Car Prices Analysis and Prediction

Bembhy, Martín-Villa, Fourati, Iannotta

The company

Emerging marketplace for used cars in Europe

Founded in June 2020

9 employees in Milan HQ

Revenue growth in 2021 +12% over the year before

The objective of the analysis

Be able to predict car prices to launch an MVP for the startup and test market fit.

For sellers: Set the right price for your car and cash in its true value

For buyers: Check if the car you are about to buy is either overpriced or a good deal

The dataset

For this analysis we extracted a sample of data from our database that would be representative of the used car market.

The dataset is very detailed as we had the ability to work with **27 different variables**, giving us the power to build a meaningful model from it.

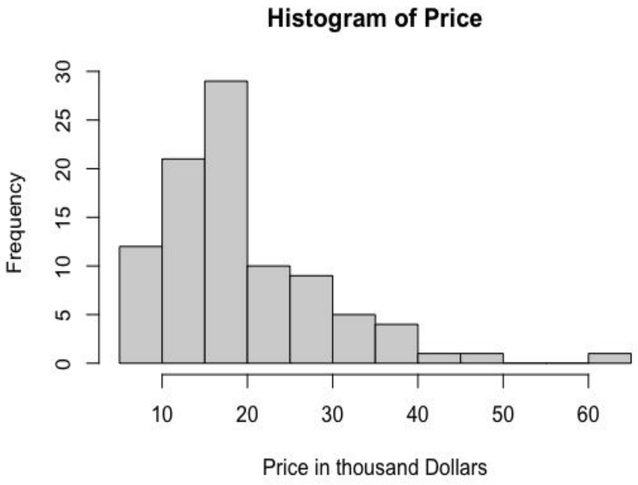
Methodology

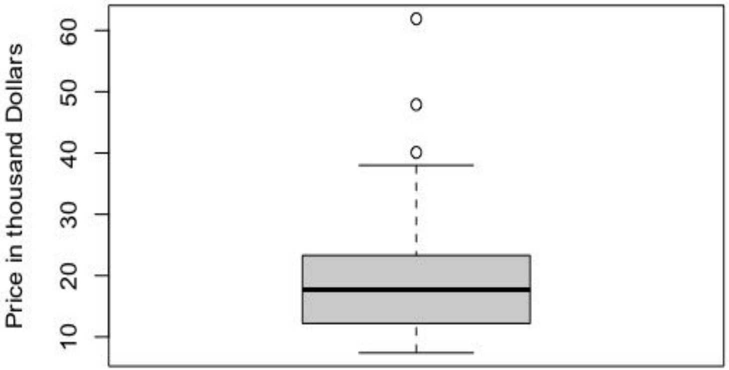
The different steps we went through:

1. **Descriptive analysis of the price**
2. **Data preparation and cleaning**
3. **Linear regression Model**
4. **Ridge and Lasso Models**
5. **Linear Model Optimisation**

Target variable analysis

Price





Min	1st Q	Median	Mean	3rd Q	Max	Kurtosis	Skewness
7.40	12.20	17.70	19.51	23.30	61.90	6.183	1.508

Data preparation

Redundant variables

Reading the dataset documentation:

- **Min. and Max. Price**
  - Price is calculated as average of maximum and minimum
- **Model**
  - Is unique for all rows
- **Make**
  - Is a concatenation of manufacturer + model

Now we are left with **23 variables**

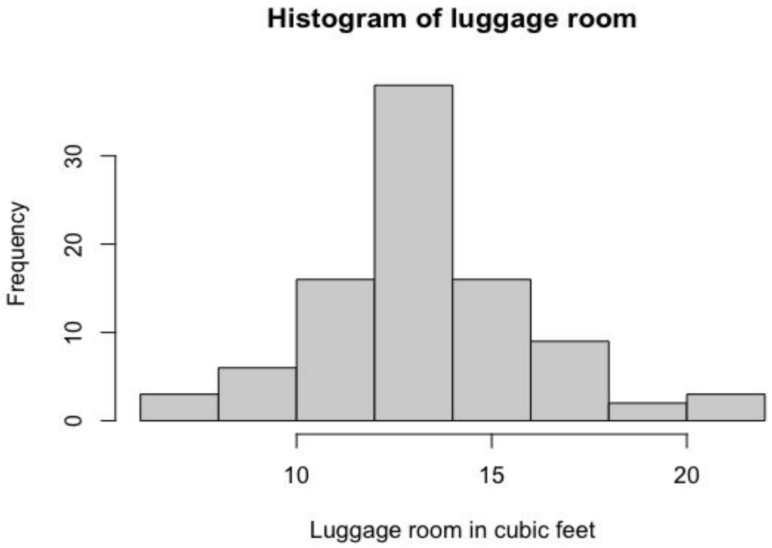
Data preparation

Handling missing values

The actions we did on the dataframe:

- Cars with 2 seats have null values in Rear seat room so we replaced those with 0

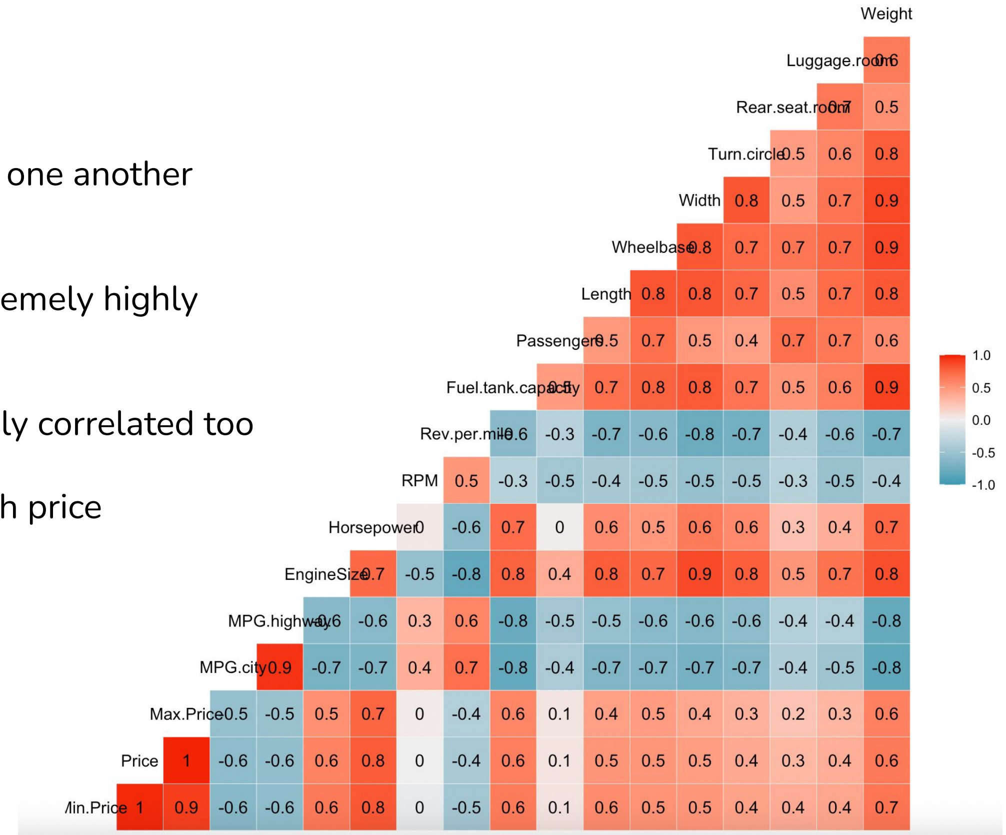
- 11 null values for luggage room. We replace those with the median



Data preparation

Collinearity

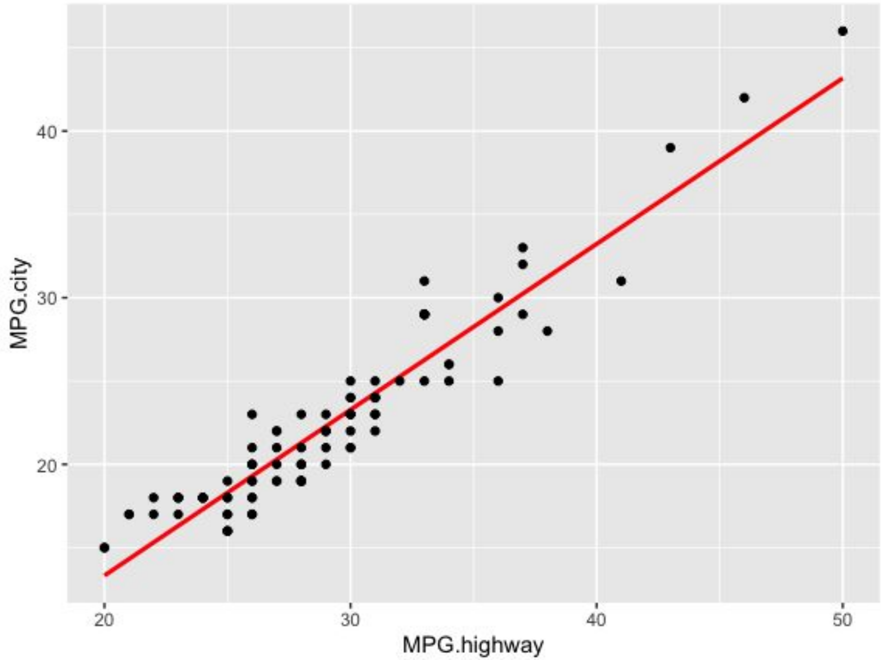
- Predictor highly correlated between one another
- MPG.city and MPG.highway are extremely highly correlated
- As said, MaxPrice and MinPrice highly correlated too
- Horsepower: highest correlation with price



	Price	Max.Price	MPG.city	MPG.highway	EngineSize	Horsepower	RPM	Rev.per.mile	Fuel.tank.capacity	Passengers	Length	Wheelbase	Width	Turn.circle	Rear.seat.room	Luggage.room	Weight
Weight																	
Luggage.room																	0.6
Rear.seat.room																0.7	0.5
Turn.circle															0.5	0.6	0.8
Width														0.8	0.5	0.7	0.9
Wheelbase													0.8	0.7	0.7	0.7	0.9
Length												0.8	0.8	0.7	0.5	0.7	0.8
Passengers											0.5	0.7	0.5	0.4	0.7	0.7	0.6
Fuel.tank.capacity										0.5	0.7	0.8	0.8	0.7	0.5	0.6	0.9
Rev.per.mile									-0.6	-0.3	-0.7	-0.6	-0.8	-0.7	-0.4	-0.6	-0.7
RPM								0.5	-0.3	-0.5	-0.4	-0.5	-0.5	-0.5	-0.3	-0.5	-0.4
Horsepower							0	-0.6	0.7	0	0.6	0.5	0.6	0.6	0.3	0.4	0.7
EngineSize						0.7	-0.5	-0.8	0.8	0.4	0.8	0.7	0.9	0.8	0.5	0.7	0.8
MPG.highway					-0.6	-0.6	0.3	0.6	-0.8	-0.5	-0.5	-0.6	-0.6	-0.6	-0.4	-0.4	-0.8
MPG.city				0.9	-0.7	-0.7	0.4	0.7	-0.8	-0.4	-0.7	-0.7	-0.7	-0.7	-0.4	-0.5	-0.8
Max.Price			-0.5	-0.5	0.5	0.7	0	-0.4	0.6	0.1	0.4	0.5	0.4	0.3	0.2	0.3	0.6
Price		1	-0.6	-0.6	0.6	0.8	0	-0.4	0.6	0.1	0.5	0.5	0.5	0.4	0.3	0.4	0.6
Min.Price	1	0.9	-0.6	-0.6	0.6	0.8	0	-0.5	0.6	0.1	0.6	0.5	0.5	0.4	0.4	0.4	0.7

Data preparation

MPG.City vs MPG.Highway



- We can see that they are extremely correlated (0.94)

- We can discard one of them and keep the one that is the more correlated with the price

- MPG city is more highly correlated, therefore we keep this one

Linear regression Model

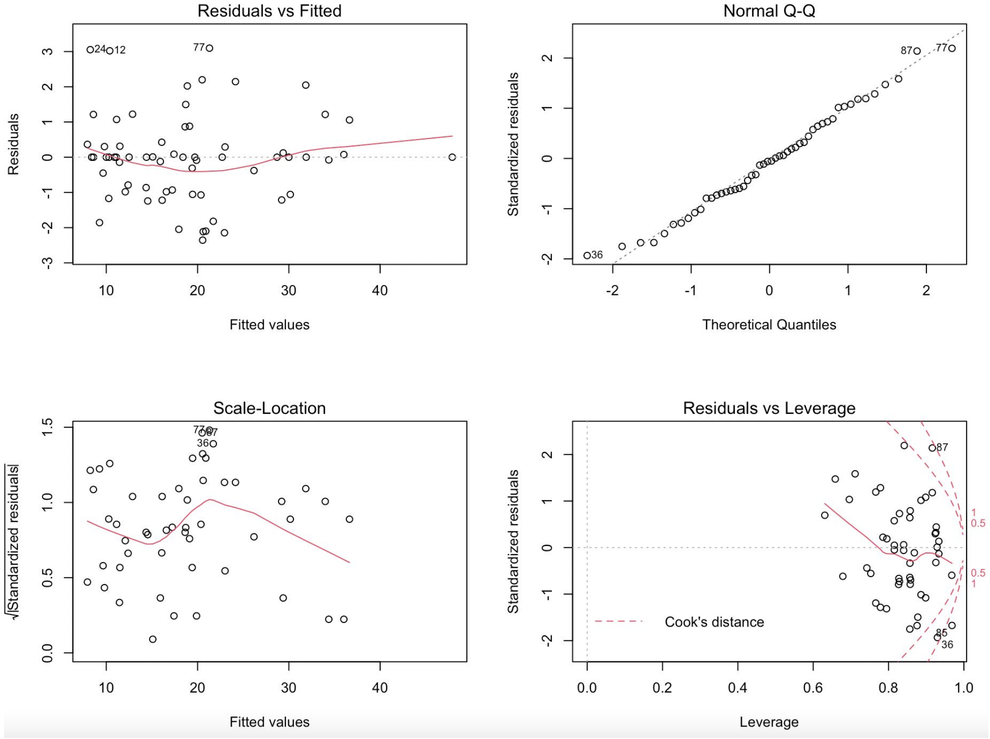
With remaining 21 variables

Evaluation metrics:

- **R2:** 0.978
- **Adjusted R2:** 0.83
- **RMSE:** 1.25

Coefficient interpretation:

- MPG.city: 0.81
- Passengers: 1.13
- Type:
  - Large 10.72
  - Midsize 8.29
  - Small -2.33



Ridge Model

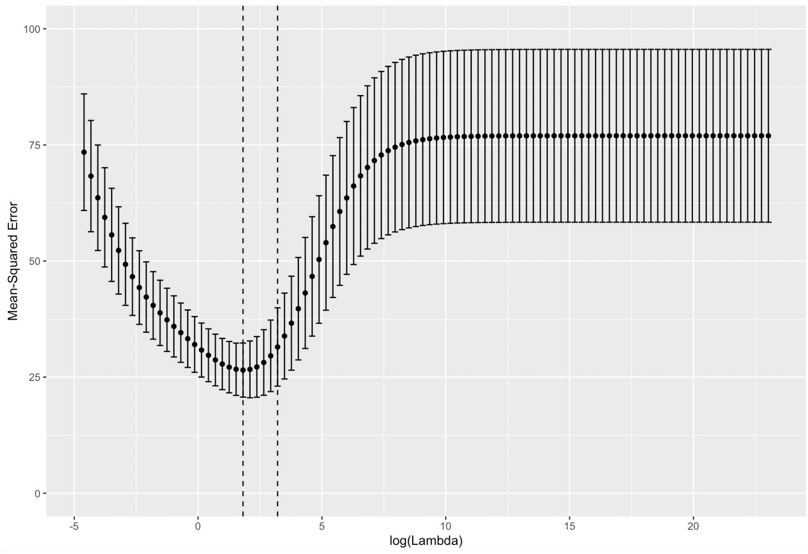
We created a Ridge model to try to solve collinearity

Evaluation metrics:

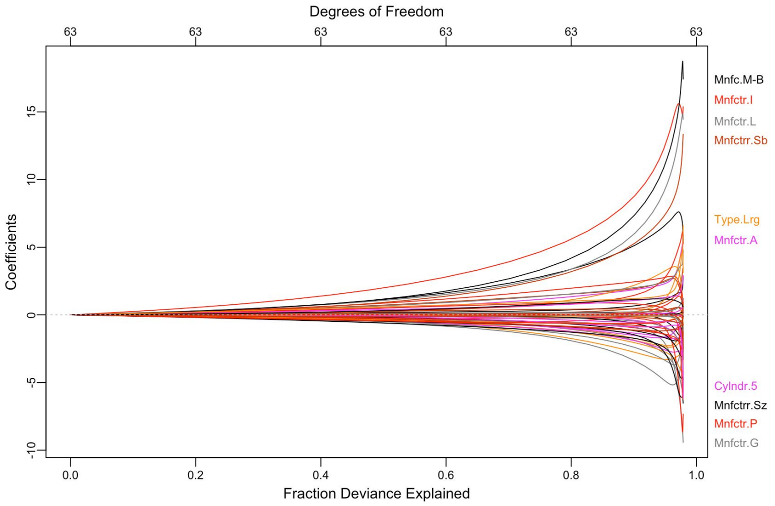
- **R2:** 0.90
- **RMSE:** 2.68

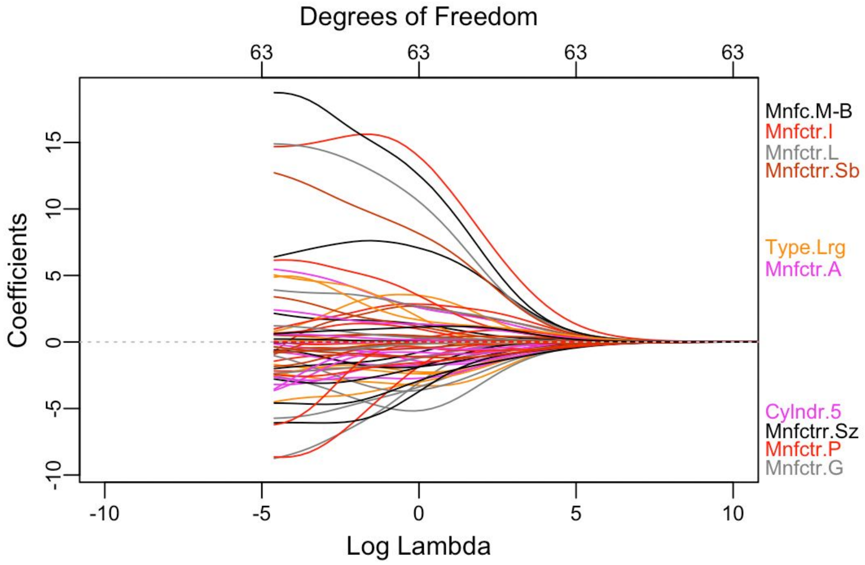
Best Lambda

- **Minimum lambda**: 3.14



Ridge Model


Degrees of Freedom
63
63
63
63
63
63
Coefficients
15
10
5
0
-5
-10
0.0
0.2
0.4
0.6
0.8
1.0
Fraction Deviance Explained
Mnfc.M-B
Mnfctr.I
Mnfctr.L
Mnfctrr.Sb
Type.Lrg
Mnfctr.A
Cylndr.5
Mnfctrr.Sz
Mnfctr.P
Mnfctr.G


Degrees of Freedom
63
63
63
63
Coefficients
15
10
5
0
-5
-10
-10
-5
0
5
10
Log Lambda
Mnfc.M-B
Mnfctr.I
Mnfctr.L
Mnfctrr.Sb
Type.Lrg
Mnfctr.A
Cylndr.5
Mnfctrr.Sz
Mnfctr.P
Mnfctr.G

Lasso Model

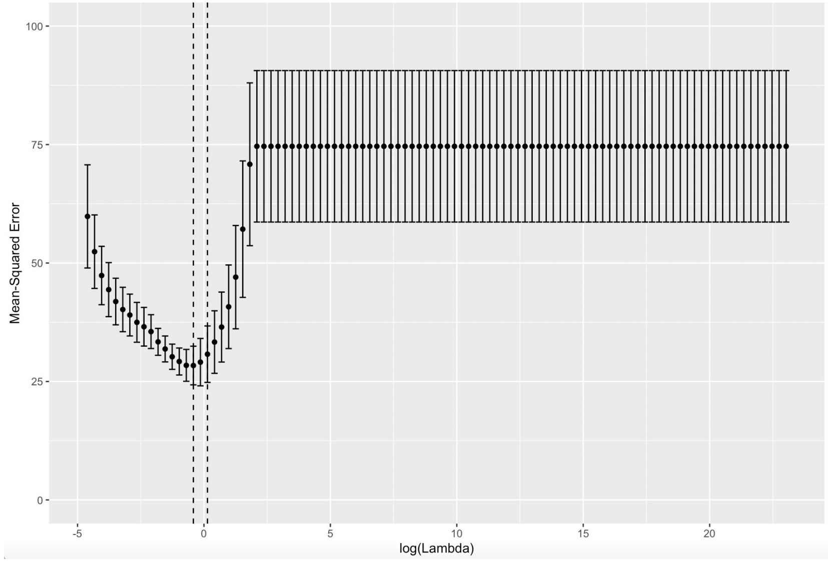
So, we created a lasso model to so if it improved

Evaluation metrics:

- **R2:** 0.89
- **RMSE:** 2.88

Best Lambda

- **Minimum lambda**: 0.66



Back to linear

And working with stepwise

- We decided to go back to the linear, which is the best model, and do a **stepwise** algorithm.

Price ~ Manufacturer + Type + MPG.city + Cylinders + Fuel.tank.capacity + Width + Rear.seat.room + Luggage.room + Weight

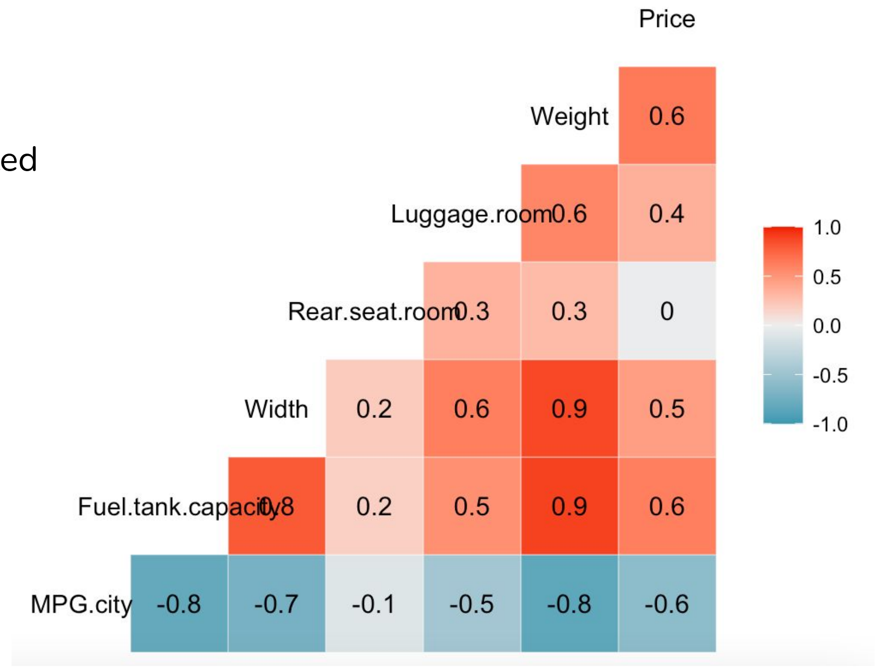
Evaluation metrics:

- **R2:** 0.976
- **Adjusted R2: 0.919**
- **RMSE:** 1.324

Back to linear

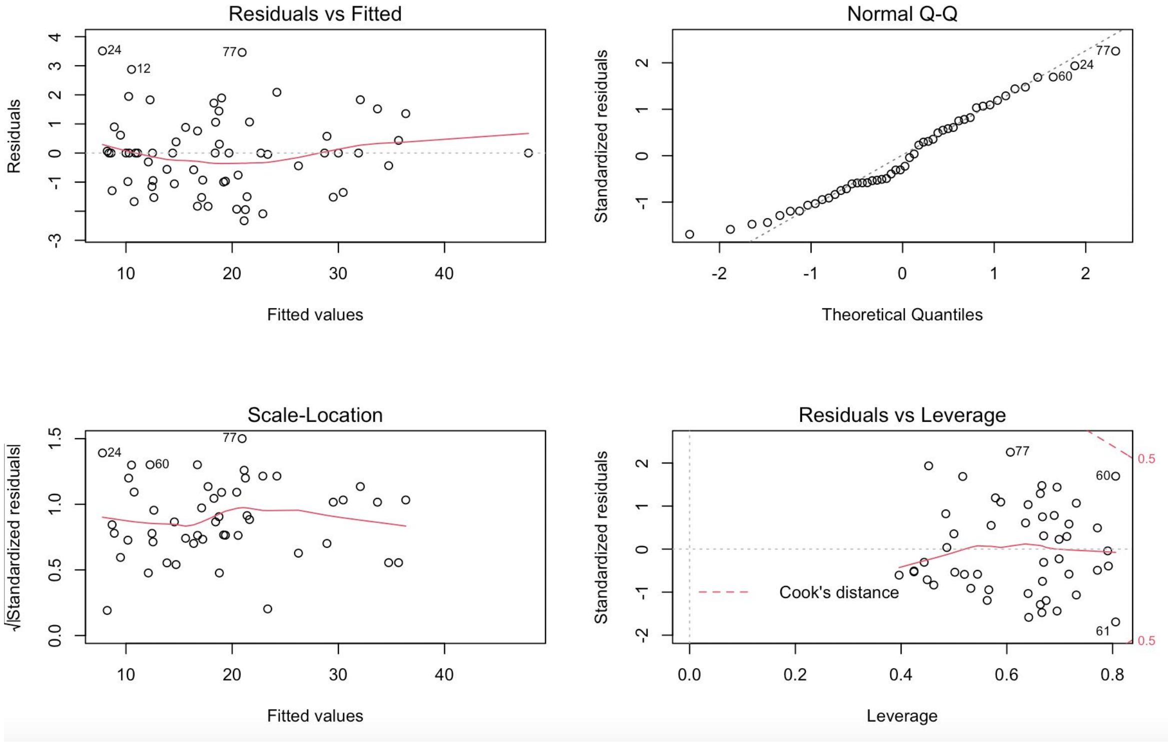
Collinearity

- Most of the variables are less correlated between each other



Back to linear

Model evaluation



Evaluation Metrics Comparison

All models

	Linear	Ridge	Lasso	Linear after AIC
R2	0.978	0.90	0.89	0.976
Adjusted R2	0.83			**0.919**
RMSE	1.25	2.68	2.88	**1.324**

Conclusion

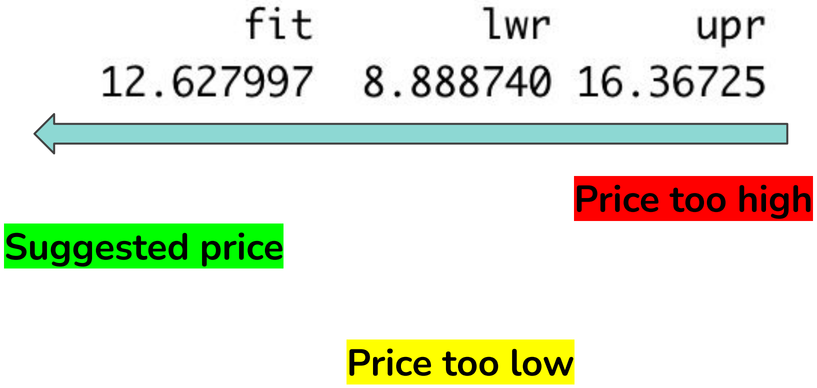
Final thoughts on model

- We tested different models and concluded that **linear model**, after stepwise (both directions) was the **best model**
- We were able to get a prediction of cars prices with very good **precision** (low RMSE, high R2)
- Our model adjusts well, is **homoskedastic** and the residuals are normally disitibuted

Conclusion

Implementation

- By using the predictions made by this model, we are able to suggest the seller a recommended price for their used car based on its main features
- Giving the user a price range based on a 90% confidence interval
- We offer flexibility in price setting and searching by providing user friendly advice on our platform
- We plan to test MVP and incorporate this feature in the following months



```
      fit      lwr      upr
12.627997 8.888740 16.36725
```

Price too high

Suggested price

Price too low

THANK YOU FOR YOUR ATTENTION