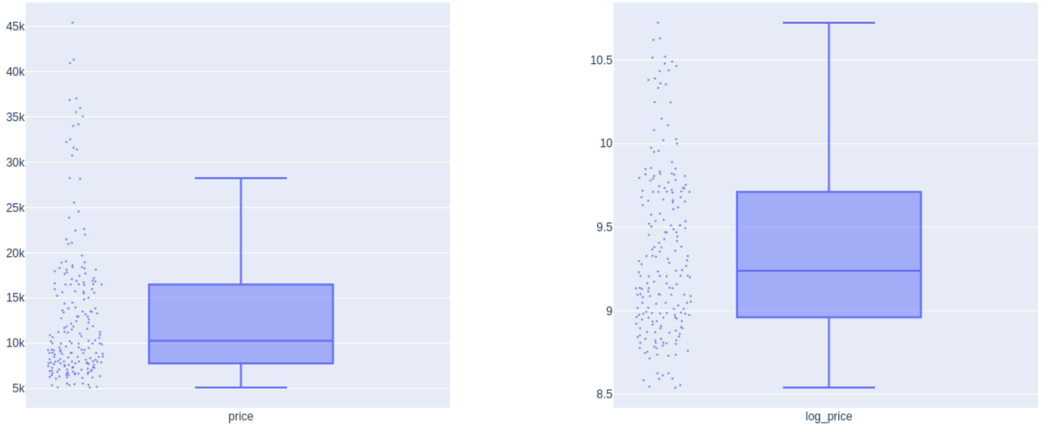


Figure 4: Emphasis on the price feature (no transformation vs. log transformation)

We will now take a closer look at the `price` feature as it is our target feature. In Figure 4, can see on the left the initial distribution of the price, which seems to have a lot of outliers on the upper part of its distribution. This tells us that our dataset contains only a few expensive cars and that these prices are widely spread. This is coherent with reality as very few people can afford a Porsche. Most people buy Ford's, Peugeot's, etc... This observation also justifies the high skewness and kurtosis of the target. However, on the right, we applied a *log* transformation on the price. The distribution of the data is now closer to a gaussian which may lead to better results. We can also apply this transformation to the other problematic features we spotted in our previous analyses.

3.4. Correlation matrix and mutual information matrix

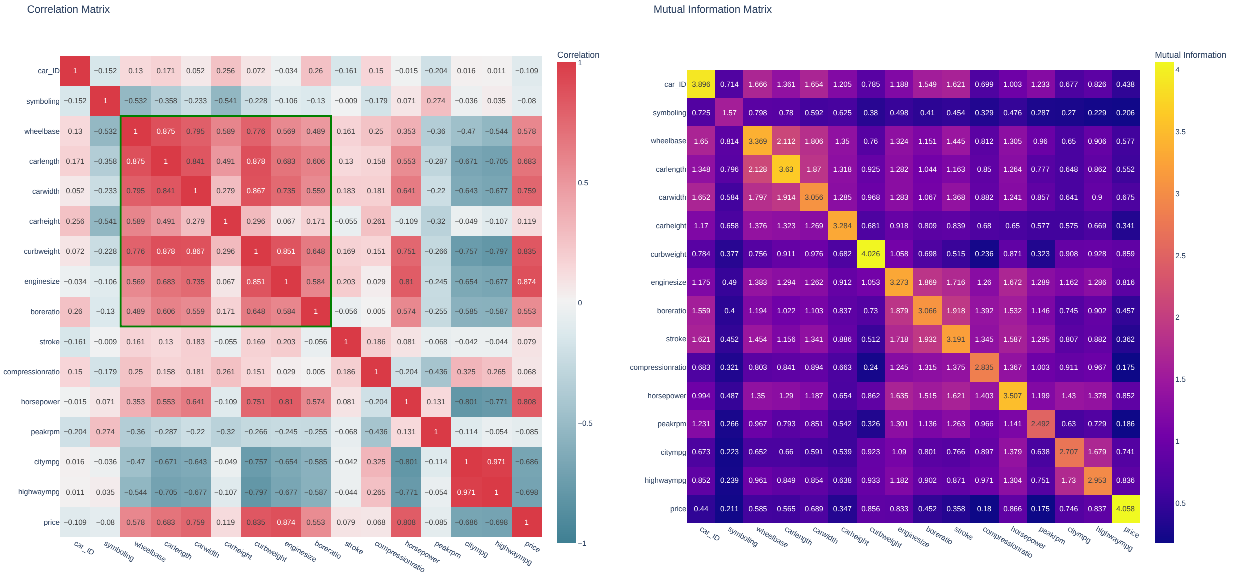


Figure 5: Correlation and Mutual Information matrices

Knowing whether collinearity between the variables and the target exists can reveal information about the significance of the variables in our regression model. We also need to spot the presence of multicollinearity between some variables[1].

On the left-hand side of Figure 5, we look at the correlation of the independent variables with the dependent ones, it shows that the variables :

- curbweight, enginesize and horsepower seem to be highly correlated with the target variable (> 0.8). We should therefore keep these variables in our model.
- symboling, stroke, compressionratio and peakrpm seem to have very little correlation with the target variable ($< 0,1$). We might think about discarding these variables to have better results.

Now if we take a look inside of the green rectangle, we can see a strong correlation between the variables. This might be a sign of multicollinearity.

Now we can take a look at the mutual information matrix (on the right in Figure 5), which gives a measure of how dependent two variables are. The mutual information index uses the entropy which takes non-linearity relationships into account. This index however lacks in its interpretability because the scale is not bounded between $[-1, 1]$ like the correlation but can theoretically output values from 0 to infinite. Anyway, we can still see that the conclusion we've made based on the correlation matrix still holds:

- the features curbweight, enginesize and horsepower are still the ones that share the most mutual information with the price but the highwaympg has an equivalent value, which implies that it is equally as important as the others.

[1] high pairwise correlation is not necessary to indicate multicollinearity, but it might give some clues.

- The features `symboling`, `compressionratio` and `peakrpm` are still the ones that share the least mutual information with the price but the `stroke` feature has a higher value than the others, which lets us think that we should maybe keep this variable.

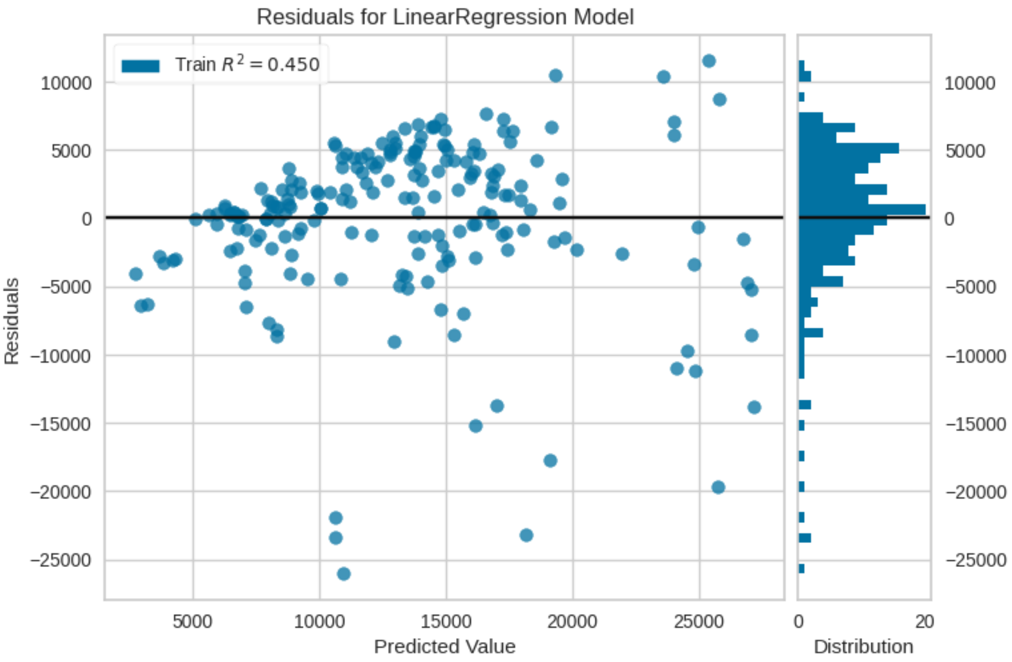
4. Strict exogeneity

Let's assume a simple OLS regression :

$$\mathbf{y} = \beta \mathbf{X} + \epsilon$$

In multiple linear regression, the strict exogeneity assumption states that the predictor variables (also called independent, or explanatory variables) are not correlated with the error term ϵ : $\mathbb{E}[\epsilon|\mathbf{X}] = 0$. Knowing $\mathbf{X}$ does not allow us to predict the coming values of Epsilon. We checked for it by :

- **Visualizing** We can see that the distribution of the residuals tends to be centered around zero.



  However, since it has a long tail to the right of the mean, the distribution deviates from a normal distribution.

- Performing a **student test** : $H_0 : \mu = 0$
  This test gives use a very high p-value (1.0) and a very low t-statistic (3.08778016e-15). The mean of residuals is thus not significantly different from zero and leads us to believe that the mean is very close to zero.

Statement

According to our OLS model and after performing the previous tests, we are confident that the assumption of strict exogeneity holds in our case

5. Basic model

Let's define a basic OLS model :

$$\mathbf{Y} = \beta_0 + \beta_0 X_1 + \cdots + \beta_{p-1} X_{p-1} + \epsilon$$

The main goal of this report is to get a model which predicts at the best of its ability the price of a car. To achieve that goal, we started with a model that takes every explanatory variable into account. We got the following results :

R-squared:	0.970	**F-statistic:**	66.74
Adj. R-squared:	0.956	**Prob (F-statistic):**	2.49e-71
AIC:	-2.250	**Log-Likelihood:**	62.125
BIC:	193.9		
Omnibus:	62.254	**Durbin-Watson:**	1.948
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	310.452
Skew:	1.170	**Prob(JB):**	3.86e-68
Kurtosis:	8.917	**Cond. No.**	1.37e+16

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 7.24e-30. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

- As we can see from the **$\mathbf{R^2}$ and $\mathbf{R^2_{adj}}$ scores**, a large percentage of the variance of the target variable is explained by the independent variables included in the model which is already pretty good.

- **The F-statistic** is also good and the **p-value** is very low, which shows that the model is significant. We will try to improve these measures later on.

- **Log-Likelihood, AIC, and BIC** are used for model comparison. As we will tune our model, we will check if we get better results than the ones we got with this basic model.

- **The Omnibus, Prob(Omnibus), Jarque-Bera(JB) and Prob(JB)** are tests which give us information about the distribution of the residuals. For both tests, we got a very low p-value which suggests that the residuals significantly deviate from a normal distribution.

- **The skew value (1.170)** identicates that the residuals' distribution is skewed to the right and **the kurtosis** indicates the presence of heavy tails, which confirms the results we got in the previous section.

- **The Durbin-Watson test** is a statistical test for the presence of autocorrelation (serial correlation) in the residuals. A value between 0 and 2 indicates positive autocorrelation, between 2 and 4 indicates negative autocorrelation, and around 2 suggests no autocorrelation at all. In our case, we have a value of 1.948 which does suggest that we have (almost) no autocorrelation in our model.

- **The Cond. No.** (condition number) is a measure of multicollinearity. A high condition number ($1.37e + 16$) indicates possible multicollinearity issues among the independent variables. Also, we can see in the notes that the smallest eigenvalue is $7.24e - 30 \sim 0$. An eigenvalue equal to

0 suggests that the matrix is ill-conditioned and thus numerically unstable. This is a sign of strong multicollinearity that we should work on later.

Analysis of the p-values

Since the linear regression p-value for each independent variable tests the null hypothesis that states if the coefficient is significant or not ($H_0 : \beta = 0$), we can interpret our p-values as [2] :

- If $p_{feature} < \alpha$: we are confident about the fact that the coefficient of the feature is significant and we reject the null hypothesis. We should thus keep this variable in our model.
- If $p_{feature} > \alpha$: we are confident about the fact that the coefficient of feature is unsignificant and we accept the null-hypothesis. We should think about discarding this feature in our model.

In our case, some features have very low (< 0.025) p-values which might indicate that we should keep them in the future models:

- `carlength, carwidth, curbweight, enginesize, boreratio, stroke, compressionratio,...`

Some others have very high p-values (> 0.3) such as:

- `const, buick, porsche, doornumber_two, dohcv, l, ohc, 4bbl, mfi, spdi`

5.1. Analysis of the 5 metrics

Non-linearity

One of the crucial assumptions that we test when building Ordinary Least Squared based or Linear Regression models is linearity in parameters. Linearity simply implies that our dependent (Y) variable can be expressed as a linear function of the explanatory variables (X). To check for linearity in our model, we decide to perform a **Rainbow-test**.

Here is what we got after performing it :

t-statistic	p-value	result
1.9479	0.0141	non-linear

We can see that the variables are non-linear with regard to the target. Maybe applying a logarithm function on some crucial variables could output better results.

Outliers and/or influential observations

To check for outliers, we based our results on the **Cook's distance**. For the basic model, we've got only two outliers and we decide to not discard them as our dataset is relatively small with only 200 observations.

Mutlicollinearity

We already know that multicollinearity was very strong by only looking at the results of our model but to get a clearer insight as to which variables influence this multicollinearity, we choose to trust the **VIF** of our variables. Unfortunately, a large number of features have a VIF value of ∞, indicating multicollinearity.

We should absolutely deal with this problem in our future model as multicollinearity indicates an overlap in the explanatory information provided by two or more predictors. These predictors are providing redundant information, making it difficult for the regression algorithm to determine how much weight every information should bear [1].

Heteroskedasticty

To Check for heteroskedasticity in our model, we choose to perform **the test of Goldfeld-Quandt**. We got these results :

t-statistic	p-value	result
1.1432	0.3379	homoskedastic

We know, after having conducted this test, that the variance is constant within the error term meaning our model doesn't suffer from heteroskedasticity.

Autocorrelation

As mentioned earlier, our model likely doesn't have autocorrelation. To confirm these results we decided to perform **the Breush-Godfrey test** which outputted the following results :

t-statistic	p-value	result
0.0949	0.7586	not autocorrelated

We are now confident in the fact that our model does not have autocorrelation.

5.2. Prediction results

When evaluating our model on the test set, we get a R^2 of 0.8802 and a $RMSE$ of 2437. These results will stand as a point of comparison for our future models.

5.3. Summary

Following is a summary of the analysis of our basic model which clarifies in which domains it should be improved :

criterium	results
non-linearity	bad
outliers	a few
mutlicollinearity	terrible
heteroskedasticty	good
autocorrelation	almost none

We will thus have to work on non-linearity and above all, multicollinearity to improve our initial model. We will also try to get better prediction results, i.e. lower the RMSE of our model on the test set.

6. Improving our model

In order to improve our model, we will focus on the basic model's weaknesses. Indeed, we need to take into account **non-linearity** and **multicollinearity**.

6.1. Improving linearity : transformation of the variables

As mentioned in the analysis of the data, some features (`horsepower, enginesize, price`) have a high skewness and kurtosis. Their distributions are far from the gaussian which deteriorates the results of the model. In this section, we choose to modify these features by applying a logarithmic transformation to them.

As the `compression ratio` is a special variable, we decide to deal with it differently. After applying the transformations, we get the following results :

R-squared:	0.974	**F-statistic:**	81.15
Adj. R-squared:	0.962	**Prob (F-statistic):**	5.70e-77
AIC:	-32.29	**Log-Likelihood:**	75.145
BIC:	157.4		
Omnibus:	1.263	**Durbin-Watson:**	2.122
Prob(Omnibus):	0.532	**Jarque-Bera (JB):**	0.914
Skewness:	-0.043	**Prob(JB):**	0.633
Kurtosis:	3.335	**Cond. No.**	1.03e+16

We can see that the majority of the results are better. There is also a better skewness and kurtosis which shows that the variables of our model follow a distribution closer to a normal. Most importantly, our Rainbow test now has a p-value of 0.1108 which shows that the relationships between features are now-linear. However, we've got a terrible RMSE (10423) so we need to improve it further.

Other variables

The `compression ratio` variable has a particular distribution as we noticed before. We decide here to convert it to a categorical variable and see if it improves the model or not. We get a better correlation with the target and the variable becomes more significant in the model, with a p-value of zero.

The variables `highwaympg` and `citympg` are also interesting as they are highly correlated (0.97), have a high mutual information and intuitively almost represent the same thing. Therefore, we decide to merge them into one single variable (by taking the mean).

6.2. Improving multicollinearity

The best way we found to improve multicollinearity and at the same time the performance of our model is to use regularization. As seen in the course, a model like Ridge is able to tackle multicollinearity, and Lasso is able to select the most relevant feature by putting some of them to zero. We decide to combine these two models by using an **Elastic Net** [5].

Elastic Net is like a Lasso regressor but with a Ridge penality. This combines the upsides of Lasso and Ridge. We also use a cross-validation based function named `GridSearchCV`[2] from the library

[2] GridSearchCV will test all combinations of parameters given in input and output the parameters performing best given a certain criterion (in our case, RMSE).

`scikit.learn` to tune our hyperparameters.

As we can see from the results in Appendix B.1, there are a lot of coefficients equal to 0 which is an expected behaviour of the Lasso regressor. But keeping these null variables into the model seems to still cause multicollinearity issues. Indeed, the matrix is still ill-conditioned and the VIF values are still huge. It seems that changing our model is not enough and that we need to discard some problematic features before really seeing the impact of the model change.

After having discarded the problematic features, we observed the results of the Elastic Net which seem to be great since we've gone from a RMSE of 10400 with the log-basic model to a value of 1939. Since this improvement, we knew we were on the right tracks, we just needed to find a solution to the multicollinearity problem.

So, after retraining the model, we still observed some coefficients equal to 0. Even if they seemed to not impact the performance, they still did by messing the multicollinearity. We decided to delete those features in order to improve the resolution of the matrix.

We obtained pretty good results applying this transformation :

- The RMSE decreased from 1939 to 1867
- The condition number dropped to 736

6.3. Feature selection

Feature selection can also be a good way to improve the model and decrease multicollinearity by discarding highly correlated predictors.

We have decided to use a backward selection wrapper on the basic model (after the log transformation). This makes sense in the context of our project since we initially started with a full model and tried to discard the problematic features, which is what the backward selection is actually doing. To accomplish this task, we used the function `SequentialFeatureSelector` from `scikit.learn`, we then selected the set of features with the smallest RMSE.

The outputted results were pretty good, but we still have multicollinearity issues. However, discarding the values with the highest VIF value resolved it. We finally get a **RMSE** of 1957 and a condition number of 874, which is not better than the Elastic Net results.

7. Final model

Starting from our basic model, we applied transformations, tested other regressors and applied feature selection, we finally get our best model, and here are the results[3] :

R-squared:	0.959	0.970	**F-statistic:**	95.34	66.74
Adj. R-squared:	0.949	0.956	**Prob (F-statistic):**	2.30e-85	2.49e-71
AIC:	7.947	-2.250	**Log-Likelihood:**	32.027	62.125
BIC:	123.7	193.9			
Omnibus:	0.255	62.254	**Durbin-Watson:**	2.083	1.948
Prob(Omnibus):	0.880	0.000	**Jarque-Bera (JB):**	0.308	310.452
Skew:	0.088	1.170	**Prob(JB):**	0.857	3.86-68
Kurtosis:	2.903	8.917	**Cond. No.**	736	1.37+16

In comparison to the initial model (in red), we can see that the main indicators have been improved:

- The first part of the table seems to get worse results. This is due to the fact that the model became more adapted to the test set and is less overfitted. We can see that the F-statistic is still good, showing that the model is relevant. Also, the BIC, which penalizes the model more for its complexity, gives a better result since our model is simpler.
- The right-hand side of the table shows better results in all metrics due to all the improvements we have made along this report.

We can see in the following table that the problematic metric have been improved :

	f-test BM	p-value BM	result BM	f-test FM	p-value FM	result FM
non-linearity	1.9479	0.0141	non-linear	1.4311	0.1108	linear
heteroskedasticty	1.1432	0.3379	homo	0.7602	0.8469	homo
autocorrelation	0.0949	0.7586	not auto	0.5336	0.4663	not auto
VIF		∞			28.98	
RMSE		2437			1867	

NB : the linear value for FM is from OLS_log as we couldn't perfrom the test for ElasticNet

7.1. Comparison of the predictions

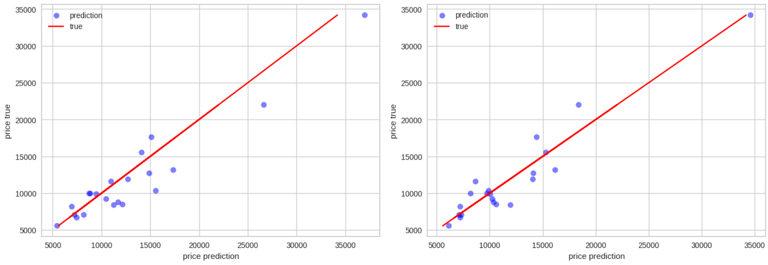


Figure 6: Prediction basic model vs Prediction final model

The closer the points are from the red line, the better the predictions are. We can see that the final model can predict the outliers better which is a good thing in the domain of predictions.

[3] full results, including test of significance of the coefficients are available at Appendix B.2

7.2. Test for significance of the estimated coefficients

Some features are more important than others, and as we standardize them, we can rank their level of significance by looking at their coefficients and perform the test $H_0 : \beta_i = 0$, with β_i one of the coefficents. From a general point of vue, the sign of the coefficients seem to be related to the sign of their correlation with the price, which is good sign of good performance of the model. `carwidth`, `curbweight` and `horsepower` are the quantitative variables with the least p-value, which is relevant because these are 3 of the most correlated quantitative variable with the price.

When looking a the brand of the cars: BMW, Porsche, Volvo have a strong positive coefficient which is relevant when we know that these cars are expensive in real life. On the other side, brand like Peugeot and Mitsubishi has a negative coefficient. Then, some features in our model still have a high p-value, meaning they might not be relevant to our model. For example, the qualitative variables `nissan`, `rwd` and `cylindernumber_four` have a p-value of 1 meaning they are totally useless, they probably share information with other variable like fwd for rwd.

And finally, we can see that the most significant coeficients are `rear` and `hatchback`. This could be due to the association of rear-engine cars with high-performance or luxury vehicles and that hatchbacks, despite being practical, might be perceived as less expensive or lower in market value compared to other body types.

7.3. Test a linear combination of at least two coefficients

In the context of our prediction problem, testing the combination of the variables `carwidth` and `wheelbase` seems to be a relevant choice since these factors are related by their physical attributes and often exhibit correlations in automotive design. Let the null hypothesis be $H_0 : \beta_{\text{carwidth}} + \beta_{\text{wheelbase}} = 0$, the obtained F-test value of 9.0865 and the associated p-value of 0.0030 strongly support the relevance of combining these two variables in our model.

7.4. Test a subset of coefficients (at least two)

Here, we decide to combine the variables related to the size of the car, i.e. `carwidth`, `carlength` and `carheight`, and see if it is really relevant to measure a car to predict its price. Let be the null-hypothesis Let the null hypothesis be $H_0 : \beta_{\text{carwidth}} = \beta_{\text{carlength}} = \beta_{\text{carheight}} = 0$, the obtained F-test value of 3.7452 and the associated p-value of 0.01247 support the fact that we choose keep the metric predictors in our model.

7.5. Prediction intervals for the validation data

When we assess the coverage percentage of our prediction intervals for the validation data, it yields a 100% outcome, indicating that all predicted values align within the calculated prediction intervals. However, this perfect match against the nominal level of 95% suggests a potential issue with overfitting in our model, meaning that it fits the training data too closely. Consequently, when calculating prediction intervals, the model might be excessively conservative, resulting in wider intervals that encompass the true values with a higher probability. To address this, simplifying our model further could mitigate these potential overfitting tendencies.

8. Conclusion

After extensive model finetuning efforts, we finaly build a model yielding significant results. When compared to the basic model, our predictions have notably improved. However, despite the decrease in green points observed with the both tolerance[4], indicating progress, we are still far away from perfect accuracy.

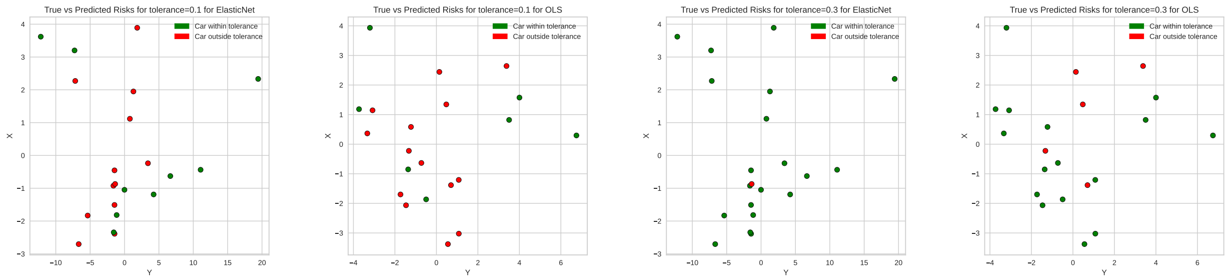
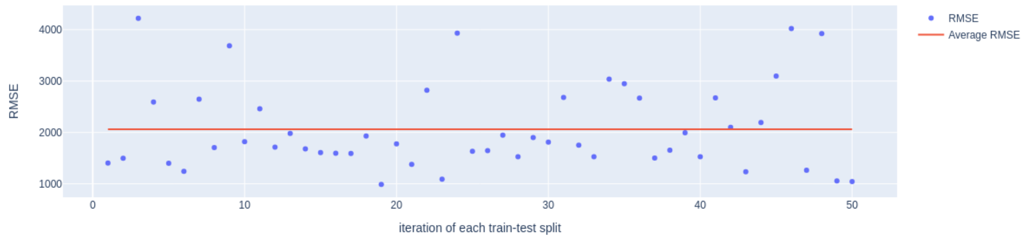


Figure 7: Representation of the points in two dimensions using PCA

This lack of accuracy might be due the small number of observations we worked with. Indeed, each time we've splitted our dataset randomly, our results were sometimes pretty good, and sometimes very bad, as showed by the following graph :



Also there exist other factors which influence the price of a car like the vehicle's mileage, maintenance history, market demand, regional factors, economic conditions, and the overall condition of the car. With this limited dataset, achieving a prediction margin of 30% is satisfactory, considering the complexity of factors influencing car prices that may not be fully captured or represented within this smaller dataset.

[4] we defined our tolerance diff(true - predict)/true

References

[1] Scott Duda. *Identifying and Addressing Multicollinearity in Regression Analysis — scottmduda.medium.com.* https://scottmduda.medium.com/identifying-and-addressing-multicollinearity-in-regression-analysis-ca86a21a347e#:~:text=A%20quick%20way%20to%20identify,just%20a%20rule%20of%20thumb.. [Accessed 17-12-2023].

[2] Jim Frost. *How to Interpret P-values and Coefficients in Regression Analysis — statisticsbyjim.com.* https://statisticsbyjim.com/regression/interpret-coefficients-p-values-regression/. [Accessed 17-12-2023].

[3] Suvarna Gawali. *Skewness and Kurtosis: Quick Guide (Updated 2023) — analyticsvidhya.com.* https://www.analyticsvidhya.com/blog/2021/05/shape-of-data-skewness-and-kurtosis/. [Accessed 16-12-2023].

[4] *Sourcing in Automotive Industry Moving Toward Regional Suppliers — beroeinc.com.* https://www.beroeinc.com/blog/sourcing-in-automotive-industry-moving-toward-regional-suppliers/. [Accessed 04-12-2023].

[5] CFI Team. *Elastic Net.* https://corporatefinanceinstitute.com/resources/data-science/elastic-net/. Accessed 21-12-2023.

A. Description of the variables

Name of the Feature	**Description**
car_ID	ID of the car, an integer from 1 to 205
symboling	Symbol of the car
car name	Name of the car
fueltype	Type of fuel that the car is using: gas or diesel
aspiration	Type of aspiration: std or turbo
doornumber	Total number of doors that the car has
carbody	Type of body of the car: convertible, hatchback, sedan, wagon, hardtop
drivewheel	Type of drivewheel: rwd or fwd
enginelocation	Location of the engine in the car: front
wheelbase	Horizontal distance between the centers of the front and rear wheels
carlength	Horizontal distance between the front and the back of the car
carwidth	Largest width of the car
carheight	Height of the car
curbweight	Total weight of a vehicle, inclusive of standard equipment and necessary operating fluids, but without passengers or cargo
enginetype	Type of engine: dohc, ohcv, ohc, rotor, l, ohcf
cylindernumber	Number of cylinders in the engine: 4, 6, or 8 typically
enginesize	Size of the engine
fuelsystem	Fuel delivery system: mpfi, 2bbl, 1bbl, spfi, 4bbl, idi, spdi
boreratio	Ratio between cylinder bore diameter and piston stroke length
stroke	Phase of the engine's cycle
compressionratio	Ratio between the volume of the cylinder and combustion chamber in an internal combustion engine
horsepower	Measurement used to calculate how quickly the force is produced from a vehicle's engine
peakrpm	RPM measures how many times the engine's crankshaft makes one full rotation every minute
citympg	Miles per gallon in city driving conditions
highwaympg	Miles per gallon in highway driving conditions
price	Price of the car in US dollars

B. results basic model

Dep. Variable:	price	**R-squared:**	0.970
Model:	OLS	**Adj. R-squared:**	0.956
Method:	Least Squares	**F-statistic:**	66.74
Date:	Tue, 19 Dec 2023	**Prob (F-statistic):**	2.49e-71
Time:	16:27:32	**Log-Likelihood:**	62.125
No. Observations:	184	**AIC:**	-2.250
Df Residuals:	123	**BIC:**	193.9
Df Model:	60		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P> |t|**	**[0.025**	**0.975]**
const	1.2738	0.631	2.018	0.046	0.024	2.524
car_ID	0.4225	0.436	0.969	0.334	-0.440	1.285
symboling	-0.0271	0.040	-0.669	0.504	-0.107	0.053
wheelbase	0.1608	0.075	2.131	0.035	0.011	0.310
carlength	-0.2793	0.087	-3.218	0.002	-0.451	-0.108
carwidth	0.2129	0.065	3.266	0.001	0.084	0.342
carheight	-0.0760	0.047	-1.614	0.109	-0.169	0.017
curbweight	0.4786	0.148	3.231	0.002	0.185	0.772
enginesize	0.4949	0.145	3.418	0.001	0.208	0.782
boreratio	-0.1120	0.066	-1.708	0.090	-0.242	0.018
stroke	-0.0289	0.046	-0.635	0.527	-0.119	0.061
compressionratio	-0.4250	0.267	-1.590	0.114	-0.954	0.104
horsepower	-0.0571	0.125	-0.457	0.648	-0.304	0.190
peakrpm	0.1423	0.040	3.532	0.001	0.063	0.222
citympg	-0.0254	0.116	-0.219	0.827	-0.255	0.204
highwaympg	0.1212	0.102	1.190	0.236	-0.080	0.323
audi	-0.0143	0.058	-0.249	0.804	-0.128	0.100
bmw	0.5497	0.323	1.704	0.091	-0.089	1.188
buick	-0.4116	0.548	-0.751	0.454	-1.496	0.673
chevrolet	-0.9617	0.373	-2.578	0.011	-1.700	-0.223
dodge	-1.0745	0.348	-3.087	0.002	-1.764	-0.385
honda	-0.5034	0.457	-1.102	0.272	-1.407	0.400
isuzu	-0.9391	0.488	-1.926	0.056	-1.904	0.026
jaguar	-0.7066	0.420	-1.684	0.095	-1.537	0.124
mazda	-0.8867	0.540	-1.643	0.103	-1.955	0.182
mercury	-1.0392	0.670	-1.550	0.124	-2.366	0.288
mitsubishi	-1.5593	0.702	-2.222	0.028	-2.948	-0.171
nissan	-1.2367	0.808	-1.531	0.128	-2.836	0.362
peugeot	-1.1790	0.553	-2.134	0.035	-2.273	-0.085
plymouth	-1.7384	0.986	-1.764	0.080	-3.690	0.213
porsche	-0.0705	0.524	-0.135	0.893	-1.107	0.966
renault	-1.5929	1.012	-1.574	0.118	-3.596	0.410
saab	-0.8643	1.065	-0.812	0.419	-2.972	1.244
subaru	-1.2836	0.587	-2.185	0.031	-2.446	-0.121
toyota	-1.7471	1.258	-1.388	0.168	-4.238	0.744

volkswagen	-1.7654	1.431	-1.233	0.220	-4.599	1.068
volvo	-1.7879	1.465	-1.220	0.225	-4.689	1.113
gas	-0.2635	0.447	-0.589	0.557	-1.149	0.622
turbo	0.2299	0.107	2.157	0.033	0.019	0.441
doornumber_two	0.0026	0.061	0.043	0.966	-0.118	0.124
hardtop	-0.0765	0.203	-0.377	0.707	-0.478	0.325
hatchback	-0.2172	0.218	-0.996	0.321	-0.649	0.214
sedan	-0.0781	0.227	-0.345	0.731	-0.527	0.370
wagon	-0.1103	0.233	-0.473	0.637	-0.572	0.351
fwd	-0.0187	0.109	-0.172	0.864	-0.235	0.197
rwd	0.0364	0.147	0.248	0.804	-0.254	0.327
rear	1.0831	0.181	5.980	0.000	0.725	1.442
dohcv	-1.1535	0.609	-1.894	0.061	-2.359	0.052
l	-0.4266	0.356	-1.199	0.233	-1.131	0.277
ohc	0.0476	0.158	0.301	0.764	-0.265	0.360
ohcf	-0.2006	0.560	-0.358	0.721	-1.308	0.907
ohcv	-0.3051	0.170	-1.800	0.074	-0.641	0.030
rotor	0.2711	0.310	0.874	0.384	-0.343	0.885
cylindernumber_five	-0.8057	0.405	-1.987	0.049	-1.608	-0.003
cylindernumber_four	-0.3932	0.488	-0.806	0.422	-1.359	0.573
cylindernumber_six	-0.5283	0.373	-1.418	0.159	-1.266	0.209
cylindernumber_three	0.7524	0.432	1.740	0.084	-0.104	1.608
cylindernumber_twelve	-0.5396	0.674	-0.800	0.425	-1.875	0.795
cylindernumber_two	0.2711	0.310	0.874	0.384	-0.343	0.885
2bbl	0.6206	0.243	2.554	0.012	0.140	1.101
4bbl	0.1659	0.343	0.484	0.629	-0.512	0.844
idi	1.5373	0.653	2.355	0.020	0.245	2.829
mfi	0.2647	0.357	0.742	0.459	-0.441	0.971
mpfi	0.4018	0.235	1.710	0.090	-0.063	0.867
spdi	0.4226	0.277	1.525	0.130	-0.126	0.971
spfi	0.5154	0.371	1.388	0.168	-0.220	1.250

Omnibus:	62.254	**Durbin-Watson:**	1.948
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	310.452
Skew:	1.170	**Prob(JB):**	3.86e-68
Kurtosis:	8.917	**Cond. No.**	1.37e+16

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 7.24e-30. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.

B.1. results Elastic Net

Dep. Variable:	price	**R-squared:**	0.952
Model:	OLS	**Adj. R-squared:**	0.930
Method:	Least Squares	**F-statistic:**	42.88
Date:	Wed, 20 Dec 2023	**Prob (F-statistic):**	1.51e-60
Time:	13:22:35	**Log-Likelihood:**	18.559
No. Observations:	184	**AIC:**	80.88
Df Residuals:	125	**BIC:**	270.6
Df Model:	58		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P> |t|**	**[0.025**	**0.975]**
car_ID	-0.0929	0.555	-0.167	0.867	-1.192	1.006
symboling	0.0523	0.051	1.016	0.311	-0.050	0.154
wheelbase	0.0250	0.094	0.266	0.790	-0.161	0.211
carlength	0.0154	0.108	0.142	0.887	-0.199	0.230
carwidth	0.1408	0.081	1.731	0.086	-0.020	0.302
carheight	-0.0073	0.059	-0.124	0.902	-0.125	0.110
curbweight	0.4267	0.184	2.317	0.022	0.062	0.791
enginesize	0	0.200	0	1.000	-0.395	0.395
boreratio	0	0.082	0	1.000	-0.163	0.163
stroke	-0.0556	0.058	-0.958	0.340	-0.171	0.059
compressionratio	0.0674	0.290	0.232	0.817	-0.507	0.642
horsepower	0.2558	0.171	1.500	0.136	-0.082	0.593
peakrpm	-0	0.054	-0	1.000	-0.107	0.107
audi	0.0106	0.071	0.148	0.882	-0.131	0.152
bmw	0.4893	0.398	1.228	0.222	-0.299	1.278
buick	0.1555	0.705	0.221	0.826	-1.239	1.551
chevrolet	-0	0.460	-0	1.000	-0.911	0.911
dodge	-0.0775	0.440	-0.176	0.860	-0.948	0.792
honda	-0	0.579	-0	1.000	-1.146	1.146
isuzu	0	0.611	0	1.000	-1.210	1.210
jaguar	-0	0.523	-0	1.000	-1.035	1.035
mazda	0.0763	0.685	0.111	0.911	-1.279	1.432
mercury	-0	0.844	-0	1.000	-1.670	1.670
mitsubishi	-0.1521	0.891	-0.171	0.865	-1.915	1.611
nissan	-0.0197	1.031	-0.019	0.985	-2.061	2.021
peugeot	-0.2068	0.686	-0.301	0.764	-1.564	1.151
plymouth	-0	1.258	-0	1.000	-2.490	2.490
porsche	0.2872	0.637	0.451	0.653	-0.973	1.547
renault	-0	1.304	-0	1.000	-2.582	2.582
saab	0.0238	1.366	0.017	0.986	-2.679	2.726
subaru	-0.1299	0.750	-0.173	0.863	-1.614	1.354
toyota	-0	1.608	-0	1.000	-3.182	3.182
volkswagen	0	1.836	0	1.000	-3.634	3.634
volvo	0.1175	1.877	0.063	0.950	-3.598	3.833
gas	0	1.191	0	1.000	-2.357	2.357

turbo	0.0258	0.145	0.178	0.859	-0.261	0.313
doornumber_two	-0.0402	0.077	-0.525	0.601	-0.192	0.111
hardtop	0.0025	0.245	0.010	0.992	-0.482	0.487
hatchback	-0.1553	0.265	-0.585	0.559	-0.680	0.370
sedan	-0	0.277	-0	1.000	-0.548	0.548
wagon	-0.1361	0.288	-0.472	0.638	-0.707	0.435
fwd	-0.0507	0.136	-0.372	0.710	-0.320	0.219
rwd	0.0565	0.183	0.309	0.758	-0.305	0.418
rear	0.7050	0.209	3.366	0.001	0.290	1.119
dohcv	-0	0.730	-0	1.000	-1.446	1.446
l	-0	0.453	-0	1.000	-0.897	0.897
ohc	0.0249	0.193	0.129	0.898	-0.358	0.407
ohcf	0	0.706	0	1.000	-1.397	1.397
ohcv	-0.0447	0.210	-0.213	0.832	-0.460	0.371
rotor	0	0.376	0	1.000	-0.744	0.744
cylindernumber_five	-0	0.409	-0	1.000	-0.810	0.810
cylindernumber_four	-0.0672	0.524	-0.128	0.898	-1.104	0.970
cylindernumber_six	-0	0.396	-0	1.000	-0.783	0.783
cylindernumber_three	0	0.515	0	1.000	-1.020	1.020
cylindernumber_twelve	-0	0.620	-0	1.000	-1.226	1.226
cylindernumber_two	0	0.376	0	1.000	-0.744	0.744
2bbl	-0.0548	0.306	-0.179	0.858	-0.661	0.552
4bbl	0	0.432	0	1.000	-0.855	0.855
idi	0	0.222	0	1.000	-0.439	0.439
mfi	-0	0.448	-0	1.000	-0.887	0.887
mpfi	0.0783	0.298	0.263	0.793	-0.511	0.668
spdi	0	0.351	0	1.000	-0.694	0.694
spfi	-0	0.468	-0	1.000	-0.926	0.926
mpg	0.0029	0.011	0.251	0.802	-0.020	0.025

Omnibus:	0.372	**Durbin-Watson:**	2.092
Prob(Omnibus):	0.830	**Jarque-Bera (JB):**	0.283
Skew:	0.096	**Prob(JB):**	0.868
Kurtosis:	3.005	**Cond. No.**	nan

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is -9.42e-14. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.

B.2. Results of our Final Model

Dep. Variable:	price	**R-squared (uncentered):**	0.959
Model:	OLS	**Adj. R-squared (uncentered):**	0.949
Method:	Least Squares	**F-statistic:**	95.34
Date:	Wed, 20 Dec 2023	**Prob (F-statistic):**	2.30e-85
Time:	13:21:25	**Log-Likelihood:**	32.027
No. Observations:	184	**AIC:**	7.947
Df Residuals:	148	**BIC:**	123.7
Df Model:	36		
Covariance Type:	nonrobust		

	coef	std err	t	P> |t|	[0.025	0.975]
car_ID	-0.0840	0.034	-2.493	0.014	-0.151	-0.017
symboling	0.0517	0.033	1.567	0.119	-0.014	0.117
wheelbase	0.0743	0.060	1.240	0.217	-0.044	0.193
carlength	-0.0256	0.077	-0.331	0.741	-0.178	0.127
carwidth	0.1350	0.057	2.384	0.018	0.023	0.247
carheight	-0.0634	0.036	-1.768	0.079	-0.134	0.007
curbweight	0.4975	0.090	5.529	0.000	0.320	0.675
stroke	-0.0378	0.032	-1.177	0.241	-0.101	0.026
compressionratio	0.0339	0.042	0.814	0.417	-0.048	0.116
horsepower	0.1564	0.084	1.871	0.063	-0.009	0.322
audi	0.0299	0.032	0.934	0.352	-0.033	0.093
bmw	0.7434	0.152	4.880	0.000	0.442	1.044
buick	0.3019	0.151	1.993	0.048	0.003	0.601
dodge	-0.1815	0.115	-1.582	0.116	-0.408	0.045
mazda	0.1473	0.081	1.825	0.070	-0.012	0.307
mitsubishi	-0.2336	0.097	-2.397	0.018	-0.426	-0.041
nissan	0	0.079	0	1.000	-0.155	0.155
peugeot	-0.3410	0.167	-2.041	0.043	-0.671	-0.011
porsche	0.3656	0.318	1.151	0.251	-0.262	0.993
saab	0.1962	0.156	1.259	0.210	-0.112	0.504
subaru	-0.2280	0.128	-1.776	0.078	-0.482	0.026
volvo	0.2154	0.121	1.786	0.076	-0.023	0.454
turbo	0.1129	0.080	1.416	0.159	-0.045	0.270
doornumber_two	-0.0694	0.061	-1.143	0.255	-0.189	0.051
hardtop	0.0075	0.113	0.067	0.947	-0.215	0.230
hatchback	-0.1529	0.058	-2.655	0.009	-0.267	-0.039
wagon	-0.1075	0.066	-1.617	0.108	-0.239	0.024
fwd	-0.0646	0.106	-0.612	0.541	-0.273	0.144
rwd	-0	0.130	-0	1.000	-0.256	0.256
rear	1.0845	0.317	3.417	0.001	0.457	1.712
ohc	-0.0586	0.096	-0.609	0.544	-0.249	0.132
ohcv	-0.0578	0.121	-0.476	0.635	-0.298	0.182
cylindernumber_four	0	0.090	0	1.000	-0.179	0.179
2bbl	-0.0612	0.079	-0.776	0.439	-0.217	0.095
mpfi	0.0965	0.087	1.112	0.268	-0.075	0.268
mpg	0.0029	0.005	0.599	0.550	-0.007	0.012

Omnibus:	0.255	**Durbin-Watson:**	2.083
Prob(Omnibus):	0.880	**Jarque-Bera (JB):**	0.308
Skew:	0.088	**Prob(JB):**	0.857
Kurtosis:	2.903	**Cond. No.**	736.

Notes:
[1] R^2 is computed without centering (uncentered) since the model does not contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.