Kai Broux (r0745431), Luigi Guerriero (r0760424)

A second-hand car price prediction system of cars in the UK

1 Introduction

The production of cars has been steadily increasing in the past decade and this has given rise to the used car market. The emergence of online second-hand car portals in the United Kingdom and Europe has facilitated the need for both the customer and the seller to be better informed about trends and patterns that determine the value of a used car [1]. A second-hand car price prediction system (for a specific car brand) is thus required to effectively determine the worthiness of the car using a variety of features. The price prediction model and insights/patterns could be then later used as a tool for a second-hand car retailer to give insights to potential customers shopping for a second-hand car.

To tackle the need of a price prediction system, we are going to divide the price prediction problem into two subproblems: a price prediction model for a specific car brand (Mercedes C-Class W205) and a price prediction model for an entire brand (Mercedes). Our main goal is thus to make a price prediction model both for a specific model and an entire brand.

For our first price prediction model of the Mercedes C-Class W205, we predict the price (output) based on the mileage (input). We first implement regularized linear regression and will further improve our model by introducing polynomial regression. For our second price prediction model of Mercedes cars, we predict the price (output) based on several features (year, fuel type, transmission type, etc.) by using multivariate regression.

2 Related work

Many research used regression techniques, but other techniques such as neural networks (NN), Support Vector Machines (SVM) and decision trees were also used.

The work of Noor K. and Jan S. [2] used multiple linear regression to make a price prediction model for second hand cars in India. The research performed several variable selection techniques in Minitab to extract the most useful features. The result of the prediction model was accurate with an r^2 of 98%. Similarly, the research of Kuiper S. [3] also used multiple linear regression to make a price prediction model of GM-cars from 2005. The research of Kuiper used several variable selection techniques using Mintab such as Mallows's CP, Akaike information criterion, stepwise regression and manually exploring multicollinearity on raw data. Kuiper found that the Mallows's CP was the best for picking the features, but he also concluded that there does not exist a "best" regression model or variable selection technique that guarantees a "best" regression model. In addition, the research of Sameerchand P. [4] also used multiple linear regression for used cars in Mauritius and observed the correlation between features to select the ones used for the model. However, they concluded that their accuracy was relatively low because the dataset, which they collected from daily newspapers, had not enough data to make an accurate model.

Peerun et al. [5] did research on using NN in used car price prediction in Mauritius but concluded that their result was inaccurate. Their research had the same weakness as [4] because their dataset was not sufficiently large. This does not mean that NN performs badly on price prediction. Sun et al. [6] used NN for a car price prediction model and introduced a new optimization method called Like Block-Monte Carlo Method (LB-MCM) to optimize hidden neurons. The optimized NN model yielded higher accuracy compared to other work using NN.

According to the research Listiani [7] for predicting the price of leased cars, SVM proves to yield a higher accuracy than both NN and multiple regression. When a large dataset is available, she found that SVM is considerably more accurate than multiple linear regression

in predicting prices. SVM is also superior at handling high-dimensional data and avoids both underfitting and overfitting problems.

Lastly, since we only made a model of one specific car and brand and not (yet) the entire dataset of different brands, there also exists research of one specific brand. The research of Erfan S. [8] used different techniques including decision trees, SVM, random forest and deep learning for predicting the price of a Tesla vehicle. He found that the decision tree yielded the best result.

3 Dataset and Features

We used the dataset from Aditya from Kaggle [9]. This dataset contains data from 100 000 used cars from the UK which are divided into separate brands. Each brand is divided into a separate file and contains information about the model, year, price, transmission, mileage, fuel type, road tax, mpg and engine size.

For the price prediction model of the Mercedes C Class W205 we used the separate cclass.csv file and for the price prediction model of all the Mercedes cars we used the merc.csv file. Table 1 shows the records for the C Class dataset. The records of the Mercedes dataset are similar, but with different models. An example of some records is shown in table 1 (of the C Class dataset).

Table 1: records of the C Class dataset

	model	year	price	transmission	mileage	fuelType	engineSize
0	C Class	2020	30495	Automatic	1200	Diesel	2.0
1	C Class	2020	29989	Automatic	1000	Petrol	1.5
2	C Class	2020	37899	Manual	500	Diesel	2.0
3	C Class	2019	30399	Automatic	5000	Diesel	2.0

Since we only want to predict the price for the W205 C-Class we filtered the C-Class dataset to only include records ranging from 2014-2020. Secondly for both the C-Class and Mercedes dataset we filtered out the small amount of high prices of both datasets since they do not represent the majority of the prices but can influence our accuracy.

Figure 1a shows the price density plot for the C Class dataset before filtering, and figure 1b. shows the price density plot for the C Class dataset after filtering (only keeping records ranging from £0-50 000). The Mercedes dataset was filtered too (only keeping records ranging from £0-60 000).

We further split our C Class dataset into a training/cross-validation/test-set consisting of 2137, 713, 713 records respectively.

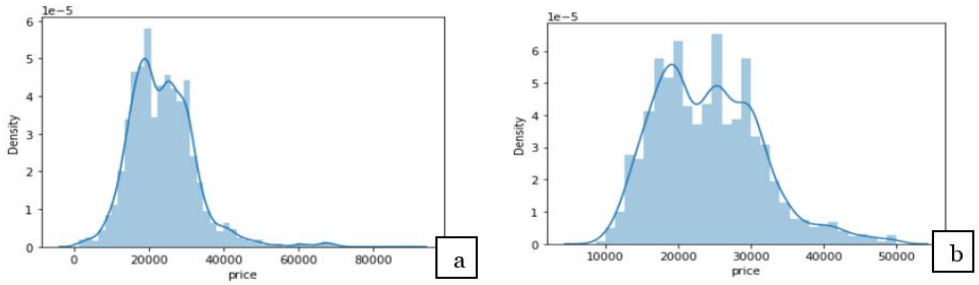

Figure 1: price density C Class dataset before filtering (a) and after filtering (b)

In addition, only for our Mercedes dataset, we transformed columns that contain more than 2 different non-numerical values (car-models, transmission, fuel type) into separate columns

with binary values to help ease our further calculations as shown in table 2.

Lastly, both the C Class and Mercedes features from both datasets were also normalized.

Table 2: records of the transformed Mercedes dataset

	year	price	mileage	tax	mpg	engineSize	model_ A Class	model_ B Class	...	transmission _Semi-Auto	fuelType_Die sel
0	2005	5200	63000	325	32.1	1.8	0	0	...	0	0
1	2017	34948	27000	20	61.4	2.1	0	0	...	0	0
2	2016	49948	6200	555	28.0	5.5	0	0	...	0	0

4 Methods

4.1 Linear regression with one variable

The first model that is used in the project is linear regression with one variable. This algorithm can be applied in two different ways: with or without regularization.

4.1.1 Without regularization

Formula (1) shows the cost function of linear regression without regularization and formula (2) shows the gradient descent function. For gradient descent it is important to repeat the formula until convergence and to update all θ simultaneously.

$$J(\theta_0, \theta_1) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2 \text{ with } h_\theta(x) = \theta_0 + \theta_1 x \quad (1)$$

$$\theta_j = \theta_j - \alpha\frac{\partial}{\partial\theta_j}J(\theta_0, \theta_1) \qquad (\text{for } j = 1 \text{ and } j = 0) \quad (2)$$

The goal is to minimize the cost function with the help of the gradient descent function to find an appropriate theta. The dot-product of this theta with the normalized X (stacked with ones) will give the predicted Y. This predicted Y can then be plotted in a graph.

4.1.2 With regularization

Formula (3) shows the cost function with regularization.

$$J(\theta) = \frac{1}{2m}\left[\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2 + \lambda\sum_{j=1}^{n}\theta_j^2\right] \quad (3)$$

The goal is again to minimize the cost function but instead of using gradient descent, another algorithm (scipy.optimize.minimize) is used this time. However, a proper regularization factor, λ, has to be selected before minimizing. This will be done by plotting the training- and cross validation error for different values of lambda. The predicted Y can then be calculated and plotted the same way as before.

Finally, this model will be analyzed with three different methods. The first method is to calculate the R^2-score. The second method is to plot the learning curve. This learning curve will plot the training- and cross validation error for the training examples. The last method is to plot the error histogram of the cross validation set.

4.2 Polynomial regression

The second model that is used in the project is polynomial regression. The main difference compared to linear regression with one variable is the hypothesis function. Formula (4) shows the hypothesis for polynomial regression. Only the version of regularization has been applied for this model as it provides a better result.

$$h_\theta = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_p x_p \quad \text{with } p = \text{the degree} \quad (4)$$

$$= \theta_0 + \theta_1(mileage) + \theta_2(mileage)^2 + \cdots + \theta_p(mileage)^p$$

The first thing that has to be calculated is the degree, p. This will be done by plotting the training- and cross validation error for different polynomial degrees. Following, the regularization parameter, λ, has to be calculated again. The same approach as for linear regression with one variable is used, thus by plotting the training- and cross validation error for different values of lambda. The last step of this model is to calculate and plot the predicted Y. This is also done in the same way as for linear regression with one variable, that means by taking the dot-product of the normalized X (stacked with ones) and theta. Theta is again calculated by the scipy.optimize.minimize algorithm.

Finally, this model will be analyzed with the same three different methods as for linear regression with one variable. That is, with the R^2-score, by plotting the learning curve and by plotting the error histogram of the cross validation set.

4.3 Linear regression with multiple variables

The last model that is used in this project is linear regression with multiple variables. The biggest difference compared to the previous models is again the hypothesis function. Formula (5) shows the hypothesis for linear regression with multiple variables. No regularization has been used for this model.

$$\begin{aligned} h_\theta &= \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n \quad \text{with } n = \text{number of features} \\ &= \theta_0 + \theta_1(year) + \theta_2(mileage) + \cdots + \theta_n(mpg) \end{aligned} \tag{5}$$

The first thing that needs to be done is to select how many and which features. To calculate this, the sklearn library will be utilized. The next step is to calculate the predicted Y. This will once again be done like the previous models. That means by taking the dot-product of the normalized X (stacked with ones) and theta. Theta is this time calculated with the gradient descent function, similar to linear regression without regularization. At last, α and the number of iterations can be checked by plotting the cost, J, in function of the number of iterations.

Finally, this model will also be analyzed, but only with two different methods. Firstly, by calculating the R^2-score and secondly by plotting the error histogram.

5 Results/Discussion

5.1 Linear regression with one variable

Figure 2a and 2b shows the mileage in miles in function of the price of a C-Class in pounds. Figure 2a shows the linear regression with one variable model without regularization and figure 2b shows the model with regularization.

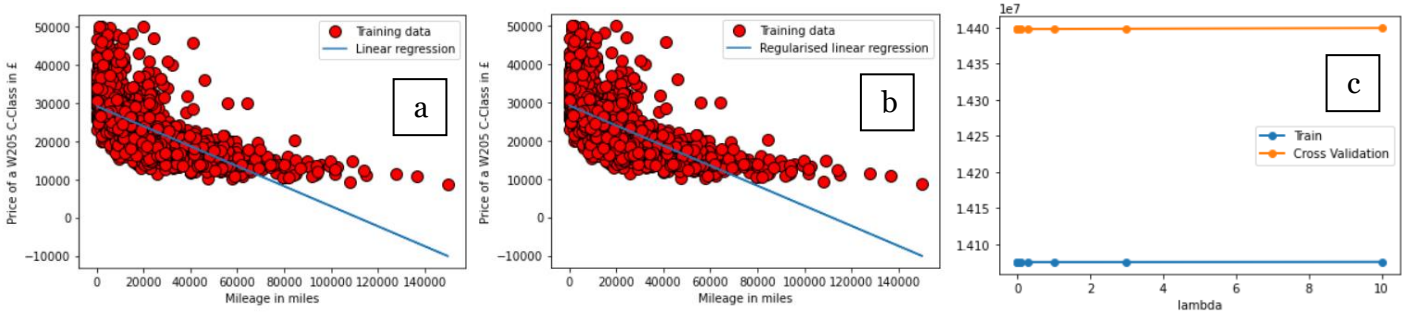

Figure 2: mileage (miles) in function of price (pounds) for linear regression with one variable without (a) and with (b) regularization and training- and cross validation error for different λ (c)

There is no great difference between figure 2a and 2b, which means that regularization does not offer an added value. In general, it is clear that this model does not fit the data properly.

Figure 2c shows the error for different values of λ. The error is almost exactly the same for all different λ. That means it does not matter which value lambda is set to. For this model λ will stick to 0.

Figure 3a shows the learning curve for linear regression with one variable with regularization. Figure 3b shows the error histogram of the cross validation set.

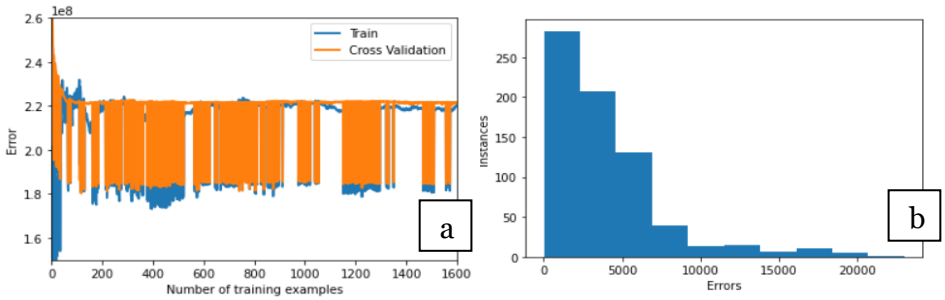

Figure 3: learning curve (a) and error histogram of the cross validation set (b).

The learning curve shows a large train- and cross validation error, but the gap between the two errors is rather small. The error histogram also shows large errors.

This model suffers a lot from high bias. This can be concluded in various ways. To start with, the regularization factor does not offer any added value, which means that there is certainly no high variance. Also, the error from the learning curve and from the error histogram are very high, which indicates that there is high bias. Finally, the R^2-score of this model is around 0.46. This is a considerably low value, which again indicates that there is high bias.

5.2 Polynomial regression

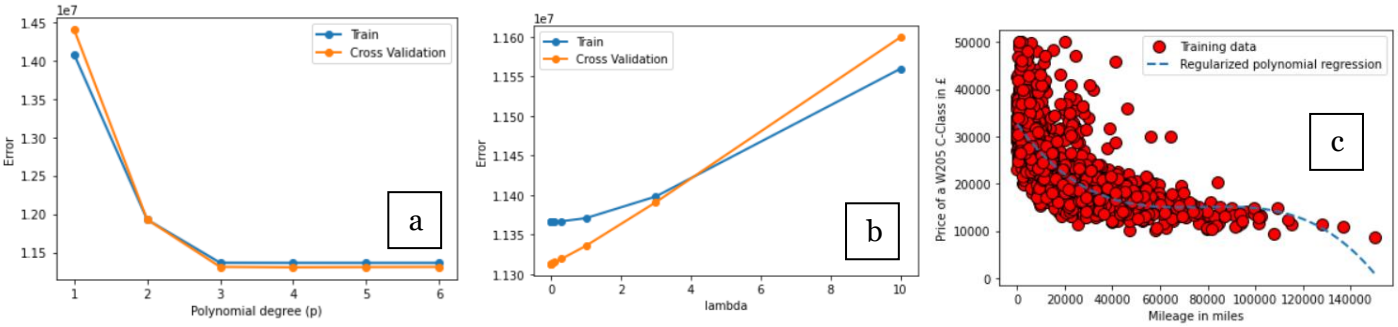

Figure 4: error for different polynomial degrees (a), error for different lambda's (b) and mileage (miles) in function of price (pounds) for polynomial regression (c)

Figure 4a shows the error for different polynomial degrees. It is clearly noticeable that the error is the smallest at a degree of 3. Figure 4b shows the error for different lambda's. The error is this time the smallest at a lambda of 0. Therefore p is set to 3 and lambda to 0.

Figure 4c shows the mileage in miles in function of the price of a C-Class in pounds. It also shows the polynomial regression model (with regularization). This model fits the data better than the previous model, but it is still not the best fit.

Figure 5a shows the learning curve of the polynomial regression model. The error is smaller than the previous model, but it is still significant. The gap between the train- and cross validation error is again very small. Figure 5b shows the error histogram of the cross validation set. The error, similar to the previous model, is large but this time there are more relatively small errors.

Like the previous model, this model also suffers from high bias. The R^2-score for this model is approximately 0.56, which is a bit larger than the previous model. This combined with the smaller errors indicates that the polynomial regression model is a better fit than the linear regression model with one variable, but it is far from a perfect fit.

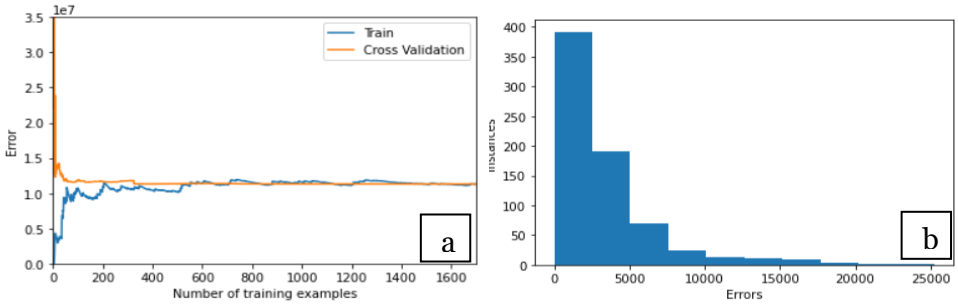

Figure 5: learning curve (a) and error histogram of the cross validation set (b).

5.3 Linear regression with multiple variables

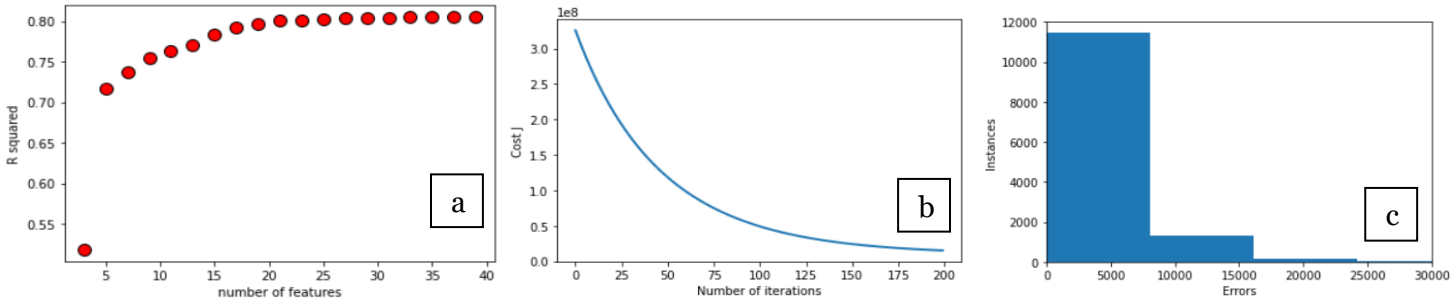

Figure 6: predicted R^2-score for different numbers of features (a), number of iterations in function of the cost J (b) and error histogram (c)

Figure 6a shows a predicted R^2-score for different numbers of features. The (predicted) R^2-score remains constant around 20 features, which means that 20 features are enough to go on. Figure 6b shows the number of iterations in function of the cost. The cost decreases to almost 0, which indicates that alpha and the number of iterations are chosen properly. Alpha is set to 0.01 and the number of iterations is set to 200. Figure 6c shows the error histogram of the linear regression with multiple variables model. It is noticeable that there are numerous relatively small errors, but fewer serious errors compared with previous models.

The R^2-score of the linear regression with multiple variables is 0.67, which is higher than the previous model but still not particularly high. However, this score combined with the smaller number of major errors means that this model fits the data most accurately.

6 Conclusion/Future Work

In this work, a price prediction model was made of the W205 C Class and Mercedes cars in general using linear/polynomial regression and linear regression with multiple variables. It makes sense that the linear regression model performed the worst with an R^2 score of 0.46 because there almost is no linear relation between the price and mileage. The polynomial regression model showed a significant improvement, but still relatively low R^2 score of 0.56. The multiple regression model of the Mercedes brand had the higher R^2 score of 0.67.

The research showed a clear evolution of techniques used in increasing accuracy. However, as stated in other research, other methodologies can also be used (such as SVM) to obtain a higher r^2. Also, because we manually wrote the functions, the use of external optimized libraries could further improve accuracy, as similar work on Kaggle yielded much a higher r^2. Apart from the used techniques, the data could be further cleaned, by for example removing outliers, to obtain a higher accuracy. Polynomial regression, should yield a higher r^2 value than we obtained, but was restrained by unclean data. In addition, our used techniques could easily be expanded to the other brands and even the entire dataset combined to gain more insights.

7 Contributions

Both team members have contributed approximately the same to this project. Here is a list with the topics each member contributed the most on. In general we have worked together for most topics so it does not mean that the one person has done everything on one specific topic.

Luigi (50%):

- Insights of the dataset + loading in the dataset
- Data filtering and normalization + dividing into train, CV and test
- Linear regression with one variable (with regularization + choosing hyper parameters)
- Analyzing linear regression with one variable (R^2, learning curve and error histogram)
- Report + presentation

Kai (50%):

- Polynomial regression (+ choosing hyper parameters)
- Analyzing polynomial regression (R^2, learning curve and error histogram)
- Linear regression with multiple variables (+ checking features and alpha)
- Analyzing linear regression with multiple variables (R^2 and error histogram)
- Report + presentation

8 References

[1] P. Venkatasubbu en G. Mukkesh, „Used Cars Price Prediction using Supervised Learning Techniques," *International Journal of Engineering and Advanced Technology (IJEAT),* vol. 2019, nr. 3, pp. 216-223, 2019.

[2] N. Kanwal en S. Jan, „Vehicle Price Prediction System using Machine Learning Techniques," *International Journal of Computer Applications,* vol. 9, nr. 167, pp. 27-31, 2017.

[3] K. Shonda, „Introduction to Multiple Regression: How Much Is Your Car Worth?," *Journal of Statistics Education,* vol. 3, nr. 16, 2008.

[4] S. Pudaruth, „Predicting the Price of Used Cars using Machine Learning Techniques," *International Journal of Information & Computation Technology,* vol. 4, nr. 7, pp. 753-764, 2014.

[5] S. Peerun, N. H. Chummun en S. Pudaruth, „Predicting the Price of Second-hand Cars using Artificial Neural," in *The Second International Conference on Data Mining*, University of Mauritius, Reduit, Mauritius, 2015.

[6] N. Sun, H. Bai, Y. Geng en H. Shi, „Price evaluation model in second-hand car system based on BP neural network theory," in *18th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*, United States, 2017.

[7] M. Listiani, „Support Vector Regression Analysis," *M.S. Thesis, Institute of Software, Technology, and Systems, Hamburg University of Technology,* 2009. [Online] Available: citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.151.6198&rep=rep1&type=pdf

[8] S. E. Arefin, „Second Hand Price Prediction for Tesla Vehicles," *arXiv preprint arXiv:2101.03788,* pp. 1-8, 2021.

[9] Aditya, „Kaggle," 2020. [Online]. Available: https://www.kaggle.com/adityadesai13/used-car-dataset-ford-and-mercedes. [Opened 19 December 2021].